



Article

Hyperspectral Image Mixed Noise Removal via Double Factor Total Variation Nonlocal Low-Rank Tensor Regularization

Yongjie Wu ^{1,2,3} , Wei Xu ^{1,3,*} and Liangliang Zheng ^{1,3}

¹ Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China; wuyongjie18@mails.ucas.ac.cn (Y.W.); zhengliangliang@ciomp.ac.cn (L.Z.)

² University of Chinese Academy of Sciences, Beijing 100049, China

³ Key Laboratory of Space-Based Dynamic & Rapid Optical Imaging Technology, Chinese Academy of Sciences, Changchun 130033, China

* Correspondence: xuwei@ciomp.ac.cn

Abstract: A hyperspectral image (HSI) is often corrupted by various types of noise during image acquisition, e.g., Gaussian noise, impulse noise, stripes, deadlines, and more. Thus, as a preprocessing step, HSI denoising plays a vital role in many subsequent tasks. Recently, a variety of mixed noise removal approaches have been developed for HSI, and the methods based on spatial–spectral double factor and total variation (DFTV) regularization have achieved comparable performance. Additionally, the nonlocal low-rank tensor model (NLR) is often employed to characterize spatial nonlocal self-similarity (NSS). Generally, fully exploring prior knowledge can improve the denoising performance, but it significantly increases the computational cost when the NSS prior is employed. To solve this problem, this article proposes a novel DFTV-based NLR regularization (DFTVNLR) model for HSI mixed noise removal. The proposed model employs low-rank tensor factorization (LRTF) to characterize the spectral global low-rankness (LR), introduces 2-D and 1-D TV constraints on double-factor to characterize the spatial and spectral local smoothness (LS), respectively. Meanwhile, the NLR is applied to the spatial factor to characterize the NSS. Then, we developed an algorithm based on proximal alternating minimization (PAM) to solve the proposed model effectively. Particularly, we effectively controlled the computational cost from two aspects, namely taking small-sized double factor as regularization object and putting the time-consuming NLR model before the main loop with fewer iterations to solve it independently. Finally, considerable experiments on simulated and real noisy HSI substantiate that the proposed method is superior to the related state-of-the-art methods in balancing the denoising effect and speed.

Keywords: hyperspectral image; mixed noise removal; low-rank tensor factorization; spatial–spectral double factor; total variation; nonlocal low-rank model; alternating direction method of multipliers; augmented Lagrangian method; proximal alternating minimization



Citation: Wu, Y.; Xu, W.; Zheng, L. Hyperspectral Image Mixed Noise Removal via Double Factor Total Variation Nonlocal Low-Rank Tensor Regularization. *Remote Sens.* **2024**, *16*, 1686. <https://doi.org/10.3390/rs16101686>

Academic Editor: Farid Melgani

Received: 10 April 2024

Accepted: 2 May 2024

Published: 9 May 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

A hyperspectral image (HSI) consists of contiguous spectral band images from ultraviolet to infrared wavelengths [1,2]. Therefore, HSI contains abundant spatial and spectral information and plays an important role in various applications [3–7], such as military surveillance, natural disaster monitoring, terrain detection, mineral exploration, and food safety detection. Nevertheless, due to the interference of the detector sensitivity, platform vibrations, and atmospheric turbulence, a real HSI is inevitably affected by various noises, such as Gaussian noise, impulse noise, stripes, deadlines, etc. [8,9]. The existence of the above noises greatly degrades the quality of the HSI, limiting the subsequent tasks, such as classification [10], unmixing [11], fusion [12], feature learning [13], super-resolution [14], and target detection [15]. Hence, HSI denoising is a fundamental preprocessing step for further applications. In the field of remote sensing (RS), it is generally assumed that the

mixed noise in optical remote sensing images is mainly additive noise, and multiplicative noise often appears in synthetic aperture radar (SAR) images. Recently, there have been many studies on denoising based on this assumption, and it can basically meet the actual denoising needs. Therefore, this article only focuses on the discussion of removing the hyperspectral mixed noise composed of additive noise.

Currently, HSI denoising approaches can be generally divided into two categories, namely model-based methods and data-based methods [2]. The present challenge of data-based methods is the lack of real HSIs that can serve as training data [16,17]. In addition, although the speed of data-based methods is faster, the cost of the previous model training stage is higher, and the universality of the training model is limited. In contrast, model-based methods based on prior knowledge have a good general performance [8,16]. The improvement in the denoising performance is closely related to the mining of prior knowledge more fully, but more prior terms inevitably lead to the increase in the computational cost. The HSI denoising methods listed below refer to model-based methods, unless otherwise specified. Recently, many advanced HSI denoising methods have been proposed, among which the works exploring and modeling the intrinsic prior knowledge have made the most outstanding achievements. The most important priors mainly include spectral global low-rankness (LR) [18], spatial-spectral local smoothness (LS) [19], spatial nonlocal self-similarity (NSS) [20], and the sparsity of sparse noise (S).

In recent years, more and more researchers have paid more attention to removing mixed noise. With the prosperity of optimization theory, methods based on low-rankness have been developed. For instance, many methods utilized tensor decomposition to exploit LR by treating the HSI as a third-order tensor [21–23]. Moreover, various total variation (TV) regularizations were introduced to remove noises and preserve the image edges. For instance, Chang et al. [24] proposed a model based on anisotropic spatial-spectral TV (SSTV) to promote LS. Fan et al. [25] employed SSTV and low-rank tensor factorization to remove the mixed noise of HSIs without changing the spatial structures. In order to further explore NSS, models based on nonlocal low-rank tensor decomposition and TV regularization were proposed in [26–28]. However, using either spatial LS or spatial-spectral LS will unavoidably result in oversmoothing or artifacts [29].

Usually, the methods based on low-rank TV (LRTV) are good at characterizing the LR and LS of HSIs, while at the expense of a higher computational cost [30]. To be clear, there are two main reasons for the above high computational cost. First, the cost of directly imposing constraints on HSIs with a large data volume is higher. Second, the cost of a high-performance denoising model based on fully mining prior knowledge is higher. Compared to panchromatic and infrared RS images, an HSI $\mathcal{X} \in \mathbb{R}^{M \times N \times B}$ has a larger spectral dimension B (usually several hundred). Therefore, this article follows the convention of [2,31] and distinguishes based on the size of spectral dimension, assuming that the HSI is “large-sized”, while the other RS images and the factors $\mathcal{U} - \mathbf{V}$ mentioned below with relatively small spectral dimensions are “small-sized”. Is it possible that the original HSI has a factor with a smaller size that inherits the prior knowledge of \mathcal{X} ? If this is true, it is hoped that we can reduce computational costs while guaranteeing the denoising performance. Inspired by this aim, we turned our attention to low-rank matrix factorization (LRMF)/low-rank tensor factorization (LRTF). For an HSI $\mathcal{X} \in \mathbb{R}^{M \times N \times B}$, its unfolded matrix along the spectral dimension can be expressed as $\mathbf{X} \in \mathbb{R}^{MN \times B}$, and the rank of \mathbf{X} is R ($R \ll B$). Then, \mathcal{X} and \mathbf{X} can be factorized as $\mathcal{X} = \mathcal{U}_3 \mathbf{V}$ and $\mathbf{X} = \mathbf{U} \mathbf{V}^T$, respectively, where $\mathbf{V} \in \mathbb{R}^{B \times R}$ denotes the spectral factor and $\mathcal{U} \in \mathbb{R}^{M \times N \times R} / \mathbf{U} \in \mathbb{R}^{MN \times R}$ corresponds to the spatial factor [32]. The LRTF/LRMF model can reflect LR and has the advantage of not requiring large-scale singular value decomposition (SVD) calculation [2]. Furthermore, the image corresponding to \mathcal{U} / \mathbf{U} can maintain the structure of the original image better [2].

Therefore, under the framework of LRMF/LRTF, several HSI denoising methods based on spatial single-factor \mathcal{U} or spatial-spectral double-factor $\mathcal{U} - \mathbf{V}$ regularization have been proposed [2,26,30,31,33,34]. For example, the LRTFDFR method proposed by

Zheng et al. [31] applied the weighted TV l_{21} norm and the data fidelity term to \mathcal{U} and \mathbf{V} , which describe the spatial and spectral LS, respectively, but it did not consider NSS. The RCTV proposed by Peng et al. [2] applied the TV l_1 norm to \mathbf{U} to describe the spatial LS, but it did not consider the spectral LS and NSS. The SNLRSF method proposed by Cao et al. [30] applied the nonlocal low-rank decomposition to \mathbf{U} to describe NSS, but it did not consider LS. The NLSSR method proposed by Zha et al. [35] characterized the NSS and LR of HSIs with nonlocal structured sparsity regularization, but it ignored LS and S. The LRTRDGS method proposed by Wang et al. [36] characterized the LR, LS, and S with weighted group sparsity-regularized low-rank tensor ring decomposition, but it did not consider NSS. In short, almost all methods fail to fully explore the prior knowledge of HSIs and noise. Furthermore, the DFTCR method proposed by Han et al. [37] fully explored the priors of HSIs and noise, but it did not balance the high computational cost caused by introducing an NSS prior. The related research conclusion in [2] also shows that the circular search and matching strategy of similar blocks often used to characterize NSS greatly increase the computational cost. In summary, there is still room to improve the denoising performance based on the existence of HSIs in the low-dimensional subspace.

Based on the above discussion, this article aims to recover the clean HSI \mathcal{X} from the observation data \mathcal{Y} polluted by additive mixed noise. In this regard, we propose a spatial-spectral double factor TV-based nonlocal low-rank tensor regularization method (DFTVNLR) to remove HSI mixed noise. We use the LRTF framework $\mathcal{X} = \mathcal{U}_{\times 3} \mathbf{V}$ to characterize the LR. Additionally, the TV regularization is added to the spatial factor \mathcal{U} and spectral factor \mathbf{V} to characterize the spatial bidirectional LS and spectral LS. Moreover, we introduce a weight strategy [18] to promote LS. Furthermore, we introduce the nonlocal low-rank tensor model (NLR) to characterize NSS and pre-estimate the initial value of \mathcal{U} . The contributions of this article are summarized in what follows:

1. First, under the framework of LRTF, by applying the weight TV regularization to the small-sized double factor $(\mathcal{U}-\mathbf{V})$, namely $\|\mathcal{W}_h \odot (\mathbf{D}_h \mathcal{U})\|_1 + \|\mathcal{W}_v \odot (\mathbf{D}_v \mathcal{U})\|_1$ and $\|\mathbf{D}_v \mathbf{V}\|_F^2$, it not only helps to reduce the computational cost, but also accurately characterizes the spatial-spectral LS of HSIs.
2. Second, we fully explore all priors of HSIs, such as LR, spatial-spectral LS, and NSS. And the NLR model is introduced before the main loop to estimate the initial value of \mathcal{U} , which indirectly characterizes NSS and solves the problem of high computational cost caused by the introduction of NLR.
3. Third, the alternating direction multiplier method (ADMM) and augmented Lagrangian method (ALM) are used to solve the proposed DFTVNLR. The experimental results show the proposed DFTVNLR is superior to several other excellent methods for mixed noise removal, spatial texture structure restoration, and spectral feature preservation.

2. Notations and Preliminaries

Throughout this article, we denote scalars, vectors, matrices, and tensors in light lowercase letters, bold lowercase letters, bold uppercase letters, and light uppercase cur-sive letters, respectively, e.g., x , \mathbf{x} , \mathbf{X} , and \mathcal{X} . A N -order tensor can be expressed as $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_N}$, whose element is denoted as $\mathcal{X}(i_1, i_2, \dots, i_N)$ or $\mathcal{X}_{i_1, i_2, \dots, i_N}$. The HSI can be viewed as a 3-D tensor $\mathcal{X} \in \mathbb{R}^{M \times N \times B}$. For the other notations used in this article, see Table 1 in [31]. Additionally, the definitions of mode- k unfolding ($\mathbf{X}_{(k)} = \text{unfold}_k(\mathcal{X})$, $\mathcal{X} = \text{Fold}_k(\mathbf{X}_{(k)})$), mode- k tensor-matrix product ($\mathcal{X}_{\times k} \mathbf{A}$), and the detailed introduction of the tensor can be found in [31,38]. The inner product of the two matrices \mathbf{X} and \mathbf{Y} with the same size is defined as $\langle \mathbf{X}, \mathbf{Y} \rangle := \sum_{i,j} x_{i,j} \cdot y_{i,j}$, where $x_{i,j}$ is (i, j) th element of \mathbf{X} [2]. Furthermore, $\|\mathbf{X}\|_F := \sqrt{\langle \mathbf{X}, \mathbf{X} \rangle}$ and $\|\mathbf{X}\|_1 := \sum_{i,j} |x_{i,j}|$ represent the Frobenius norm and l_1 norm, respectively [2]. The TV norm, which characterizes LS, is introduced below. Each

band of \mathcal{X} can be regarded as a gray image $\mathbf{X} \in \mathbb{R}^{M \times N}$. Therefore, the anisotropic TV norm [39] or the l_0 gradient [40] of \mathbf{X} is defined as follows:

$$l_0\text{TV}(\mathbf{X}) = \|\mathbf{X}\|_{\text{TV}} := \sum_{i=1}^M \sum_{j=1}^N \mathbf{C}[|x_{i+1,j} - x_{i,j}| + |x_{i,j+1} - x_{i,j}|] \quad (1)$$

$$= \sum_{i=1}^{M \cdot N} \mathbf{C}[\|\text{vec}(\mathbf{D}_h \mathbf{X})\|_1 + \|\text{vec}(\mathbf{D}_v \mathbf{X})\|_1]$$

where \mathbf{D}_h and \mathbf{D}_v represent first-order forward finite difference operators along the spatial horizontal and vertical modes, respectively. $\mathbf{C}(\cdot)$ is a binary function, and its specific definition can be found in [29]. The $\text{vec}(\cdot)$ stands for a matrix vectorization operator. For an HSI $\mathcal{X} \in \mathbb{R}^{M \times N \times B}$, the simplest and most direct TV model is $\sum_{k=1}^B \|\mathcal{X}(:, :, k)\|_{\text{TV}}$, in which the spectral LS of the HSI is neglected. The TV norm or l_0 gradient considering the spatial-spectral LS of the HSI is defined as follows [29]:

$$l_0\text{HTV}(\mathcal{X}) = \|\mathcal{X}\|_{\text{TV}} := \sum_{i=1}^M \sum_{j=1}^N \mathbf{C} \left\{ \sum_{k=1}^B \left[|\mathcal{X}_{i+1,j,k} - \mathcal{X}_{i,j,k}| + |\mathcal{X}_{i,j+1,k} - \mathcal{X}_{i,j,k}| \right] \right\} \quad (2)$$

Table 1. Computational complexity of solving the operators involved in common prior terms.

Prior Term	Subject	Algorithm	Computational Complexity
Spectral Global Low-Rankness (LR)	$\mathcal{X} \in \mathbb{R}^{M \times N \times B}$	SVD	$\mathcal{O}(MNB^2)$
	$\mathbf{V} \in \mathbb{R}^{B \times R}$	SVD	$\mathcal{O}(BR^2)$
Spatial-Spectral Local Smoothness (LS)	$\mathcal{X} \in \mathbb{R}^{M \times N \times B}$	2-D FFT	$\mathcal{O}[MNB \log(MN)]$
	$\mathcal{U} \in \mathbb{R}^{M \times N \times R}$	2-D FFT	$\mathcal{O}[MNR \log(MN)]$
	$\mathbf{V} \in \mathbb{R}^{B \times R}$	1-D FFT	$\mathcal{O}[BR \log(B)]$
Spatial Nonlocal Self-Similarity (NSS)	$\mathcal{X} \in \mathbb{R}^{M \times N \times B}$	SBS + SVD	$\mathcal{O} \left[\frac{M^2 N^2 B (\text{PatSize})^2}{(\text{StepSize})^4} + (\text{GroupNum}) B^2 (\text{PatNum}) (\text{PatSize})^2 \right]$
	$\mathcal{U} \in \mathbb{R}^{M \times N \times R}$	SBS + SVD	$\mathcal{O} \left[\frac{M^2 N^2 R (\text{PatSize})^2}{(\text{StepSize})^4} + (\text{GroupNum}) R^2 (\text{PatNum}) (\text{PatSize})^2 \right]$
Sparsity of Sparse Noise (S)	$\mathcal{S} \in \mathbb{R}^{M \times N \times B}$	SoftThreshold	$\mathcal{O}(MNB)$

$l_0\text{HTV}(\mathcal{X})$ has the obvious advantage of obtaining the spatial edge of the HSI [29]. Additionally, another TV norm considering the spatial-spectral LS of the HSI is SSTV, which can be used to restore most structural details of the HSI. It incorporated the 2-D spatial TV and 1-D spectral TV, and is defined as follows [41]:

$$\|\mathcal{X}\|_{\text{SSTV}} := \|\mathbf{D}\mathcal{X}\|_1 = \|\mathbf{D}_h \mathcal{X}\|_1 + \|\mathbf{D}_v \mathcal{X}\|_1 + \|\mathbf{D}_s \mathcal{X}\|_1 = \|\mathbf{D}_h \mathcal{X} \mathbf{D}_s\|_1 + \|\mathbf{D}_v \mathcal{X} \mathbf{D}_s\|_1 \quad (3)$$

where \mathbf{D}_s represents the first-order forward finite difference operator along the spectral mode, $\mathbf{D} = [\mathbf{D}_h, \mathbf{D}_v, \mathbf{D}_s]$ stands for the 3-D forward finite difference operator, and they are defined as follows [41]:

$$\mathbf{D}_h \mathcal{X} = \mathcal{X}_{i,j+1,b} - \mathcal{X}_{i,j,b}, \quad \mathbf{D}_v \mathcal{X} = \mathcal{X}_{i+1,j,b} - \mathcal{X}_{i,j,b}, \quad \mathbf{D}_s \mathcal{X} = \mathcal{X}_{i,j,b+1} - \mathcal{X}_{i,j,b} \quad (4)$$

3. HSI Mixed Noise Removal via DFTVNLRL

3.1. Problem Formulation

For simplicity, it is assumed a real observed noisy HSI is only corrupted by additive mixed noise, which typically includes additive Gaussian noise and additive sparse noise

(impulse noise, deadlines and stripes, etc.) [42]. Then, the HSI degradation process can be approximately formulated as an additive mixed system [2]:

$$\mathcal{Y} = \mathcal{X} + \mathcal{N} + \mathcal{S} \quad (5)$$

where $\mathcal{Y} \in \mathbb{R}^{M \times N \times B}$, $\mathcal{X} \in \mathbb{R}^{M \times N \times B}$, $\mathcal{N} \in \mathbb{R}^{M \times N \times B}$, and $\mathcal{S} \in \mathbb{R}^{M \times N \times B}$ denote the observed noisy HSI, the underlying clean HSI, additive Gaussian noise, and additive sparse noises, respectively. M , N , and B are the height of the spatial image, the width of the spatial image, and the number of spectral bands, respectively. The key task is to recover \mathcal{X} from \mathcal{Y} . Generally, the general HSI denoising model is expressed as follows [31]:

$$\min_{\mathcal{X}, \mathcal{S}} \frac{1}{2} \|\mathcal{Y} - \mathcal{X} - \mathcal{S}\|_F^2 + \lambda_1 \mathcal{R}(\mathcal{X}) + \lambda_2 \mathcal{R}(\mathcal{S}) \quad (6)$$

where $\frac{1}{2} \|\mathcal{Y} - \mathcal{X} - \mathcal{S}\|_F^2$ is a data fidelity term and indirectly characterizes Gaussian noise. $\mathcal{R}(\mathcal{X})$ is the regularization term associated with the clean HSI, which can be used to explore the LR, LS, and NSS of the HSI. $\mathcal{R}(\mathcal{S})$ is designed to suppress sparse noise, which can usually be characterized as l_0 -norm, l_1 -norm, or l_{21} -norm, etc. The parameters λ_1 and $\lambda_2 (\geq 0)$ are utilized to combine the three terms and to balance their contribution on the final result. The performance of a denoising model is heavily dependent on the choice of regularization, which encodes the prior of the HSI and noise. Nevertheless, adding regularization to the large-sized HSI unavoidably causes a high computational cost. Fortunately, due to the high correlation between the spectral bands of the HSI, it is reasonable to use LRTF (usually expressed as mode-3 tensor–matrix product) to characterize the LR prior of the large-sized clean HSI \mathcal{X} [43]:

$$\mathcal{X} = \mathcal{U}_{\times 3} \mathbf{V} \quad (7)$$

where $\mathcal{U} \in \mathbb{R}^{M \times N \times R}$ and $\mathbf{V} \in \mathbb{R}^{B \times R} (R \ll B)$ represent the small-sized spatial factor and the small-sized spectral factor, respectively. The symbol $\times 3$ represents the mode-3 tensor–matrix product. Specifically, the element in \mathcal{U} denotes the representation coefficient of \mathcal{X} with respect to \mathbf{V} . Particularly, \mathcal{U} is often called representation coefficient image (RCI), and its mode-3 slices are termed eigen-images [30].

Except for the LR, to further improve the denoising performance, most existing works adopt some other spatial priors of HSI, such as LS [9] and NSS [44]. Related research [2] has proven that the spatial information of original large-sized \mathcal{X} can be reflected on small-sized \mathcal{U} . Enlightened by this, we can explore the spatial priors on \mathcal{U} instead of \mathcal{X} . Additionally, \mathcal{U} is robust to noise, and the robustness is analyzed in E3DTV [19]. Furthermore, according to [20,45], \mathcal{U} can inherit two important properties of \mathcal{X} , namely the LR and the NSS. Therefore, many methods [2,30,31,46] based on the subspace and LRMF/LRTF have been proposed, in which various regularization $\mathcal{R}(\mathcal{U})$ are employed. Therefore, the general denoising model based on LRTF is established as follows [31]:

$$\min_{\mathbf{V}, \mathcal{U}, \mathcal{S}} \frac{1}{2} \|\mathcal{Y} - \mathcal{U}_{\times 3} \mathbf{V} - \mathcal{S}\|_F^2 + \lambda_1 \mathcal{R}(\mathcal{U}) + \lambda_2 \mathcal{R}(\mathcal{S}) \quad (8)$$

It is generally assumed that \mathbf{V} is orthogonal, i.e., $\mathbf{V}^T \mathbf{V} = \mathbf{I}$. However, the existing research [32] shows that, even if \mathbf{V} is not orthogonal, \mathcal{U} can also inherit the spatial prior information of \mathcal{X} to some extent. \mathbf{V} usually can be learned from \mathcal{Y} through the HySime algorithm [47]. In recent years, the priors of HSI denoising have been fully mined. However, in order to control the computational cost, only partial priors have been utilized at the same time. Therefore, the overall denoising performance of the HSI still has room for improvement. The key issue is how to balance the contradiction between fully representing priors and controlling the computational cost.

3.2. Proposed Denoising Model

3.2.1. Spectral Global Low-Rankness (LR) Characterized by LRTF

By comparing the HSI denoising performance based on the LR method [2,30,31,41,46] in recent years, it was found the LRTF can fully explore LR and has a certain noise robustness. Hence, we used Equation (7) to characterize LR.

3.2.2. Spatial–Spectral Local Smoothness (LS) Characterized by DFTV

The distribution of the SSTV model in Equation (3) is the same in all directions. However, in an actual scene, the different distributions of each band along the spatial–spectral three directions are different [9,19]. To solve this issue, based on the weighted strategy, some adaptive SSTV (ASSTV) models have been proposed [9,41]. The ASSTV model [41] selected in this article is defined as follows:

$$\|\mathcal{X}\|_{\text{ASSTV}} = \|\mathcal{W} \odot (\mathbf{D}\mathcal{X})\|_1 = \|\mathcal{W}_h \odot (\mathbf{D}_h\mathcal{X})\|_1 + \|\mathcal{W}_v \odot (\mathbf{D}_v\mathcal{X})\|_1 + \|\mathcal{W}_s \odot (\mathbf{D}_s\mathcal{X})\|_1 \quad (9)$$

where $\mathcal{W} = [\mathcal{W}_h, \mathcal{W}_v, \mathcal{W}_s] \in \mathbb{R}^{M \times N \times 3B}$ is the weight tensor used to adjust the smoothing intensity of different pixel positions in different directions.

Additionally, remarks 1 and 2 in [31] have proven that the LS constraint on small-sized $\mathcal{U} - \mathbf{V}$ can promote LS in \mathcal{X} . The simulation results of [31] also show that the strategy of regularizing the $\mathcal{U} - \mathbf{V}$ can enrich the spatial texture structure and smooth the spectral feature. In particular, the differential operations $\mathbf{D}_h\mathcal{U}$ and $\mathbf{D}_v\mathcal{U}$ of the third-order tensor \mathcal{U} are equivalent to the tensor–matrix product operations $\mathcal{U}_{\times 1}\mathbf{D}_h$ and $\mathcal{U}_{\times 2}\mathbf{D}_v$, respectively [31]. The symbol $\times 1/\times 2$ represents the mode-1/2 tensor–matrix product. Therefore, we obtained the idea of [31,41], and a double-factor TV model (DFTV) based on weight strategy is proposed, which is specifically defined as follows:

$$\begin{aligned} \|\mathcal{X}\|_{\text{DFTV}} &= \lambda_1 \|\mathcal{W}_h \odot (\mathbf{D}_h\mathcal{U})\|_1 + \lambda_1 \|\mathcal{W}_v \odot (\mathbf{D}_v\mathcal{U})\|_1 + \lambda_2 \|\mathbf{D}_s\mathbf{V}\|_F^2 \\ &= \lambda_1 \|\mathcal{W}_h \odot (\mathcal{U}_{\times 1}\mathbf{D}_h)\|_1 + \lambda_1 \|\mathcal{W}_v \odot (\mathcal{U}_{\times 2}\mathbf{D}_v)\|_1 + \lambda_2 \|\mathbf{D}_s\mathbf{V}\|_F^2 \end{aligned} \quad (10)$$

For the determination of the weight tensor \mathcal{W} , please refer to the calculation process in [41]. However, unlike [41], we calculated the weight tensor \mathcal{W} based on the denoised spatial factor $\hat{\mathcal{U}}$ instead of the denoised HSI $\hat{\mathcal{X}}$ during the iteration; so, the corresponding weight tensor is revised as follows:

$$\begin{cases} \mathcal{U}_0 = \text{LRA}_1(\hat{\mathcal{U}}) \\ \mathcal{W}_h = \frac{\max(|\mathbf{D}_h\mathcal{U}_0|)}{|\mathbf{D}_h\mathcal{U}_0| + \delta \max(|\mathbf{D}_h\mathcal{U}_0|)} = \frac{\max(|\mathcal{U}_{0 \times 1}\mathbf{D}_h|)}{|\mathcal{U}_{0 \times 1}\mathbf{D}_h| + \delta \max(|\mathcal{U}_{0 \times 1}\mathbf{D}_h|)} \\ \mathcal{W}_v = \frac{\max(|\mathbf{D}_v\mathcal{U}_0|)}{|\mathbf{D}_v\mathcal{U}_0| + \delta \max(|\mathbf{D}_v\mathcal{U}_0|)} = \frac{\max(|\mathcal{U}_{0 \times 2}\mathbf{D}_v|)}{|\mathcal{U}_{0 \times 2}\mathbf{D}_v| + \delta \max(|\mathcal{U}_{0 \times 2}\mathbf{D}_v|)} \end{cases} \quad (11)$$

where $\text{LRA}_1(\cdot)$ stands for the rank-1 low-rank approximation operator, and δ is a threshold set to avoid unexpected large values in the weight tensor [41].

3.2.3. Spatial Nonlocal Self-Similarity (NSS) Characterized by NLR

At present, there are two main methods to characterize the NSS prior of HSIs, including the method based on low-rank tensor decomposition [9,21] and the method based on patch low-rank approximation [28,48]. The literature [41] points out that the method based on patch low-rank approximation can better characterize the NSS of HSIs. However, it is extremely time-consuming to directly perform a full-band group matching operation on a large-sized \mathcal{X} , which is an important reason why many researchers choose to ignore the characterization of NSS. According to [20,45], the spatial factor \mathcal{U} (its forward slice is called feature image) can inherit two important properties of \mathcal{X} , namely the NSS of feature images and the LR between feature images.

Therefore, we adopted a patch-based nonlocal low-rank tensor model (NLR) proposed in [41], which takes small-sized \mathcal{U} as the operation object instead of large-sized \mathcal{X} . This

model first learns the spectral factor \mathbf{V} through the HySime algorithm and then applies the NLR model to the spatial factor \mathcal{U} , which can be expressed as follows. Note that we use P_i to represent the execution of similar block searching, matching, denoising, stacking, and other processes involved in the NLR model. The specific execution process can be found in the open source code provided in [41].

$$\|\mathcal{X}\|_{\text{NLR}} = \sum_{i=1}^P \|P_i \mathcal{U}\|_{w,*} \quad (12)$$

This method can reduce the computational cost by using the small-sized \mathcal{U} to characterize the NSS of the HSI. But, the repeated similar 3-D patch search and match of the NLR model in the main loop are still an important reason that affects the computational efficiency. For further reducing the computational cost, we chose to solve the NLR model outside the main loop and decreased the iteration times. This is not only an effective method to characterize NSS and roughly estimate the initial value of \mathcal{U} in the main loop at the same time, but also greatly reduces the cost.

3.2.4. DFTVNLR-Based HSI Denoising Model

Based on Sections 3.2.1–3.2.3, we preliminarily determined the relatively advanced strategy and model for characterizing the LR, LS, and NSS priors of the HSI as LRTE, DFTV, and NLR, respectively. The above three advanced models were obtained from previous studies, but there has been no research on coupling and improving these three advanced models to remove HSI mixed noise. Therefore, in order to balance the denoising performance and computational cost, we propose a DFTV-based NLR regularization (DFTVNLR) model for HSI mixed noise removal as follows:

$$\{\tilde{\mathcal{U}}, \tilde{\mathbf{V}}, \tilde{\mathcal{S}}\} = \min_{\mathcal{U}, \mathbf{V}, \mathcal{S}} \left\{ \begin{aligned} & \frac{1}{2} \|\mathcal{Y} - \mathcal{U}_{\times 3} \mathbf{V} - \mathcal{S}\|_F^2 + \lambda_1 [\|\mathcal{W}_h \odot (\mathbf{D}_h \mathcal{U})\|_1 + \|\mathcal{W}_v \odot (\mathbf{D}_v \mathcal{U})\|_1] \\ & + \lambda_2 \|\mathbf{D}_s \mathbf{V}\|_F^2 + \lambda_3 \sum_{i=1}^P \|P_i \mathcal{U}\|_{w,*} + \lambda_4 \|\mathcal{W}_s \odot \mathcal{S}\|_1 \end{aligned} \right\} \quad (13)$$

where $\lambda_i (i = 1, 2, 3, 4) > 0$ is a regularization parameter, which is used to balance the weights of the items in the objective function. The proposed model contains five components. The first item $\frac{1}{2} \|\mathcal{Y} - \mathcal{U}_{\times 3} \mathbf{V} - \mathcal{S}\|_F^2$ uses the l_F -norm to characterize the data fidelity of the normal distribution of \mathcal{N} , and the $\mathcal{U}_{\times 3} \mathbf{V}$ contained in it is used to characterize the LR of \mathcal{X} . The second item $\|\mathcal{W}_h \odot (\mathbf{D}_h \mathcal{U})\|_1 + \|\mathcal{W}_v \odot (\mathbf{D}_v \mathcal{U})\|_1$ adds adaptive spatial bidirectional TV to \mathcal{U} and uses l_1 -norm to characterize the spatial LS of \mathcal{X} . \mathcal{W}_h and \mathcal{W}_v are the weight tensors [49] for promoting LS. The third term $\|\mathbf{D}_s \mathbf{V}\|_F^2$ adds TV to \mathbf{V} and uses l_F -norm to characterize the spectral LS of \mathcal{X} . The fourth term, $\sum_{i=1}^P \|P_i \mathcal{U}\|_{w,*}$, adds a nonlocal low-rank constraint to \mathcal{U} to characterize the NSS of \mathcal{X} , and puts it outside the main loop of the algorithm to solve the initial value of \mathcal{U} in the main loop. The fifth item $\|\mathcal{W}_s \odot \mathcal{S}\|_1$ characterizes the sparsity of \mathcal{S} with l_1 -norm combined with weight strategy, which is used to remove impulse noise, deadlines, and stripes. \mathcal{W}_s is the weight tensor used to promote the sparsity of \mathcal{S} . The flowchart of the proposed DFTVNLR method is shown in Figure 1.

The key of the proposed DFTVNLR model to improve the denoising performance is that it fully characterizes all the priors of the HSI and mixed noise. However, considering the fact that fully exploiting prior knowledge will increase the computational cost, the proposed model can effectively control the computational cost by using small-sized double factors rather than large-sized HSIs as constraint objects, and placing the time-consuming NLR model outside the main loop to indirectly characterize the NSS prior. Hence, the main advantages of the proposed DFTVNLR model are as follows.

1. Exploration of the prior knowledge of clean HSIs and mixed noise more fully. Compared to HSI mixed noise removal methods [2,29–31,41,46], which only consider partial priors, the proposed DFTVNLR comprehensively and fully considers LR, the spatial–spectral LS, the NSS of \mathcal{X} , data fidelity of \mathcal{N} , and the sparsity of \mathcal{S} .

2. A better balance of the contradiction between the comprehensive representation of priors and the high computational cost. Compared to directly applying regularization to the large-sized \mathcal{X} [29,41] or only to \mathcal{U} [2], we applied regularization to small-sized $\mathcal{U} - \mathcal{V}$, which not only fully explores prior knowledge but also reduces the cost. Furthermore, we characterized NSS in the assisted loop instead of the main loop [30,41], which is another important measure to effectively control the cost.

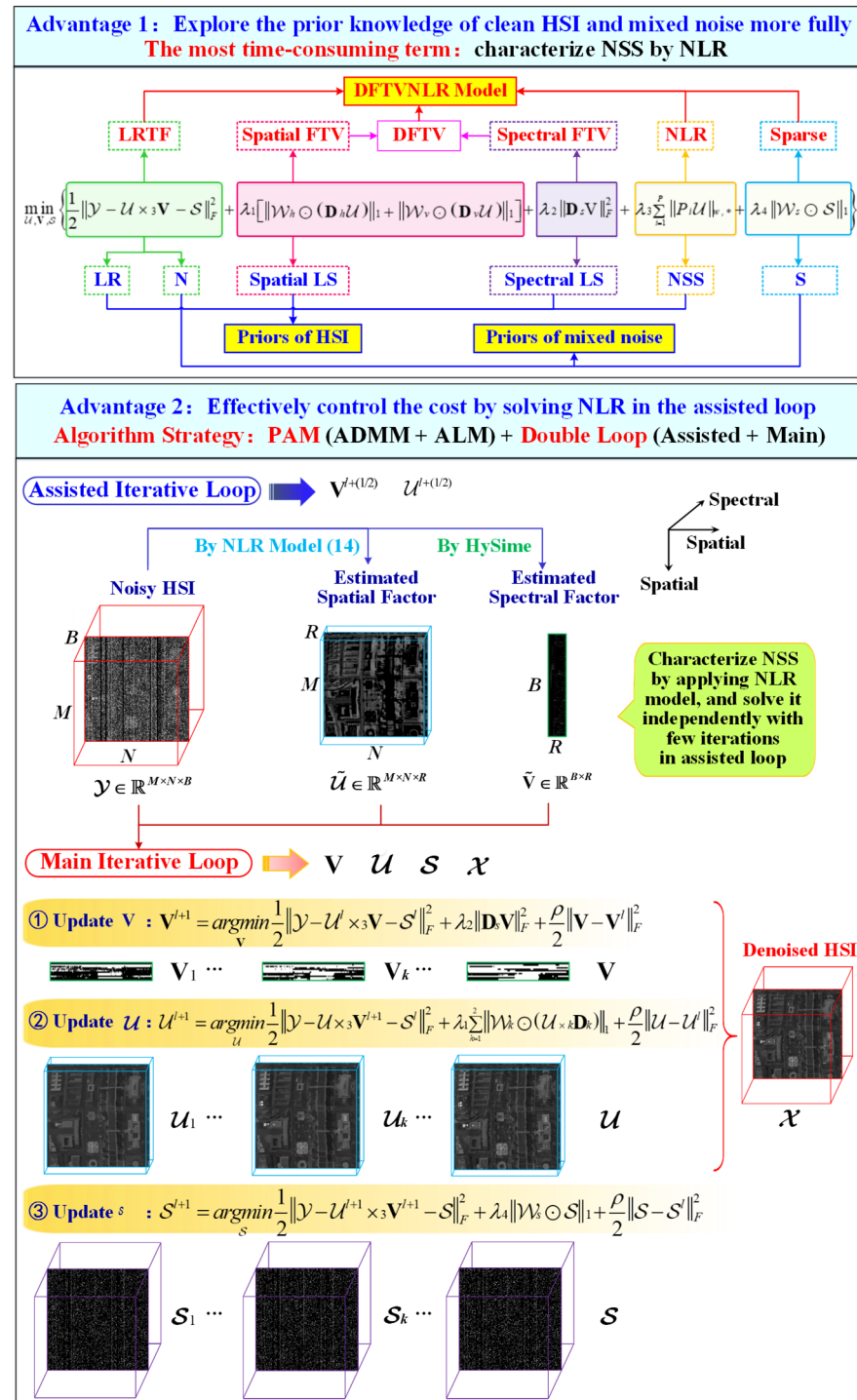


Figure 1. The flowchart of the proposed DFTVNLR method. It includes two main segments. First, in the assisted loop, the NLR model and Hysime algorithm are used to estimate the initial values of \mathcal{U} and \mathcal{V} . Second, in the main loop, the estimated $\mathcal{U} - \mathcal{V}$ initial value and other prior knowledge are used iteratively to obtain the denoised HSI.

3.3. Solving the Optimization Algorithm

3.3.1. Estimate the Initial Values of \mathcal{U} and \mathbf{V}

In this article, HySime algorithm was used to learn the $\mathbf{V}^{l+(1/2)}$ from \mathcal{Y} and take it as the initial value of \mathbf{V} in the main loop of the algorithm. In addition, in order to reduce the computational cost, the $\sum_{i=1}^P \|P_i \mathcal{U}\|_{w,*}$ in model (13) was placed outside the main loop, and $\mathcal{U}^{l+(1/2)}$ was solved independently with few iterations, which was taken as the rough initial value of \mathcal{U} in the subsequent main loop. We referred to the solution strategy of \mathcal{U} in reference [41], and the process of estimating the initial value of \mathcal{U} is as follows:

$$\mathcal{U}^{l+(1/2)} = \underset{\mathcal{U}}{\operatorname{argmin}} \lambda_3 \sum_{i=1}^P \|P_i \mathcal{U}\|_{w,*} + \|\mathcal{U} - \mathcal{U}^l\|_F^2 \quad (14)$$

Furthermore, we used the famous ADMM [50] to solve model (13) by alternately updating the following equation:

$$\begin{cases} \text{Step1: } \mathbf{V}^{l+1} = \underset{\mathbf{V}}{\operatorname{argmin}} f(\mathbf{V}, \mathcal{U}^l, \mathcal{S}^l) + \frac{\rho}{2} \|\mathbf{V} - \mathbf{V}^l\|_F^2 \\ \text{Step2: } \mathcal{U}^{l+1} = \underset{\mathcal{U}}{\operatorname{argmin}} f(\mathbf{V}^{l+1}, \mathcal{U}, \mathcal{S}^l) + \frac{\rho}{2} \|\mathcal{U} - \mathcal{U}^l\|_F^2 \\ \text{Step3: } \mathcal{S}^{l+1} = \underset{\mathcal{S}}{\operatorname{argmin}} f(\mathbf{V}^{l+1}, \mathcal{U}^{l+1}, \mathcal{S}) + \frac{\rho}{2} \|\mathcal{S} - \mathcal{S}^l\|_F^2 \end{cases} \quad (15)$$

where $f(\mathbf{V}, \mathcal{U}, \mathcal{S})$ is the objective function of model (13) and $\rho > 0$ is the proximity parameter.

3.3.2. Update \mathbf{V}

The solution equation of the \mathbf{V} -subproblem is detailed as follows:

$$\mathbf{V}^{l+1} = \underset{\mathbf{V}}{\operatorname{argmin}} \frac{1}{2} \|\mathcal{Y} - \mathcal{U}^l \times_3 \mathbf{V} - \mathcal{S}^l\|_F^2 + \lambda_2 \|\mathbf{D}_s \mathbf{V}\|_F^2 + \frac{\rho}{2} \|\mathbf{V} - \mathbf{V}^l\|_F^2 \quad (16)$$

The above equation can be transformed into the following Sylvester matrix equation and then be solved:

$$2 \lambda_2 \mathbf{D}_s^T \mathbf{D}_s \mathbf{V} + \mathbf{V} \mathbf{U}_{(3)}^l (\mathbf{U}_{(3)}^l)^T + \rho \mathbf{V} = (\mathbf{Y}_{(3)} - \mathbf{S}_{(3)}^l) (\mathbf{U}_{(3)}^l)^T + \rho \mathbf{V}^l \quad (17)$$

For a fast solution of the Sylvester matrix equation, please refer to Theorem 1 in [31]. The typical Sylvester matrix equation satisfies the form $\mathbf{A}\mathbf{X} + \mathbf{X}\mathbf{B} = \mathbf{Y}$, where $\mathbf{A} \in \mathbb{R}^{M \times M}$, $\mathbf{B} \in \mathbb{R}^{N \times N}$, $\mathbf{X} \in \mathbb{R}^{M \times N}$, and $\mathbf{Y} \in \mathbb{R}^{M \times N}$. In the above \mathbf{V} subproblem, the four terms corresponding to \mathbf{X} , \mathbf{A} , \mathbf{B} , and \mathbf{Y} in the typical Sylvester matrix equation are: $\mathbf{X} = \mathbf{V}$, $\mathbf{A} = 2\lambda_2 \mathbf{D}_s^T \mathbf{D}_s$, $\mathbf{B} = \mathbf{U}_{(3)}^l (\mathbf{U}_{(3)}^l)^T$, and $\mathbf{Y} = (\mathbf{Y}_{(3)} - \mathbf{S}_{(3)}^l) (\mathbf{U}_{(3)}^l)^T + \rho \mathbf{V}^l$, respectively. The diagonalization process of $\mathbf{D}_s^T \mathbf{D}_s$ and $\mathbf{U}_{(3)}^l (\mathbf{U}_{(3)}^l)^T$ are as follows:

$$\mathbf{D}_s^T \mathbf{D}_s = \mathbf{F}_1^T \boldsymbol{\Psi}_1 \mathbf{F}_1, \quad \mathbf{U}_{(3)}^l (\mathbf{U}_{(3)}^l)^T = \mathbf{U}_1 \boldsymbol{\Sigma}_1 \mathbf{U}_1^T. \quad (18)$$

where \mathbf{F}_1 represents the 1-D discrete fast Fourier transformation (FFT). Therefore, the \mathbf{V} -subproblem can be solved as follows:

$$\mathbf{V}^{l+1} = \mathbf{F}_1^T ((1/\mathbf{T}_1) \odot (\mathbf{F}_1 \mathbf{G} \mathbf{U}_1)) \mathbf{U}_1^T \quad (19)$$

$$\begin{cases} \mathbf{G} = (\mathbf{Y}_{(3)} - \mathbf{S}_{(3)}^l) (\mathbf{U}_{(3)}^l)^T + \rho \mathbf{V}^l \\ \mathbf{T}_1 = 2\lambda [\text{diag}(\boldsymbol{\Psi}_1), \dots, \text{diag}(\boldsymbol{\Psi}_1)] + \rho \cdot \text{ones}(\mathbf{B}, \mathbf{R}) + [\text{diag}(\boldsymbol{\Sigma}_1), \dots, \text{diag}(\boldsymbol{\Sigma}_1)]^T \end{cases} \quad (20)$$

where $\text{diag}(\boldsymbol{\Lambda})$ is a column vector, whose elements are diagonal elements of $\boldsymbol{\Lambda}$, and $\text{ones}(\mathbf{B}, \mathbf{R})$ is an $\mathbf{B} \times \mathbf{R}$ matrix, whose elements are all 1.

In particular, the initial iteration value of $\mathbf{V}^{l+(1/2)}$ was estimated in advance by the HySime algorithm in the assisted loop of Section 3.3.1, which helps the algorithm to accelerate the convergence.

3.3.3. Update \mathcal{U}

The solution equation of the \mathcal{U} -subproblem is detailed as follows:

$$\mathcal{U}^{l+1} = \underset{\mathcal{U}}{\text{argmin}} \frac{1}{2} \left\| \mathcal{Y} - \mathcal{U}_{\times 3} \mathbf{V}^{l+1} - \mathcal{S}^l \right\|_F^2 + \lambda_1 \sum_{k=1}^2 \left\| \mathcal{W}_k \odot (\mathcal{U}_{\times k} \mathbf{D}_k) \right\|_1 + \frac{\rho}{2} \left\| \mathcal{U} - \mathcal{U}^l \right\|_F^2 \quad (21)$$

By introducing the auxiliary variable $\mathcal{A}_k (k = 1, 2)$, (21) is rewritten as:

$$\begin{aligned} \mathcal{U}^{l+1} &= \underset{\mathcal{U}}{\text{argmin}} \frac{1}{2} \left\| \mathcal{Y} - \mathcal{U}_{\times 3} \mathbf{V}^{l+1} - \mathcal{S}^l \right\|_F^2 + \lambda_1 \sum_{k=1}^2 \left\| \mathcal{W}_k \odot \mathcal{A}_k \right\|_1 + \frac{\rho}{2} \left\| \mathcal{U} - \mathcal{U}^l \right\|_F^2 \\ \text{s.t. } \mathcal{A}_k &= \mathcal{U}_{\times k} \mathbf{D}_k, \quad k = 1, 2. \end{aligned} \quad (22)$$

We used ALM [51] to solve the above (22), and its augmented Lagrangian function is as follows:

$$\begin{aligned} L_\mu(\mathcal{U}, \mathcal{A}_k, \mathcal{M}_k) &= \frac{1}{2} \left\| \mathcal{Y} - \mathcal{U}_{\times 3} \mathbf{V}^{l+1} - \mathcal{S}^l \right\|_F^2 \\ &\quad + \sum_{k=1}^2 \left\{ \lambda_1 \left\| \mathcal{W}_k \odot \mathcal{A}_k \right\|_1 + \frac{\mu}{2} \left\| \mathcal{U}_{\times k} \mathbf{D}_k - \mathcal{A}_k + \frac{\mathcal{M}_k}{\mu} \right\|_F^2 \right\} + \frac{\rho}{2} \left\| \mathcal{U} - \mathcal{U}^l \right\|_F^2 \end{aligned} \quad (23)$$

where $\mathcal{M}_k (k = 1, 2)$ is a Lagrange multiplier and $\mu > 0$ is the penalty parameter. Under the framework of ADMM, we can alternately optimize (23) to solve \mathcal{U} , \mathcal{A}_k , and \mathcal{M}_k .

(1) Update $\mathcal{U}^{l+1,p+1}$

The subproblem for updating \mathcal{U} can be expressed as follows:

$$\mathcal{U}^{l+1,p+1} = \underset{\mathcal{U}}{\text{argmin}} \frac{1}{2} \left\| \mathcal{Y} - \mathcal{U}_{\times 3} \mathbf{V}^{l+1} - \mathcal{S}^l \right\|_F^2 + \frac{\rho}{2} \left\| \mathcal{U} - \mathcal{U}^l \right\|_F^2 + \frac{\mu}{2} \sum_{k=1}^2 \left\| \mathcal{U}_{\times k} \mathbf{D}_k - \mathcal{A}_k^p + \frac{\mathcal{M}_k^p}{\mu} \right\|_F^2 \quad (24)$$

Deriving (24) with respect to \mathcal{U} , we can obtain the following equation:

$$\begin{cases} \mu \sum_{k=1}^2 \mathcal{U}_{\times k} (\mathbf{D}_k^T \mathbf{D}_k) + \mathcal{U}_{\times 3} \left[(\mathbf{V}^{l+1})^T \mathbf{V}^{l+1} \right] + \rho \mathcal{U} = \mathcal{K} \\ \mathcal{K} = (\mathcal{Y} - \mathcal{S}^l)_{\times 3} (\mathbf{V}^{l+1})^T + \rho \mathcal{U}^l + \mu \sum_{k=1}^2 \left(\mathcal{A}_k^p - \frac{\mathcal{M}_k^p}{\mu} \right)_{\times k} \mathbf{D}_k^T \end{cases} \quad (25)$$

The solution of (25) can be obtained by the following Sylvester matrix Equation:

$$\mu \left[(\mathbf{I}_N \otimes \mathbf{D}_1^T \mathbf{D}_1) + (\mathbf{D}_2^T \mathbf{D}_2 \otimes \mathbf{I}_M) \right] \mathbf{U}_{(3)}^T + \mathbf{U}_{(3)}^T \left[(\mathbf{V}^{l+1})^T \mathbf{V}^{l+1} \right] + \rho \mathbf{U}_{(3)}^T = \mathbf{K}_{(3)}^T \quad (26)$$

The diagonalization process is as follows:

$$\mu [(\mathbf{I}_N \otimes \mathbf{D}_1^T \mathbf{D}_1) + (\mathbf{D}_2^T \mathbf{D}_2 \otimes \mathbf{I}_M)] = \mathbf{F}_2^T \boldsymbol{\Psi}_2 \mathbf{F}_2, \quad (\mathbf{V}^{l+1})^T \mathbf{V}^{l+1} = \mathbf{U}_2 \boldsymbol{\Sigma}_2 \mathbf{U}_2^T \quad (27)$$

where \mathbf{F}_2 represents 2-D discrete FFT.

Therefore, the $\mathcal{U}^{l+1,p+1}$ -subproblem can be solved as follows:

$$\mathbf{U}_{(3)} = \left[\mathbf{F}_2^T \left((1/\mathbf{T}_2) \odot \left(\mathbf{F}_2 \mathbf{D}_{(3)}^T \mathbf{U}_2 \right) \right) \mathbf{U}_2^T \right]^T, \quad \mathcal{U}^{l+1,p+1} = \text{Fold}_{(3)} \left[\mathbf{U}_{(3)} \right] \quad (28)$$

where

$$\mathbf{T}_2 = [\text{diag}(\boldsymbol{\Psi}_2), \text{diag}(\boldsymbol{\Psi}_2), \dots, \text{diag}(\boldsymbol{\Psi}_2)] + \rho \cdot \text{ones}(\text{MN}, \text{R}) \\ [\text{diag}(\boldsymbol{\Sigma}_2), \text{diag}(\boldsymbol{\Sigma}_2), \dots, \text{diag}(\boldsymbol{\Sigma}_2)]^T \quad (29)$$

In particular, the initial iteration value of $\mathcal{U}^{l+(1/2)}$ was estimated in advance by the NLR model in the assisted loop of Section 3.3.1, which helps the algorithm to accelerate the convergence.

(2) Update \mathcal{A}_k^{p+1}

The subproblem for updating \mathcal{A}_k can be expressed as follows:

$$\mathcal{A}_k^{p+1} = \underset{\mathcal{A}_k}{\text{argmin}} \lambda_1 \|\mathcal{W}_k \odot \mathcal{A}_k\|_1 + \frac{\mu}{2} \left\| \mathcal{U}^{l+1,p+1} \times_k \mathbf{D}_k - \mathcal{A}_k + \frac{\mathcal{M}_k^p}{\mu} \right\|_F^2 \quad (k=1, 2) \quad (30)$$

The closed-form solution of \mathcal{A} can be obtained by using the known soft-thresholding operator [52] as follows:

$$\mathcal{A}_k^{p+1}(i, j, :) = \mathcal{S}_{|\mathbf{W}_k(i,j)| \cdot \frac{\lambda_1}{\mu}} \left(\mathcal{U}^{l+1,p+1} \times_k \mathbf{D}_k + \frac{\mathcal{M}_k^p}{\mu} \right) \quad (k=1, 2) \quad (31)$$

where $\mathcal{S}_\lambda(\cdot)$ represents the soft-thresholding operator with parameter λ and $\mathcal{S}_\lambda(x) = \text{sgn}(x) \max(|x| - \lambda, 0)$. Additionally, the selection method and calculation process of $\mathbf{W}_k(i, j) (k=1, 2)$ are the same as those in Section 3.2.2 of this article.

(3) Update \mathcal{M}_k^{p+1}

The multipliers \mathcal{M}_k are updated by the following Equation:

$$\mathcal{M}_k^{p+1} = \mathcal{M}_k^p + \mu \left(\mathcal{U}^{l+1,p+1} \times_k \mathbf{D}_k - \mathcal{A}_k^{p+1} \right) \quad (k=1, 2) \quad (32)$$

3.3.4. Update \mathcal{S}

The solution equation of the \mathcal{S} -subproblem is detailed as follows:

$$\mathcal{S}^{l+1} = \underset{\mathcal{S}}{\text{argmin}} \frac{1}{2} \left\| \mathcal{Y} - \mathcal{U}^{l+1} \times_3 \mathbf{V}^{l+1} - \mathcal{S} \right\|_F^2 + \lambda_4 \|\mathcal{W}_s \odot \mathcal{S}\|_1 + \frac{\rho}{2} \left\| \mathcal{S} - \mathcal{S}^l \right\|_F^2 \quad (33)$$

The solution of \mathcal{S} can be obtained by the soft-thresholding operator as follows:

$$\mathcal{S}^{l+1} = \mathcal{S}_{\mathcal{W}_s \odot \frac{\lambda_4}{1+\rho}} \left(\frac{\mathcal{Y} - \mathcal{U}^{l+1} \times_3 \mathbf{V}^{l+1} + \rho \mathcal{S}^l}{1 + \rho} \right) \quad (34)$$

where let $\hat{\mathcal{S}} = (\mathcal{Y} - \mathcal{U}^{l+1} \times_3 \mathbf{V}^{l+1} + \rho \mathcal{S}^l) / (1 + \rho)$; then, $\mathcal{W}_s = 1 / (|\hat{\mathcal{S}}| + \varepsilon)$. ε is a small constant for avoiding the appearance of singularities.

To sum up, we describe the pseudocode of the developed algorithm for solving the DFTVNLR-based HSI denoising model in Algorithm 1.

Algorithm 1. The pseudocode of the algorithm for solving the DFTVNLN model

Input: The observed noisy HIS $\mathcal{Y} \in \mathbb{R}^{M \times N \times B}$, rank R (the dimension of subspace); the regularization parameters $\lambda_1, \lambda_2, \lambda_3$, and λ_4 ; the penalty parameter of ALM function $\mu = 15,000$; and the proximal error parameter $\rho = 0.1$.

Initialization: In the assisted loop iteration: $n = 0, n_{\max} = 10, \mathcal{U}^0$ and \mathbf{V}^0 are estimated by SVD (under the condition of rank R), $\mathcal{S}^0 = 0, \mathcal{A}_1^0 = 0, \mathcal{A}_2^0 = 0, \mathcal{M}_1^0 = 0, \mathcal{M}_2^0 = 0$, PatSize = 5, PatNum = 200, SearchWin = 55, Step = 5. In the main loop iteration: $l = 0, l_{\max} = 50$, the alternate iteration of AL function: $p = 0, p_{\max} = 10$, and the maximum tolerance of iterative results $\varepsilon = 10^{-4}$.

Assisted loop: Pre-estimation of the $\mathcal{U}-\mathbf{V}$ initial value.

- 1: $\mathbf{V}^{l+(1/2)}$ is obtained from \mathcal{Y} by the HySime algorithm, and it is taken as the initial value \mathbf{V}^0 in the main loop.
- 2: The NLR regularization term $\sum_{i=1}^p \|P_i \mathcal{U}\|_{w,*}$ is used to solve $\mathcal{U}^{l+(1/2)}$ independently, and it is taken as the initial value \mathcal{U}^0 in the main loop.
- 3: **while** $n < n_{\max}$ **do**
- 4: Update $\mathcal{U}^{l+(1/2)}$ by Equation (14).
- 5: $n = n + 1$.
- 6: **end**
- 7: $\mathbf{V}^0 = \mathbf{V}^{l+(1/2)}, \mathcal{U}^0 = \mathcal{U}^{l+(1/2)}$.

Main loop: Solution of the $\mathbf{V}-\mathcal{U}-\mathcal{S}$ subproblem.

- 1: **while** not converged and $l < l_{\max}$ **do**
- 2: Update \mathbf{V}^{l+1} by Equations (19) and (20).
- 3: **while** $p < p_{\max}$ **do**
- 4: Update \mathcal{U}^{l+1} by Equations (28) and (29).
- 5: Update the auxiliary variable \mathcal{A}_k^{p+1} ($k = 1, 2$) by Equation (31).
- 6: Update the Lagrange multiplier \mathcal{M}_k^{p+1} ($k = 1, 2$) by Equation (32).
- 7: $p = p + 1$.
- 8: Check the convergence condition: $\|\mathcal{U}^{l+1, p+1} - \mathcal{U}^{l+1, p}\|_F / \|\mathcal{U}^{l+1, p}\|_F \leq \varepsilon$.
- 9: **end while**
- 10: $\mathcal{U}^{l+1} = \mathcal{U}^{l+1, p+1}$.
- 11: Update \mathcal{S}^{l+1} by Equation (34).
- 12: $l = l + 1$;
- 13: $\mathcal{X}^{l+1} = \mathcal{U}^{l+1} \times_3 \mathbf{V}^{l+1}$.
- 14: Check the convergence condition: $\|\mathcal{X}^{l+1} - \mathcal{X}^l\|_F / \|\mathcal{X}^l\|_F \leq \varepsilon$.
- 15: **end while**

Output: The denoised HSI $\mathcal{X}^{l+1} \in \mathbb{R}^{M \times N \times B}$.

3.3.5. Computational Complexity Analysis

According to Algorithm 1, the computational cost of the proposed DFTVNLN is mainly reflected in operations such as FFT, SVD, similar block search (SBS), soft-thresholding operation (SoftThreshold), and the matrix or tensor product. Based on the relevant conclusions in reference [2,31], we summarize the computational complexity of solving operators involved in common prior terms in Table 1.

In this article, the algorithm cost calculation method mentioned in reference [2] was adopted, that is, the cheap matrix or tensor product operation cost was ignored, and only operators with a high computational cost, such as FFT, SVD, SBS, and SoftThreshold, were considered. First, updating $\mathcal{U}^{l+(1/2)}$ via (14) mainly involves SBS and SVD, which leads to the $\mathcal{O}\left[M^2 N^2 R (\text{PatSize})^2 / (\text{StepSize})^4 + (\text{GroupNum}) R^2 (\text{PatNum}) (\text{PatSize})^2\right]$ cost approximately. Second, updating \mathbf{V} via (19) and (20) mainly involves SVD and 1-D FFT, which leads to the $\mathcal{O}(BR^2 + BR \log(B))$ cost approximately. Third, updating \mathcal{U} via (28) and (29) needs about $\mathcal{O}(BR^2 + MNR \log(MN))$ of computational cost, since it mainly involves SVD and 2-D FFT. Finally, updating \mathcal{A}_1 and \mathcal{A}_2 via (31) and updating \mathcal{S} via (34) mainly involve SoftThreshold, and the computational cost is about $\mathcal{O}(MNB)$. In addition, when updating the initial value of \mathbf{V} by HySime and updating \mathcal{M}_1 and \mathcal{M}_2 by (32), only the product

operation between the matrices or tensors is involved, and the computational cost can be ignored. To sum up, the cumulative computational cost of assisted and main loop iteration can be roughly estimated by the following Equation (35).

$$\mathcal{O}\left\{n\left[M^2N^2R(PatSize)^2/(StepSize)^4 + (GroupNum)R^2(PatNum)(PatSize)^2\right] + l[BR^2 + BR\log(B) + MNR\log(MN) + pMNB]\right\} \quad (35)$$

4. Experimental Results and Discussion

In this section, to demonstrate the performance of the proposed DFTVNLR in terms of visual quality and quantitative evaluation, we conducted a series of experiments on both the simulated and real HSI data. Additionally, we adopted nine state-of-art model-based HSI denoising methods for comparison, which are most relevant to our method, i.e., LRTFDFR [31], WNLRATV [41], RCTV [2], SNLRSF [30], FastHyMix [46], L0L1HTV [29], LRTDTV [9], LRMR [53], and FGSLR [54]. Their implementation codes can be directly obtained from the authors' websites. These methods can be divided into four categories. Specifically, the methods based on noise modeling include WNLRATV and FastHyMix. The methods based on factor or representation coefficient include LRTFDFR, WNLRATV, RCTV, and SNLRSF. The methods considering NSS prior include WNLRATV and SNLRSF. The methods based on TV include LRTFDFR, WNLRATV, RCTV, and L0L1HTV.

The differences between our method and the selected comparison method are summarized in Table 2. All relevant parameters involved in these competing methods are set by the default parameters presented by the author. For our DFTVNLR solver, we would like to present a detailed discussion of parameter setting in Section 4.3. Before the simulated and real experiments, the gray values of the HSI are normalized into the interval [0, 1] band-by-band. All experiments were implemented on a PC with Windows 10 in the environment of an Intel Core i7-8700CPU@3.20GHz processor and 32 GB RAM.

Table 2. The detailed relations and distinctions between the proposed method and the compared methods.

Method	Spectral Global Low-Rankness (LR)	Spatial-Spectral Local Smoothness (LS)	Spatial Nonlocal Self-Similarity (NSS)	Gaussian Noise (G)	Sparse Noise (S)
LRTFDFR	$\mathcal{X} = \mathcal{U}_{\times 3} \mathbf{V}$	$\tau \sum_{k=1}^2 \ \mathcal{W}_k \odot (\mathcal{U}_{\times k} \mathbf{D}_k)\ _{2,1} + \lambda \ \mathbf{D}_3 \mathbf{A}\ _F^2$	—	$\frac{1}{2} \ \mathcal{Y} - \mathcal{U}_{\times 3} \mathbf{V} - \mathcal{S}\ _F^2$	$\mu \ \mathcal{W}_s \odot \mathcal{S}\ _1$
WNLRATV	$\mathcal{X} = \mathcal{U}_{\times 3} \mathbf{V}$	$\lambda_1 \ \mathcal{W} \odot (\mathbf{D} \mathcal{X})\ _1$	$\lambda_2 \sum_{i=1}^P \ P_i \mathcal{U}\ _{w,*}$	$\ \mathcal{W} \odot (\mathcal{Y} - \mathcal{X})\ _F^2$	—
RCTV	$\mathbf{X} = \mathbf{U} \mathbf{V}^T$	$\sum_{k=1}^2 \tau_k \ \nabla_k(\mathbf{U})\ _1$	—	$\beta \ \mathcal{N}\ _F^2$	$\lambda \ \mathcal{S}\ _1$
SNLRSF	$\mathbf{X} = \mathbf{U} \mathbf{V}^T$	—	$\lambda_1 \sum_i (\frac{1}{\delta_i^2} \ \mathbf{R}_i \mathbf{U} - \mathbf{L}_i\ _F^2 + \mathbf{K}(\mathbf{L}_i))$	$\frac{1}{2} \ \mathcal{Y} - \mathbf{U} \mathbf{V}^T - \mathcal{S}\ _F^2$	$\lambda_2 \ \mathcal{S}\ _1$
L0L1HTV	—	$\lambda_1 (\ \mathbf{D}_h \mathcal{X} \mathbf{D}_z\ _1 + \ \mathbf{D}_v \mathcal{X} \mathbf{D}_z\ _1) + \mathbf{I}_{\ \mathbf{B}\ _{1,0}^0}(\mathbf{B} \mathbf{D} \mathcal{X})$	—	$\ \mathcal{Y} - \mathcal{X} - \mathcal{S}\ _F^2$	$\lambda_2 \ \mathcal{S}\ _1$
LRTDTV	$\mathcal{X} = \mathcal{C}_{\times 1} \mathbf{U}_{1 \times 2} \mathbf{U}_{2 \times 3} \mathbf{U}_3$	$\tau \ \mathcal{X}\ _{\text{SSTV}}$	—	$\beta \ \mathcal{N}\ _F^2$	$\lambda \ \mathcal{S}\ _1$
LRMR	Minimize the rank of matrix	—	—	$\ \mathbf{Y} - \mathbf{X} - \mathbf{S}_{i-1}\ _F^2$ When $\text{rank}(\mathbf{X}) \leq r$	$\ \mathbf{Y} - \mathbf{X}_t - \mathbf{S}\ _F^2$ When $\text{card}(\mathbf{S}) \leq r$
FGSLR	$\mathcal{X} = \mathcal{U}_{\times 3} \mathbf{V} \ \mathbf{V}\ _{2,1} + \ \mathcal{U}\ _{2,1}$	—	—	$\frac{\beta}{2} \ \mathcal{Y} - \mathcal{U}_{\times 3} \mathbf{V} - \mathcal{S}\ _F^2$	$\tau \ \mathcal{S}\ _1$
DFTVNLR	$\mathcal{X} = \mathcal{U}_{\times 3} \mathbf{V}$	$\lambda_1 [\ \mathcal{W}_h \odot (\mathbf{D}_h \mathcal{U})\ _1 + \ \mathcal{W}_v \odot (\mathbf{D}_v \mathcal{U})\ _1] + \lambda_2 \ \mathbf{D}_s \mathbf{V}\ _F^2$	$\lambda_3 \sum_{i=1}^P \ P_i \mathcal{U}\ _{w,*}$	$\frac{1}{2} \ \mathcal{Y} - \mathcal{U}_{\times 3} \mathbf{V} - \mathcal{S}\ _F^2$	$\lambda_4 \ \mathcal{W}_s \odot \mathcal{S}\ _1$

4.1. Simulated Data Experiments

We conducted the simulated experiment on two public HSIs in this section: one was the simulated Indian Pines dataset (<http://www.ehu.eus/ccwintco/index.php?title=>

[Hyperspectral_Remote_Sensing_Scenes](http://engineering.purdue.edu/biehl/MultiSpec/hyperspectral.html), accessed on 1 March 2023), with the size of 145 (rows) \times 145 (columns) \times 224 (bands). which has the obvious characteristics of LS and large edge gradient, and is also used in the compared methods LRTFDFR, WNL-RATV, L0L1HTV, and LRTDTV; the other was the Washington DC (WDC) Mall dataset (<http://engineering.purdue.edu/biehl/MultiSpec/hyperspectral.html>, accessed on 5 March 2023), with the size of 256 (rows) \times 256 (columns) \times 191 (bands), which is rich in local texture details, and is also used in the compared methods LRTFDFR, WNL-RATV, RCTV, SNLRSE, FastHyMix, L0L1HTV, LRMR, and FGSLR.

Because the noise in real HSIs usually shows a mixture of several types of noise, we considered the following five additional noise mixing cases, which is basically consistent with [31]. It should be noted the salt and pepper noise belongs to a typical and common impulse noise.

Case 1 (Gaussian Noise): Add Gaussian noise with the average value of 0 and noise standard deviation randomly sampled from $[0.1, 0.2]$ to all bands.

Case 2 (Gaussian Noise + Salt and Pepper Noise): First, Gaussian noise is added as in case 1. Then, add the salt and pepper noise with the noise proportion randomly sampled from $[0.1, 0.2]$ to all bands.

Case 3 (Gaussian Noise + Salt and Pepper Noise + Deadlines): First, add Gaussian noise and salt and pepper noise as in case 2. Then, add deadlines to the randomly selected 20% bands. The width of the deadlines is sampled from the set $[1, 2, 3]$. The number of deadlines in each band is sampled from the set $[6, 7, 8, 9, 10]$.

Case 4 (Gaussian Noise + Salt and Pepper Noise + Stripes): First, add Gaussian noise and salt and pepper noise as in case 2. Furthermore, add stripes to the randomly selected 40% bands. The number of stripes in each band is sampled from the set $[6, 7, 8, \dots, 15]$.

Case 5 (Gaussian Noise + Salt and Pepper Noise + Stripes + Deadlines): First, add Gaussian noise and salt and pepper noise as in case 2. Then, 20% of all bands are randomly selected to add deadlines as in case 3. Finally, 40% of all bands are randomly selected to add stripes as in case 4.

In the simulated experiment, six frequently used evaluation metrics, i.e., the mean of peak signal-to-noise ratio (MPSNR), the mean of structural similarity (MSSIM), the mean of feature similarity (MFSIM), the mean of spectral angle mapping (MSAM), the Erreur Relative Globale Adimensionnelle de Synthèse (ERGAS), and the running time of the algorithm, were employed to evaluate the overall quality of the denoised results quantitatively. Among them, MPSNR, MSSIM, and MFSIM are the mean of PSNR [53], SSIM [55], and FSIM [56] over the bands, respectively. In addition, the better denoising effect and algorithm performance are indicated by the larger MPSNR, MSSIM, and MFSIM values and the smaller MSAM, ERGAS, and Time values.

4.1.1. Visual Quality Comparison

To visually compare the denoising results, Figures 2–5, respectively, show some pseudocolor synthetic bands of the Indian Pines dataset and the WDC Mall dataset denoised under noise cases 1 and 5 and cases 4 and 5. For a better visual comparison, the common region of each figure is marked by a red box and enlarged by 2.5 times.

From Figures 2–5, several observations can be easily made. First, all methods can remove mixed noise for the two simulated datasets to some extent. Second, the proposed DFTVNLR can both effectively remove mixed noise and preserve the global structure and local details, as shown in the enlarged subregions of Figures 2l, 3l, 4l and 5l, indicating that our method is reasonable, effective, and robust. Third, the denoising performances of RCTV, FastHyMix, L0L1HTV, LRTDTV, LRMR, and FGSLR are not as good as that of DFTVNLR, as shown in Figure 3e,g–l, Figure 4e,g–l and Figure 5e,g–l, which show that the denoising ability of our proposed method is stronger, thanks to the adoption of LRTE, DFTV, and NLR. Finally, the proposed DFTVNLR is better than LRTFDFR, WNL-RATV, and SNLRSE, as shown in Figure 3c,d,f,l, Figure 4c,d,f,l and Figure 5c,d,f,l, which show that the strategies, exploration of prior knowledge more fully, adoption of small-sized factor

regularization, and prediction of the initial values of double factors by NLR and HySime can improve the denoising performance. Additionally, RCTV, LOL1HTV, and FGSLR are not universal and robust, and their denoising effects are highly dependent on the image types.

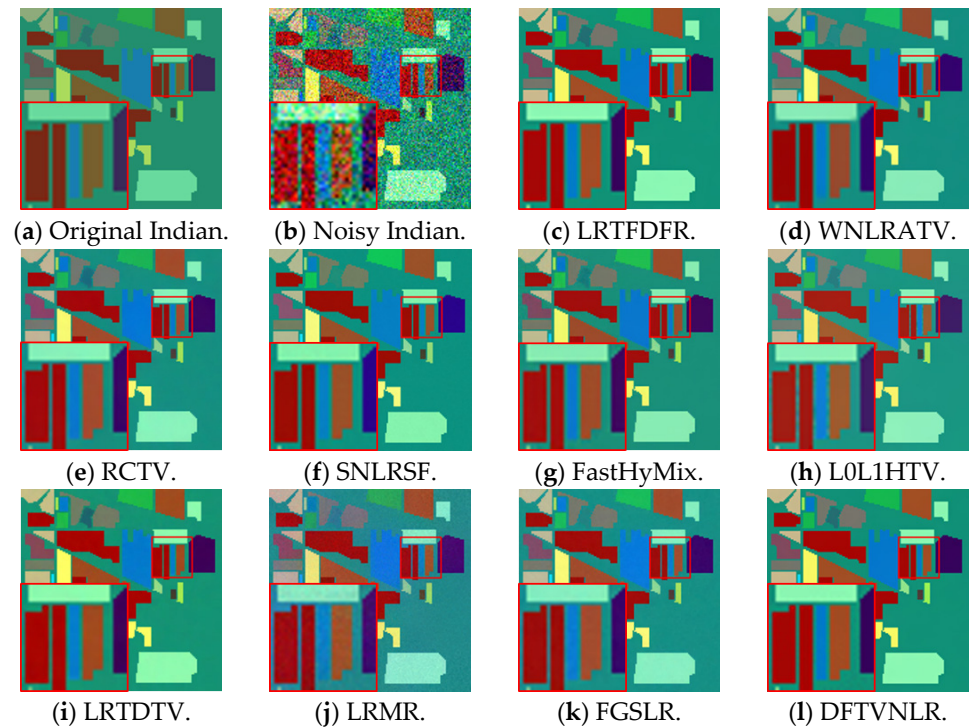


Figure 2. Denoising results of the Indian Pines dataset under noise case 1. The pseudocolor images are composed of three bands (R: 224, G: 112, and B: 10).

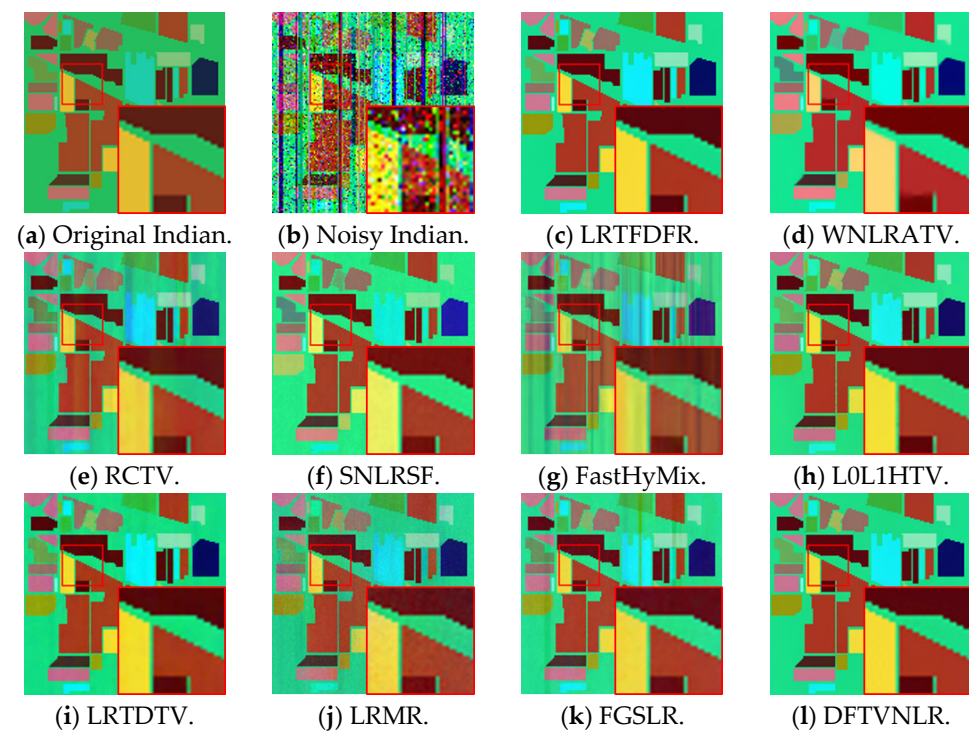


Figure 3. Denoising results of the Indian Pines dataset under noise case 5. The pseudocolor images are composed of three bands (R: 173, G: 63, and B: 2).

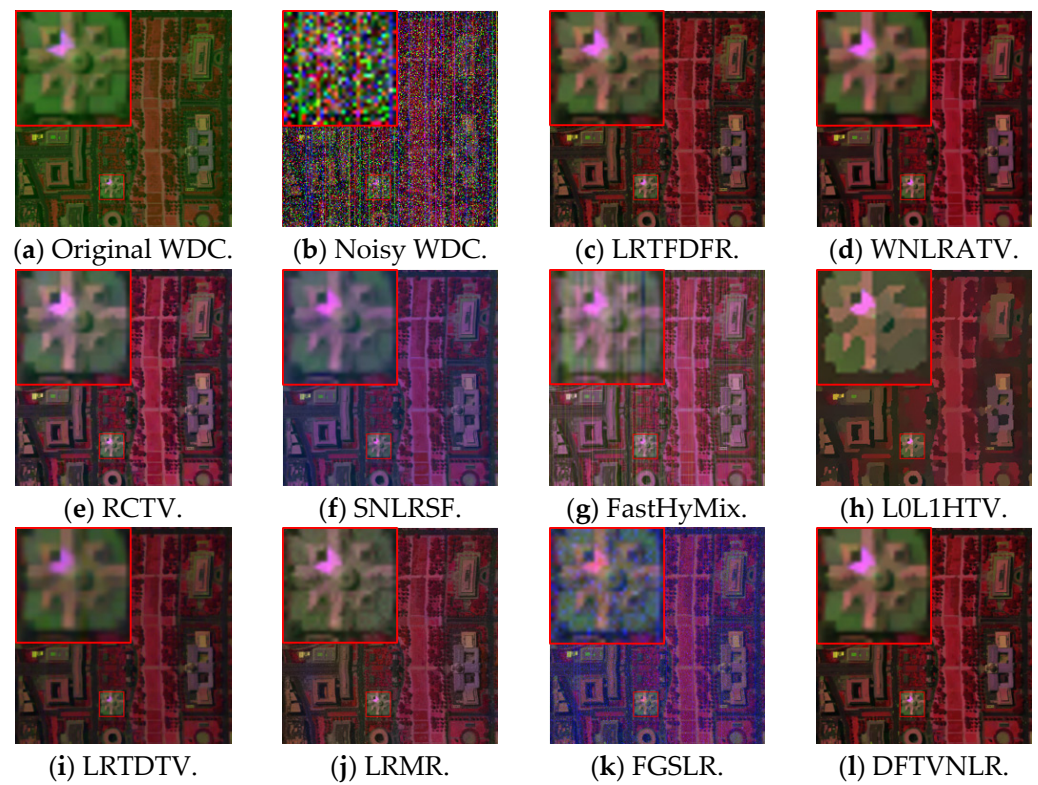


Figure 4. Denoising results of the WDC Mall dataset under noise case 4. The pseudocolor images are composed of three bands (R: 101, G: 13, and B: 188).

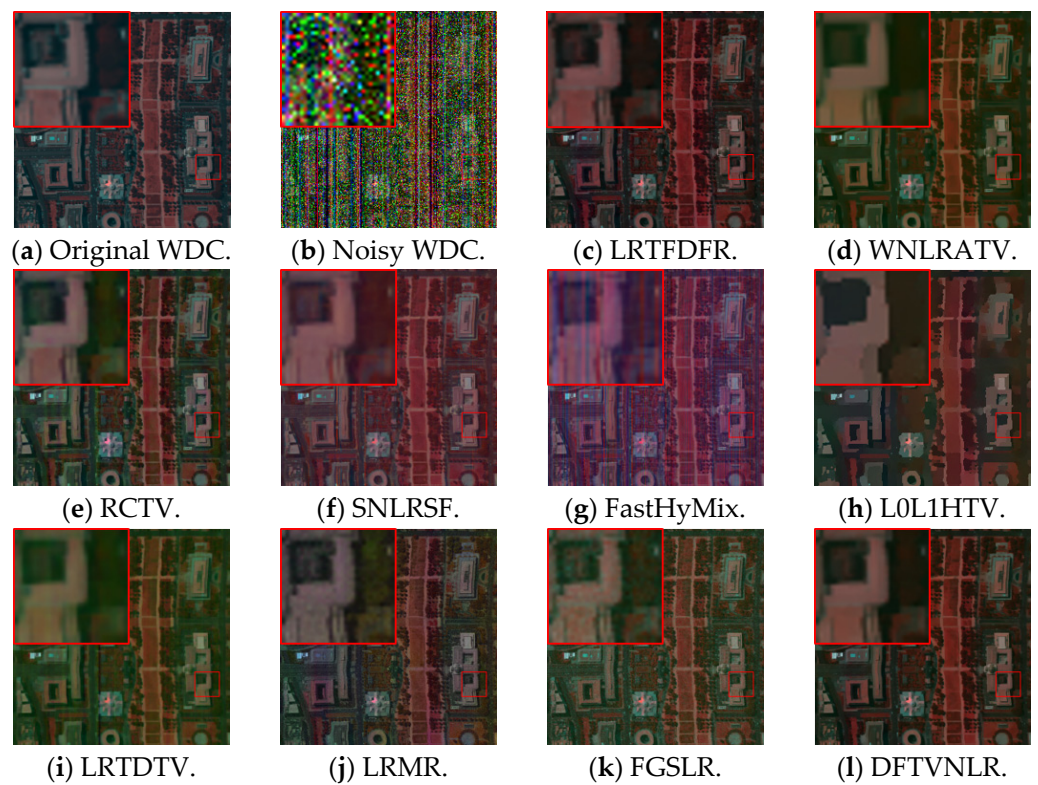


Figure 5. Denoising results of the WDC Mall dataset under noise case 5. The pseudocolor images are composed of three bands (R: 124, G: 24, and B: 19).

4.1.2. Quantitative Comparison

In terms of the aforementioned six quality metrics, Tables 3 and 4 present the denoised quantitative results of the Indian Pines and the WDC Mall datasets, respectively. Especially, we respectively highlight the top three results using bold, underlining, and italicization for each quality metrics. According to the quantitative results, the following four aspects can be obviously observed:

1. The proposed DFTVNLR can achieve the best results under most noise cases, but its MSSIM, MFSIM, or MSAM is slightly lower than LRTFDFR or WNLRTATV under some noise cases. This may be because our model is still difficult to completely match the real degraded model, resulting in a further “enhancement”. However, it enhances the visual effect to a certain extent.
2. The principle of DFTVNLR is closest to that of LRTFDFR, but our results are better than those of LRTFDFR on the whole, which shows it is reasonable and effective to introduce an NSS prior. Additionally, the NLR term of DFTVNLR is derived from WNLATV, but our results are better than those of WNLATV on the whole, which proves that the strategy based on double factor is more effective and robust.
3. Compared to the methods (LRTDTV, FGSLR, and L0L1HTV) that do not characterize the time-consuming NSS prior but impose constraints on the large-sized HSI, our algorithm is 1~2 times faster than them, thanks to our strategies of imposing constraints on small-sized factors and characterizing NSS outside the main loop. Compared to the methods (WNLRTATV and SNLRSF) that also consider NSS prior and regularize the small-sized spatial factor, our computational cost is reduced by about 4 times and 10 times, respectively, which is due to our strategy of independently solving the NLR model with fewer iterations outside the main loop.
4. Although our approach has made efforts to reduce or control the computational cost, it is currently difficult for us to achieve the optimal cost due to the fact we explore almost all major priors. Therefore, our cost is still higher than those of RCTV, FastHyMix, LRMR, and LRTFDFR, but our denoising performance is better. The faster execution of FastHyMix is due to the introduction of a deep image prior. The cost of RCTV is lower because it only imposes fewer regularization terms on small-sized spatial factor. LRMR is fast because only a few priors are mined. But the denoising performance of the above approaches is obviously insufficient. The cost of LRTFDFR is twice lower than ours because it does not consider an NSS prior.

For comparing the performance of various methods in each band, we show the partial bands’ PSNR and SSIM of the two simulated datasets in Figures 6 and 7, respectively. It can be seen that the proposed DFTVNLR can achieve the highest PSNR and SSIM values in most bands and noise cases. For further comparing the performance of the various methods in restoring and maintaining the spectral characteristics, we present the spectral curves of the two simulated datasets in Figures 8 and 9 under noise case 5. In addition, we also present the spectral curves of these two simulated datasets at other spatial locations and under other noise cases in the Supplementary Materials. It is observed that the spectral curves obtained by the proposed DFTVNLR can better approximate the original ones than those produced by the compared methods.

Table 3. Quantitative comparison of all compared methods for the Indian Pines dataset.

Metrics	Noisy	LRTFDFR	WNLRTV	RCTV	SNLRSF	FastHyMix	L0L1HTV	LRTDTV	LRMR	FGSLR	DFTVNLR
Case 1: Gaussian Noise											
MPSNR	16.312	<u>37.702</u>	28.352	32.682	27.496	31.397	31.303	<i>32.801</i>	27.614	30.330	38.139
MSSIM	0.3592	<u>0.9823</u>	<i>0.9694</i>	0.9680	0.9447	0.9642	0.9665	0.9647	0.8845	0.9486	0.9839
MFSIM	0.4863	<u>0.9960</u>	0.9830	0.9812	0.9730	0.9782	<i>0.9920</i>	0.9763	0.9066	0.9656	0.9963
MSAM	0.2763	<u>0.0304</u>	0.0797	0.0526	0.0872	0.0569	0.0654	<i>0.0483</i>	0.0859	0.0640	0.0284
ERGAS	367.81	<u>37.587</u>	101.56	67.008	110.15	71.166	88.168	<i>61.708</i>	113.16	82.106	36.447
Time (s)	0.0000	26.958	196.86	4.7124	560.02	<u>5.0000</u>	178.21	73.618	35.845	98.515	50.770
Case 2: Gaussian Noise + SaltandPepper Noise											
MPSNR	13.747	<u>37.687</u>	27.659	32.593	26.015	27.813	31.980	32.034	26.439	29.582	38.197
MSSIM	0.2181	<u>0.9816</u>	0.9622	0.9615	0.9107	0.9351	<i>0.9637</i>	0.9556	0.8537	0.9372	0.9840
MFSIM	0.3870	<u>0.9957</u>	0.9769	0.9747	0.9451	0.9432	<i>0.9900</i>	0.9654	0.8807	0.9537	0.9958
MSAM	0.3689	<u>0.0289</u>	0.0821	0.0525	0.0986	0.0820	0.0559	<i>0.0473</i>	0.1004	0.0710	0.0268
ERGAS	486.39	<u>37.496</u>	109.02	67.773	125.93	106.98	81.603	<i>65.330</i>	134.78	92.077	35.186
Time	0.0000	28.341	226.70	<u>4.6043</u>	557.89	4.2800	175.70	75.821	38.530	159.56	53.312
Case 3: Gaussian Noise + SaltandPepper Noise + Deadlines											
MPSNR	13.521	<u>36.658</u>	27.701	29.690	25.837	26.278	29.969	29.754	25.971	29.512	37.007
MSSIM	0.2119	<u>0.9825</u>	<i>0.9611</i>	0.9312	0.9077	0.8757	0.9589	0.9423	0.8460	0.9243	0.9830
MFSIM	0.3835	<u>0.9953</u>	0.9761	0.9509	0.9380	0.8873	<i>0.9845</i>	0.9559	0.8727	0.9415	0.9954
MSAM	0.3803	<u>0.0339</u>	0.0814	0.0835	0.0932	0.1067	<i>0.0701</i>	0.0794	0.1030	0.0753	0.0323
ERGAS	500.19	<u>42.8395</u>	106.77	103.35	126.83	138.39	95.602	99.151	137.78	94.809	41.345
Time	0.0000	28.868	220.43	<u>4.5730</u>	562.29	4.1600	176.19	73.441	35.975	437.95	53.962
Case 4: Gaussian Noise + SaltandPepper Noise + Stripes											
MPSNR	13.630	<u>35.671</u>	27.573	30.489	25.749	26.742	29.473	<i>30.814</i>	26.129	29.270	35.954
MSSIM	0.2137	<u>0.9797</u>	0.9586	0.9529	0.9086	0.9165	<i>0.9589</i>	0.9501	0.8479	0.9328	0.9805
MFSIM	0.3831	0.9951	<i>0.9753</i>	0.9691	0.9400	0.9206	<u>0.9882</u>	0.9606	0.8744	0.9502	0.9951
MSAM	0.3730	<u>0.0362</u>	0.0839	0.0712	0.1007	0.0957	0.0730	<i>0.0560</i>	0.1053	0.0749	0.0355
ERGAS	494.81	<u>47.767</u>	116.31	94.314	132.61	121.42	99.085	<i>75.942</i>	138.41	99.220	46.485
Time	0.0000	25.833	228.73	<u>4.6481</u>	562.17	3.9400	175.14	73.202	37.221	171.56	55.544
Case 5: Gaussian Noise + SaltandPepper Noise + Stripes + Deadlines											
MPSNR	13.379	<u>34.921</u>	27.497	28.394	25.577	24.792	29.784	29.921	25.364	28.118	35.269
MSSIM	0.2069	0.9795	<i>0.9582</i>	0.9194	0.9005	0.8395	0.9560	0.9400	0.8375	0.9128	<u>0.9790</u>
MFSIM	0.3801	0.9948	0.9749	0.9393	0.9336	0.8567	<i>0.9819</i>	0.9521	0.8638	0.9303	<u>0.9943</u>
MSAM	0.3856	<u>0.0410</u>	0.0841	0.0932	0.1019	0.1268	<i>0.0652</i>	0.0713	0.1149	0.0867	0.0370
ERGAS	509.64	<u>52.076</u>	115.46	120.25	133.71	162.50	<i>91.164</i>	92.999	150.12	112.97	47.555
Time	0.0000	28.918	199.38	3.7623	569.10	<u>5.8200</u>	126.32	75.541	36.245	502.65	53.756

The results marked in bold, underlined and italicized represent the top three results, respectively.

Table 4. Quantitative comparison of all compared methods for the WDC Mall dataset.

Metrics	Noisy	LRTDFR	WNLRTV	RCTV	SNLRSF	FastHyMix	L0L1HTV	LRTDTV	LRMR	FGSLR	DFTVNLR
Case 1: Gaussian Noise											
MPSNR	13.168	29.001	28.513	27.960	<u>30.601</u>	29.935	22.821	26.430	26.998	27.282	30.725
MSSIM	0.2864	0.9240	0.8878	0.8973	<u>0.9426</u>	0.9339	0.6131	0.8231	0.8643	0.8873	0.9463
MFSIM	0.7176	0.9709	0.9304	0.9500	<u>0.9745</u>	0.9718	0.7426	0.9203	0.9440	0.9605	0.9760
MSAM	0.5006	0.1954	0.1732	0.1968	<u>0.1551</u>	0.1644	0.3787	0.3222	0.2114	0.2047	0.1488
ERGAS	1054.5	162.38	153.17	187.61	<u>134.84</u>	140.30	286.22	216.23	201.65	207.81	125.91
Time (s)	0.0000	<i>90.134</i>	626.39	<u>13.174</u>	1692.8	10.130	478.94	156.18	105.53	1276.1	188.87
Case 2: Gaussian Noise + SaltandPepper Noise											
MPSNR	12.731	27.871	<u>28.248</u>	25.751	26.228	24.490	22.696	26.112	26.429	24.987	29.338
MSSIM	0.1659	<u>0.9145</u>	0.8642	0.8675	0.8591	0.8343	0.6074	0.8014	0.8426	0.8073	0.9252
MFSIM	0.6305	0.9667	0.9179	<i>0.9414</i>	0.9382	0.9350	0.7360	0.9079	0.9327	0.9279	<u>0.9615</u>
MSAM	0.5345	0.2320	0.1562	0.2921	0.2312	0.3102	0.3818	0.3391	<i>0.2180</i>	0.3757	<u>0.1784</u>
ERGAS	1028.6	<i>168.98</i>	<u>154.60</u>	242.27	224.57	312.79	287.84	228.78	204.37	447.56	138.97
Time	0.0000	<i>89.041</i>	629.39	12.377	1691.5	<u>16.210</u>	462.36	157.67	118.98	6651.2	190.96
Case 3: Gaussian Noise + SaltandPepper Noise + Deadlines											
MPSNR	12.764	27.965	<u>28.311</u>	25.801	24.283	23.878	22.721	25.537	26.130	24.054	28.716
MSSIM	0.1649	<u>0.9136</u>	0.8638	<i>0.8674</i>	0.8377	0.8282	0.6017	0.7948	0.8401	0.7846	0.9170
MFSIM	0.6294	0.9658	0.9188	0.9373	0.9353	0.9318	0.7339	0.9039	0.9293	0.9133	<u>0.9628</u>
MSAM	0.5367	<i>0.2449</i>	0.1584	0.2806	0.3165	0.3416	0.3800	0.3743	0.2462	0.4275	<u>0.2088</u>
ERGAS	1029.5	<i>171.05</i>	<u>154.25</u>	219.39	267.26	313.00	290.93	243.99	208.08	473.78	149.87
Time	0.0000	<i>91.121</i>	629.16	12.824	1687.8	<u>17.130</u>	472.53	156.49	112.65	7140.0	187.42
Case 4: Gaussian Noise + SaltandPepper Noise + Stripes											
MPSNR	12.578	<u>27.517</u>	27.366	24.414	24.442	22.031	21.785	25.400	25.150	21.781	28.450
MSSIM	0.1615	<u>0.9085</u>	<i>0.8679</i>	0.8443	0.8330	0.7566	0.5908	0.7899	0.8322	0.7502	0.9163
MFSIM	0.6260	0.9659	0.9250	<i>0.9372</i>	0.9331	0.9103	0.7354	0.9053	0.9278	0.9040	<u>0.9615</u>
MSAM	0.5378	<i>0.2714</i>	0.2072	0.3638	0.3155	0.4391	0.4011	0.3980	0.2908	0.5032	<u>0.2274</u>
ERGAS	1048.0	<i>182.88</i>	<u>178.34</u>	290.17	288.67	461.74	323.23	264.59	223.34	538.91	166.86
Time	0.0000	89.871	638.22	12.990	1699.2	<u>15.440</u>	471.36	156.19	128.12	4645.8	189.11
Case 5: Gaussian Noise + SaltandPepper Noise + Stripes + Deadlines											
MPSNR	12.657	<u>28.652</u>	<i>28.094</i>	24.615	24.104	21.210	23.119	24.907	25.639	24.320	29.631
MSSIM	0.1610	<u>0.9165</u>	<i>0.8648</i>	0.8422	0.8255	0.7309	0.6072	0.7892	0.8312	0.7812	0.9235
MFSIM	0.6253	0.9649	0.9185	<i>0.9320</i>	0.9307	0.8991	0.7339	0.9041	0.9258	0.9133	<u>0.9594</u>
MSAM	0.5374	<i>0.2162</i>	0.1671	0.3042	0.3300	0.4835	0.2878	0.3907	0.2652	0.4081	<u>0.1789</u>
ERGAS	1037.7	<i>160.21</i>	<u>156.70</u>	280.78	306.97	548.08	270.20	253.70	228.40	482.66	138.54
Time	0.0000	<i>90.389</i>	718.10	13.266	2640.6	<u>14.440</u>	369.64	157.34	123.19	7038.9	189.68

The results marked in bold, underlined and italicized represent the top three results, respectively.

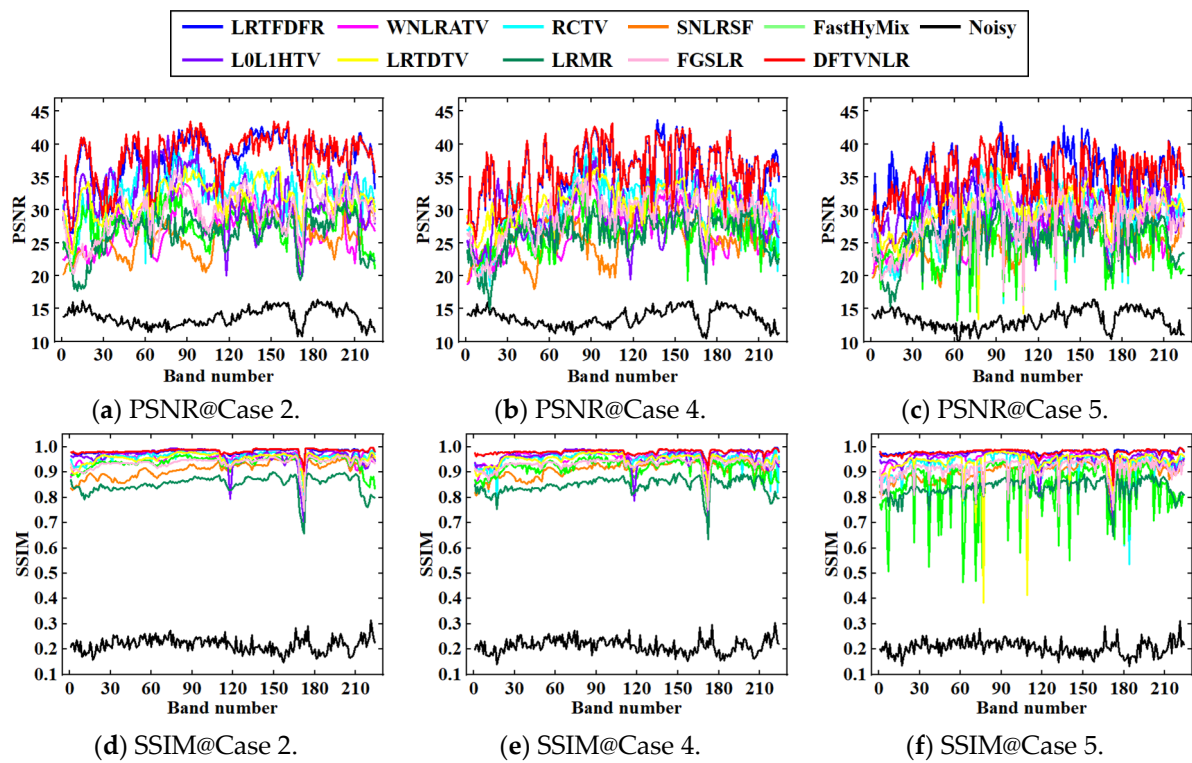


Figure 6. The PSNR and SSIM values of all bands of the Indian Pine dataset by the different, compared methods for the different noise cases.

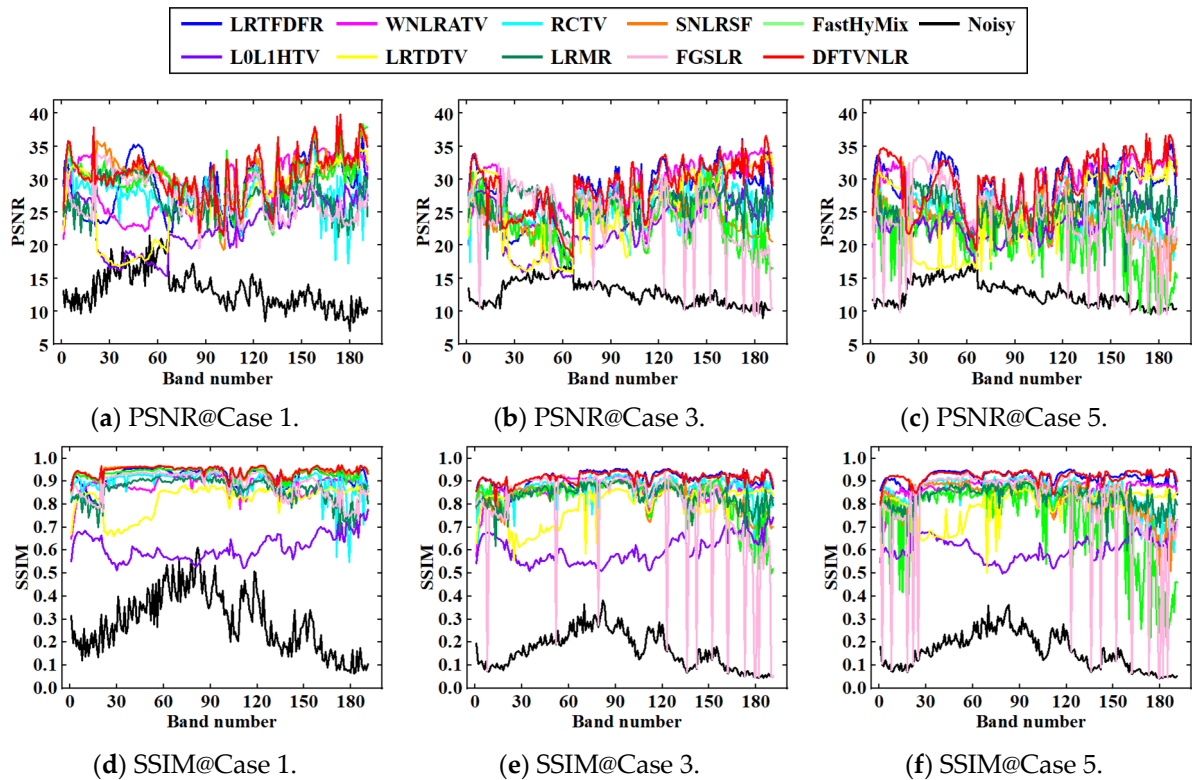


Figure 7. The PSNR and SSIM values of all bands of the WDC Mall dataset by the different, compared methods for the different noise cases.

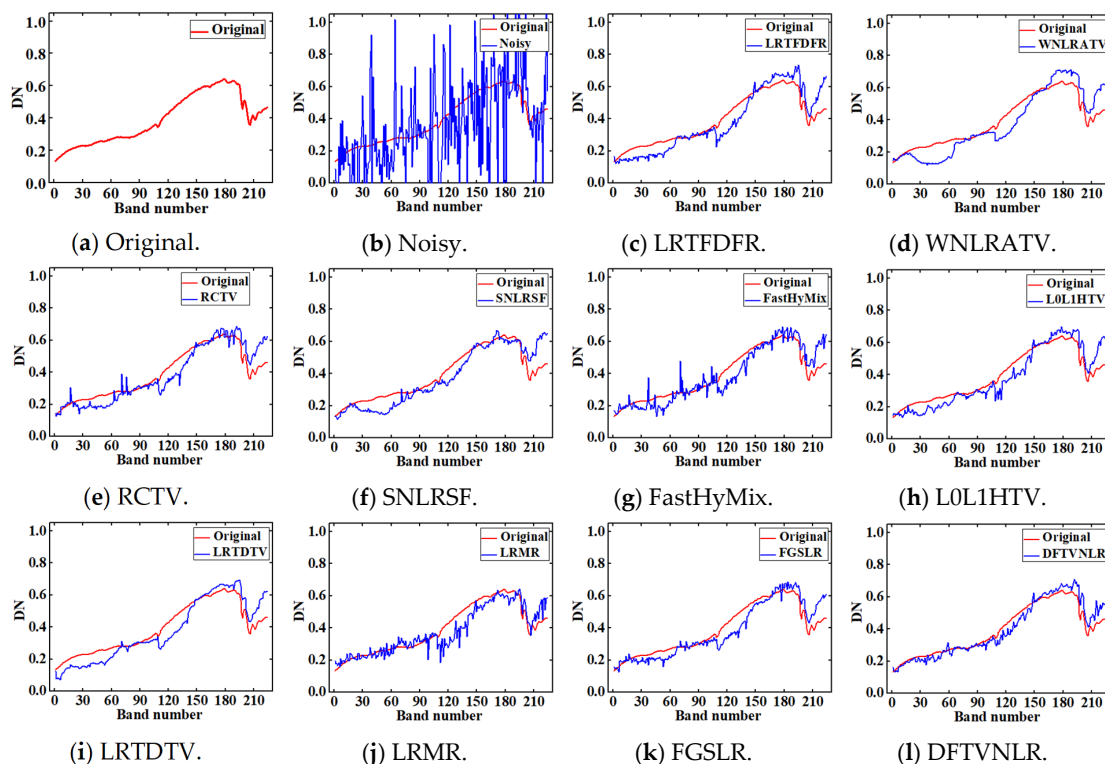


Figure 8. The spectral curves at the spatial location (70, 70) of the Indian Pine dataset for the noise case 5 by the different, compared methods.

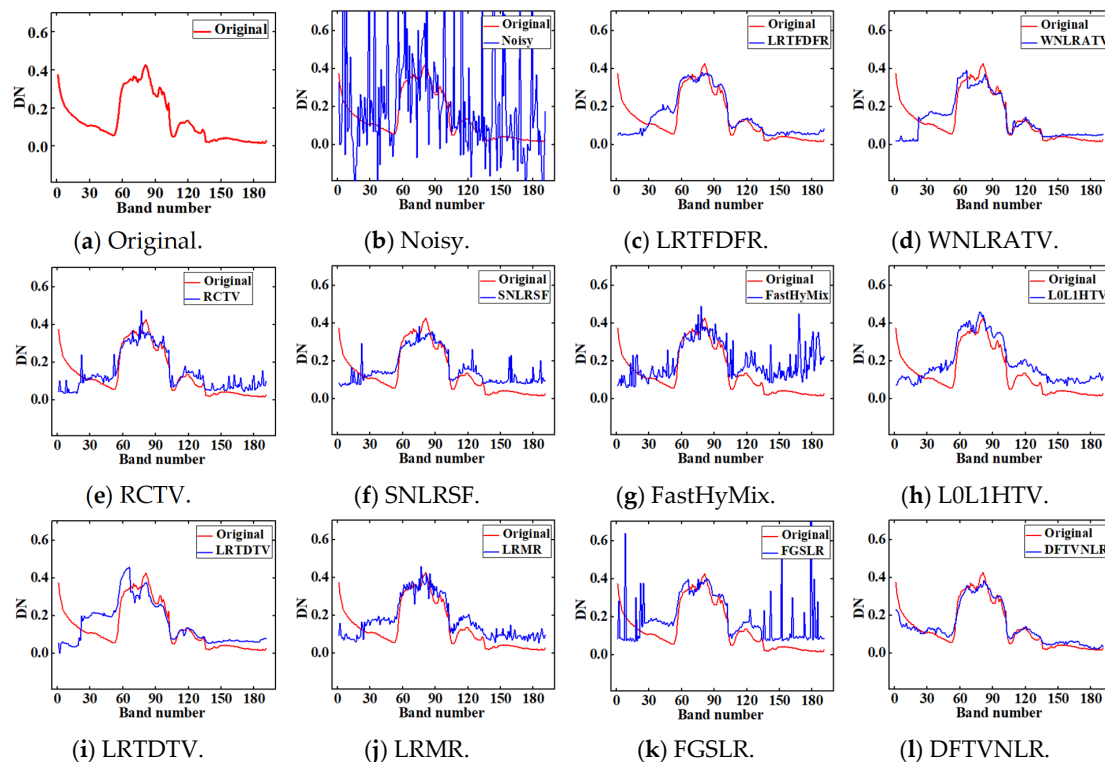


Figure 9. The spectral curves at the spatial location (188, 188) of the WDC Mall dataset for the noise case 5 by the different, compared methods.

4.2. Real Data Experiments

In this section, we conducted further experiments on two real HSI datasets. One was the Earth Observing-1 (EO-1) Hyperion dataset (<http://datamirror.csdn.cn/admin/dataEO1Main.jsp>, accessed on 10 April 2023), with the size of 200 (rows) \times 200 (columns) \times 166 (bands), which has a rich texture and is also used in SNLRSE and LRMR. The other was the AVIRIS Indian Pines dataset (<http://www.ehu.es/ccwintco/index.php>, accessed on 15 April 2023), with the size of 145 (rows) \times 145 (columns) \times 220 (bands), which is rich in edges and is also used in LRTFDFR, WNLRTV, SNLRSE, L0L1HTV, LRTDTV, and FGSLR. In both datasets, some bands are heavily polluted by Gaussian noise, salt and pepper noise, stripes, deadlines, or other unknown types of noise.

For the real HSI, we could not obtain the noiseless image and evaluation metrics. Alternatively, we randomly selected three bands and synthesized pseudocolor images for display. Furthermore, we presented the column DN mean curve in some bands of the denoised image, which should have a certain degree of smoothness.

4.2.1. EO-1 Hyperion Dataset

Figure 10 shows the denoising results of bands 166, 18, and 1 of the real EO-1 Hyperion dataset, including the pseudocolor visual images and the column DN mean curve in the corresponding bands. According to the pseudocolor image results, the following aspects can be obviously observed. First, the selected bands of EO-1 Hyperion are seriously polluted by Gaussian noise, stripes, and deadlines. Second, the denoising results of L0L1HTV and LRTDTV are oversmoothed, which may be related to the improper setting of the TV terms. Third, LRTFDFR and FastHyMix are excellent at keeping image details, but their denoising results tend to decline in pixel DN and image contrast. Fourth, the other compared methods remove mixed noise to varying degrees without image distortion, but the proposed DFTVNLR has greater advantages in maintaining the global structure and local details, and fully restoring the contrast.

Furthermore, we quantitatively evaluated the actual denoising effect in terms of the column DN mean curves. The curves show rapid fluctuations, as presented in Figure 10a-1,a-3, which imply the existence of noise. Meantime, it can be observed that band 1 and band 18 are less affected by noise. After denoising, the fluctuations in band 166 are suppressed more or less. Specifically, the curves of RCTV, SNLRSE, FastHyMix, LRTDTV, LRMR, and FGSLR still fluctuate slightly locally, indicating that there is still some noises. Although the curves of LRTFDFR and L0L1HTV are relatively smooth, the image contrast of LRTFDFR is low and L0L1HTV is oversmoothed. By contrast, the curves of WNLRTV and DFTVNLR are smooth and the DN curves are closest to the actual image.

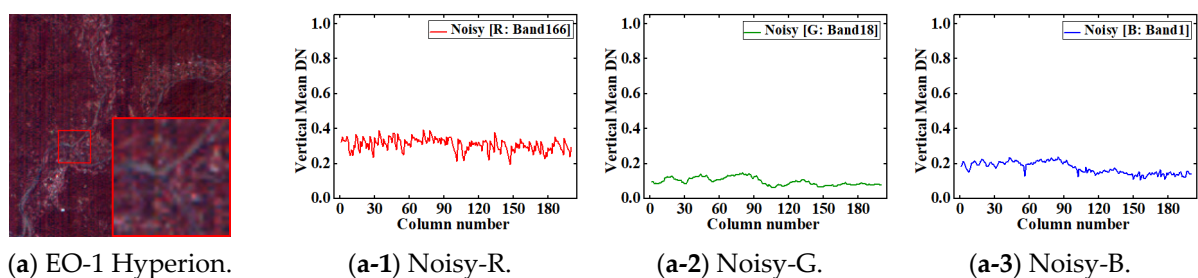


Figure 10. Cont.

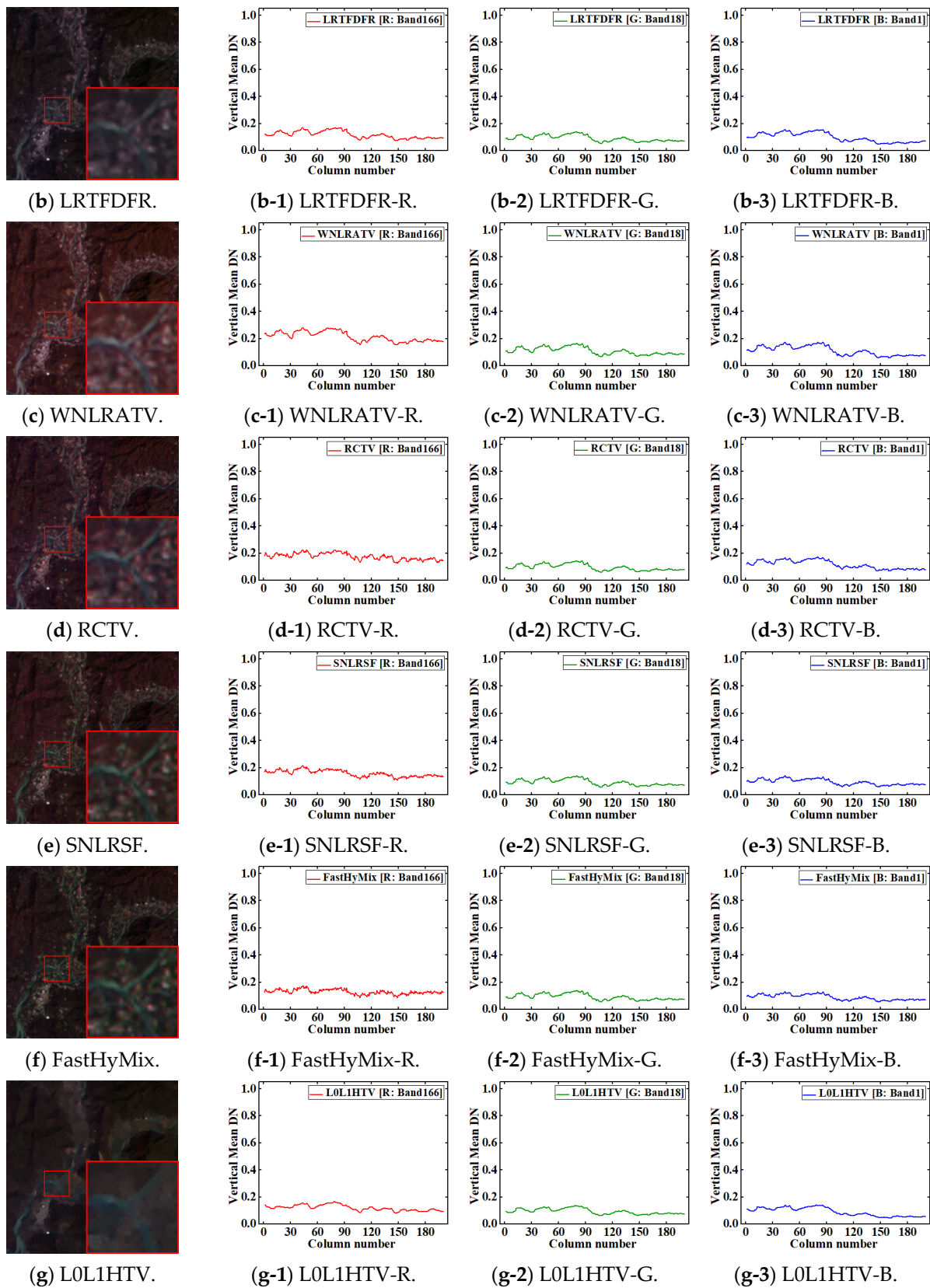


Figure 10. Cont.

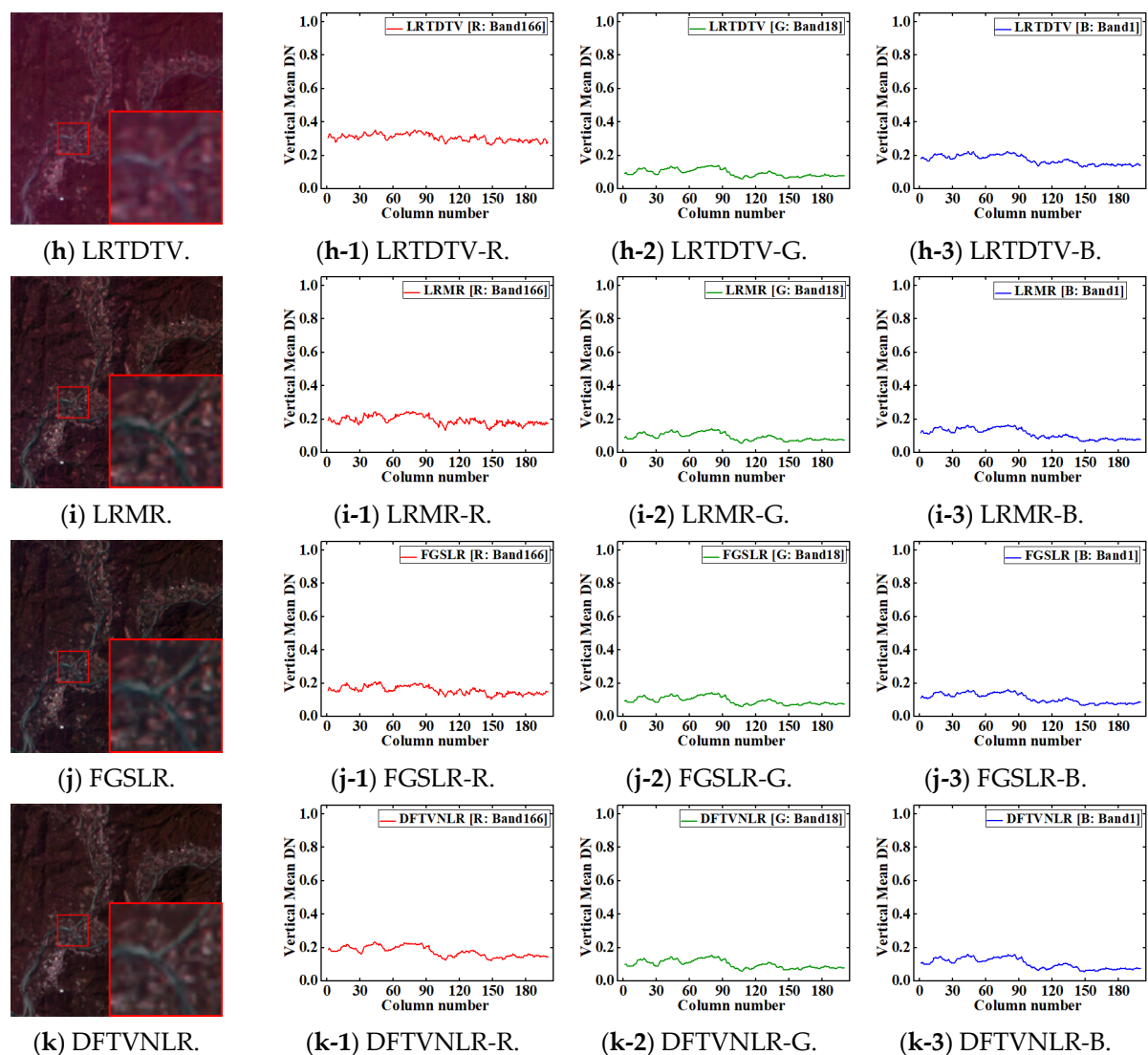


Figure 10. Denoising results of pseudocolor synthetic bands (R: 166, G: 18, and B: 1) selected in the real EO-1 Hyperion dataset. The rows from top to bottom, respectively, represent the noisy /denoised pseudocolor images and the column DN mean curve in the corresponding bands, where (a) Noisy Hyperion, (b) LRTDFR, (c) WNLRTV, (d) RCTV, (e) SNLRSE, (f) Fast Hymix, (g) LOL1 HTV, (h) LRTDTV, (i) LRMR, (j) FGSLR, (k) DFTVNLR (proposed).

4.2.2. AVIRIS Indian Pines Dataset

Figure 11 shows the denoising results of bands 220, 161, and 1 of the real AVIRIS Indian Pines dataset, including the pseudocolor visual images and the column DN mean curve in the corresponding bands. According to the pseudocolor image results, the following aspects can be obviously observed. First, the selected bands of AVIRIS Indian Pines are seriously polluted by Gaussian noise and salt and pepper noise. Second, there is still some noises in RCTV, SNLRSE, LRMR, and FGSLR, which is related to insufficient prior exploration. Third, the image contrast is seriously changed in FastHyMix, LOL1HTV, LRTDTV, and LRMR, which may be related to the LS or LR prior being ignored. Fourth, LRTDFR, WNLRTV, and the proposed DFTVNLR are relatively excellent in removing mixed noise and retaining image contrast, but the proposed DFTVNLR has greater advantages in restoring the global structure, local details, and clarity.

Moreover, based on the column DN mean curves, we quantitatively evaluated the actual denoising effect to some extent. Due to the existence of noise, the curves show rapid fluctuations, as presented in Figure 11a–a-3. After denoising, the fluctuations are

suppressed more or less. The column DN mean curve tends to be oversmoothed and closer to a constant in FastHyMix, L0L1HTV, LRTDTV, and LRMR, which is consistent with the phenomenon of blurring and contrast reduction, as shown in Figure 11f–i. In contrast, LRTFDFR, WNLRATV, and the proposed DFTVNLN can produce a relatively smooth column DN mean curve, but there is a certain fluctuation range, which indirectly shows that it can not only remove noises but also retain some local texture information. However, for the proposed DFTVNLN, its overall fluctuation is more natural, the local fluctuation is smoother, and the DN is closest to the actual image.

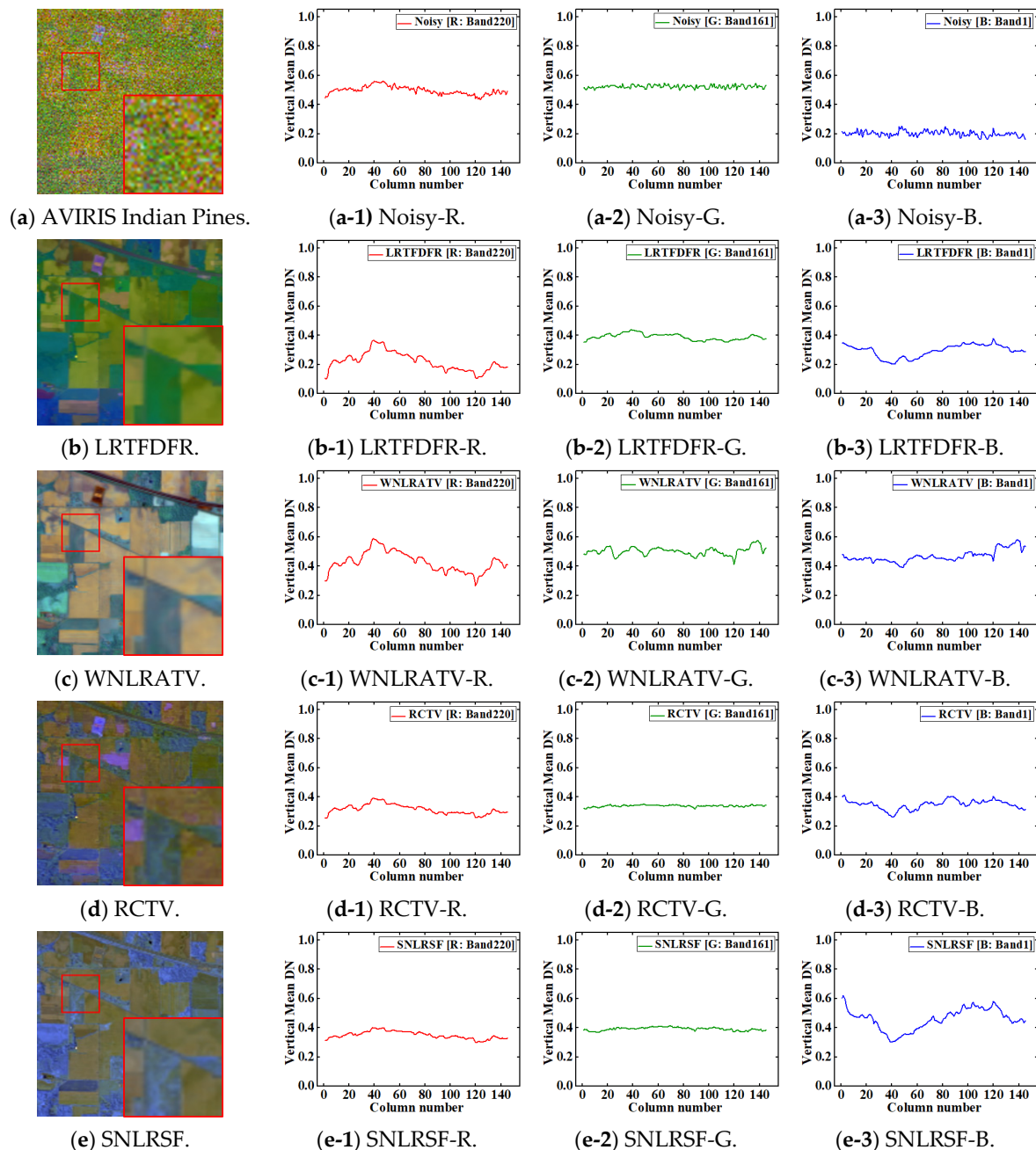


Figure 11. Cont.

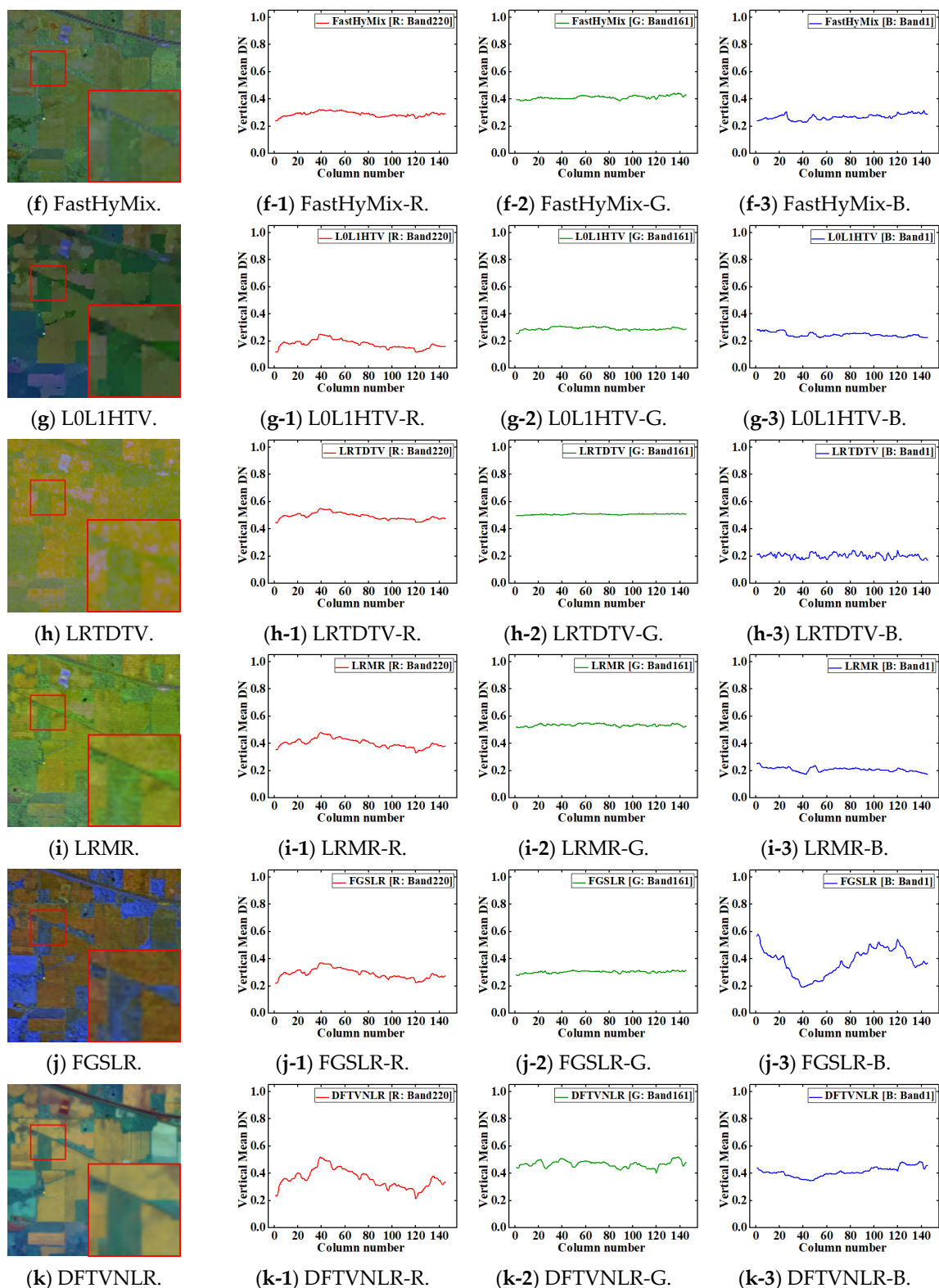


Figure 11. Denoising results of pseudocolor synthetic bands (R: 220, G: 161, and B: 1) selected in the real AVIRIS IndianPines dataset. The rows from top to bottom, respectively, represent the noisy/denoised pseudocolor images and the column DN mean curve in the corresponding bands, where (a) Noisy Hyperion, (b) LRTDFR, (c) WNLRTV, (d) RCTV, (e) SNRLRSF, (f) FastHyMix, (g) L0L1 HTV, (h) LRTDTV, (i) LRM, (j) FGSLR, and (k) DFTVNLR (proposed).

4.3. Discussion

As shown in Algorithm 1, there are seven crucial parameters in the proposed DFTVNLR, which need to be set in advance, i.e., rank R , regularization parameters λ_1 , λ_2 , λ_3 and λ_4 , penalty parameter μ , and proximal parameter ρ . Among them, the parameter ρ is used to ensure the theoretical convergence of PAM solver. References [31,54] suggested that ρ should be set to 0.1. Next, we investigated the sensibility of other parameters.

4.3.1. Parameter Analysis

The R mainly characterizes the LR of the HSI, whose sensitivity analysis is presented in Figure 12. When the R is between [12,34], the results exhibit a stable and superior performance for various datasets and noise cases. Considering a larger R leads to a higher calculation cost; so, the R should be set 12 or estimated by the HySime.

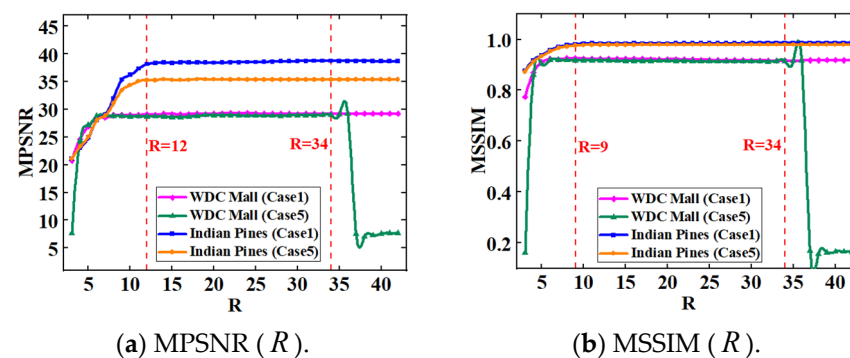


Figure 12. Sensitivity analysis of the DFTVNLR model to rank R .

The λ_1 and λ_2 determine the weights of the \mathcal{U} – \mathcal{V} -based regularization terms. The λ_3 and λ_4 determine the weight of NLR term and \mathcal{S} , respectively. The sensitivity analyses of λ_1 , λ_2 , λ_3 , and λ_4 are presented in Figures 13–16, respectively. It can be seen that λ_1 and λ_4 are sensitive to both datasets and noise cases, but the change trend of sensitivity is consistent. Additionally, λ_3 is sensitive to datasets, but is robust to noise cases. The proposed DFTVNLR is relatively robust to λ_2 . However, when $\lambda_1 = 0.2$, $\lambda_2 \in [0.005, 0.03]$, $\lambda_3 \in [0.2, 10]$, and $\lambda_4 \in [0.03, 0.04]$, the denoising performance is stable and superior under different noise cases and datasets.

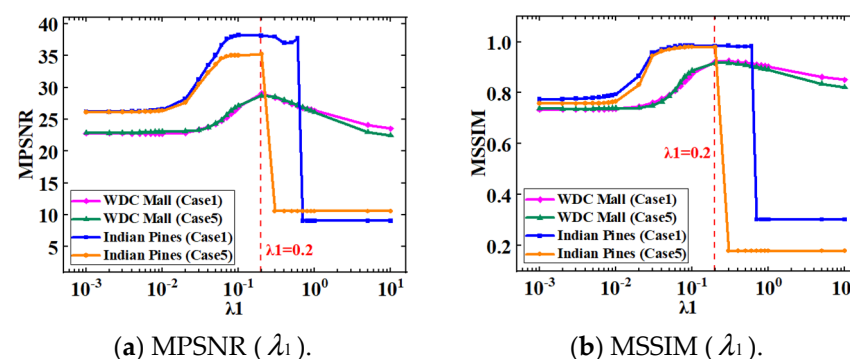
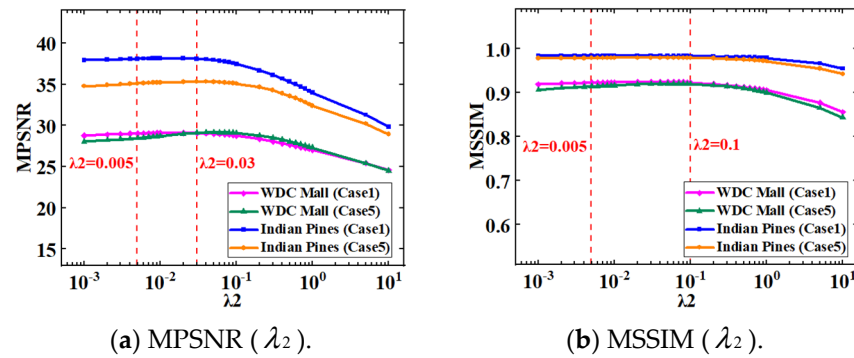
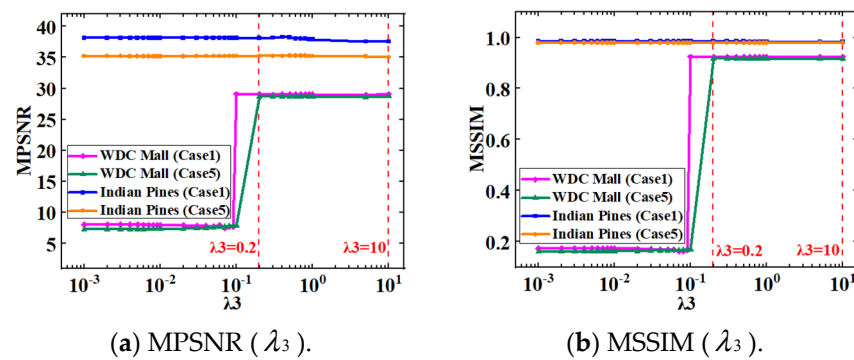
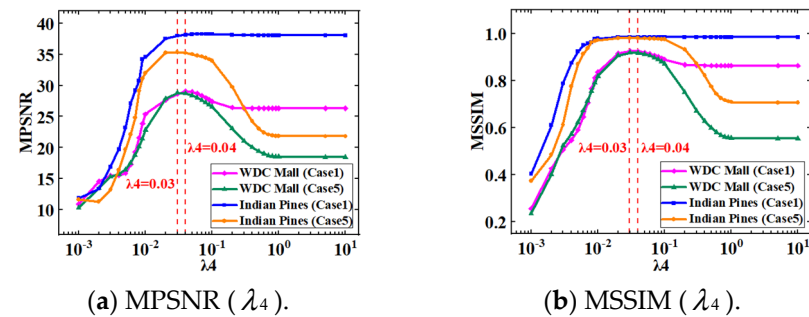
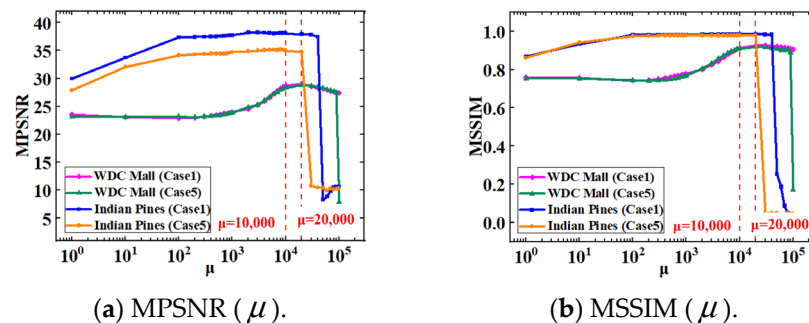


Figure 13. Sensitivity analysis of the DFTVNLR model to λ_1 .

Figure 14. Sensitivity analysis of the DFTVNLR model to λ_2 .Figure 15. Sensitivity analysis of the DFTVNLR model to λ_3 .Figure 16. Sensitivity analysis of the DFTVNLR model to λ_4 .

The sensitivity analysis of μ is presented in Figure 17. It can be clearly observed that the sensitivity trend of μ to different datasets and noise cases is consistent. When $\mu \in [10,000, 20,000]$, higher MPSNR and MSSIM can be obtained under different noise cases and datasets.

Figure 17. Sensitivity analysis of the DFTVNLR model to μ .

4.3.2. Convergence Analysis

In this section, we indirectly verify the convergence of Algorithm 1 from the perspective of numerical analysis. Figure 18 shows the MPSNR and MSSIM of the denoising results for different simulated datasets under noise cases 1 and 5, and the relative changes in adjacent iteration results. As the number of iterations increases, the MPSNR and MSSIM monotonically increase, and the relative changes monotonically decrease. When $L \geq 35$, the relative variation errors are stable at 0. This verifies the convergence of Algorithm 1 numerically. Based on the above analysis results, we list the parameter setting suggestions of DFTVNLR in Table 5.

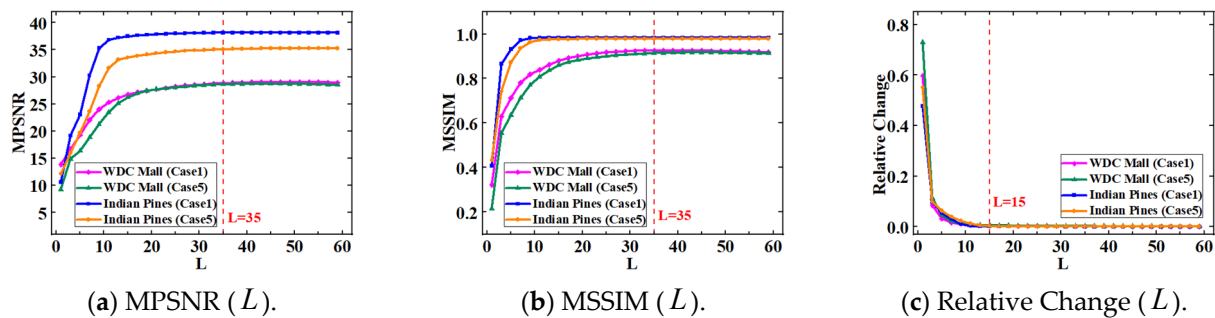


Figure 18. Convergence analysis of the proposed DFTVNLR.

Table 5. Suggestions on the parameter setting of DFTVNLR when it is used in simulated and real noisy HSI datasets.

Parameter		R	λ_1	λ_2	λ_3	λ_4	μ	L
Dataset								
Simulated HSI dataset	Indian Pines	12	0.2	0.005	0.2	0.03	10,000	≥ 35
	WDC Mall	9 or 12						
Real noisy HSI dataset	Hyperion	12 or estimated by HySime	0.2	~ 0.03	10	~ 0.04	20,000	≥ 35
	Indian Pines							

5. Conclusions

In this article, we propose a novel DFTVNLR method for HSI mixed noise removal, which fully considers the intrinsic priors of HSI and mixed noise. First, the LRTF framework was used to characterize the LR. Then, the DFTV and weighted strategy were applied to describe and promote the spatial-spectral LS. Additionally, we employed the NLR model to characterize the NSS. In order to solve the proposed DFTVNLR model, we proposed an PAM algorithm based on ADMM and ALM, which is theoretically convergent. Particularly, we control the computational cost in two ways, namely regularizing the small-sized double-factors and solving the NLR outside the main loop. Additionally, in the extensive numerical experiments, the proposed DFTVNLR exhibits its superior performance in mixed noise removal, spatial information recovery, and spectral signatures preserving.

In the future, we will focus on and try to solve the following problems. First, we will incorporate the noise modeling idea [41,46] into our DFTVNLR method to further enhance its capability in real-life scenarios. Second, there may be many kinds of degradations in actual HSIs that are difficult to decouple, such as noise and blur [57]. Therefore, it is of great practical value to study the HSI restoration method that simultaneously removes multiple degradations. Third, the model-based and the data-based methods will be combined to reduce the computational cost. Fourth, we assumed that the mixed noise in actual HSIs is mainly additive noise, which has certain limitations [58]. Studying a reasonable noise model that is more in line with the actual situation will help to further improve the denoising performance.

Supplementary Materials: The following supporting information can be downloaded at: <https://www.mdpi.com/article/10.3390/rs16101686/s1>, Figure S1. The spectral curves at spatial location (30, 30) of the Indian Pines dataset under noise Case 3 by different compared methods. (a) Original, (b) Noisy, (c) LRTFDFR, (d) WNLRATV, (e) RCTV, (f) SNLRSE, (g) FastHyMix, (h) L0L1HTV, (i) LRTDTV, (j) LRMR, (k) FGSLR, (l) DFTVNLR (proposed). Figure S2. The spectral curves at spatial location (151, 151) of the WDC Mall dataset under noise Case 4 by different compared methods. (a) Original, (b) Noisy, (c) LRTFDFR, (d) WNLRATV, (e) RCTV, (f) SNLRSE, (g) FastHyMix, (h) L0L1HTV, (i) LRTDTV, (j) LRMR, (k) FGSLR, (l) DFTVNLR (proposed).

Author Contributions: Conceptualization, Y.W., W.X. and L.Z.; methodology, Y.W. and W.X.; validation, Y.W.; formal analysis, Y.W.; investigation, Y.W.; writing—original draft preparation, Y.W.; writing—review and editing, Y.W., L.Z. and W.X.; visualization, Y.W.; supervision, W.X.; funding acquisition, W.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (62075219).

Data Availability Statement: Data is contained within the article.

Acknowledgments: The authors would like to thank the Editors and the anonymous reviewers for their constructive comments, which significantly improved the quality of the article.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

For facilitating a concise expression, the frequently used technical terms in this article are summarized as follows in the form of abbreviations:

HSI	hyperspectral image
RS	remote sensing
SAR	synthetic aperture radar
RCI	representation coefficient image
LR	spectral global low-rankness
LS	local smoothness
NSS	spatial nonlocal self-similarity
S	sparsity of sparse noise
TV	total variation
SSTV	spatial–spectral TV
ASSTV	adaptive SSTV
DFTV	spatial–spectral double factor and total variation
LRTF	low-rank tensor factorization
LRMF	low-rank matrix factorization
LRTV	low-rank TV
LRMR	low-rank matrix recovery
NLR	nonlocal low-rank tensor model
PAM	proximal alternating minimization
ADMM	alternating direction method of multipliers
ALM	augmented Lagrangian method
SVD	singular value decomposition
FFT	fast Fourier transformation
SBS	similar block search
SoftThreshold	soft-thresholding operation
MPSNR	the mean of peak signal-to-noise ratio
MSSIM	the mean of structural similarity
MFSIM	the mean of feature similarity
MSAM	the mean of spectral angle mapping
ERGAS	the Erreur Relative Globale Adimensionnelle de Synthèse

References

- Green, R.O.; Eastwood, M.L.; Sarture, C.M.; Chrien, T.G.; Aronsson, M. Imaging Spectroscopy and the Airborne Visible Infrared Imaging Spectrometer (Avisis). *Remote Sens. Environ.* **1998**, *65*, 227–248. [\[CrossRef\]](#)
- Peng, J.J.; Wang, H.L.; Cao, X.Y.; Liu, X.L.; Rui, X.Y.; Meng, D.Y. Fast Noise Removal in Hyperspectral Images Via Representative Coefficient Total Variation. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5546017. [\[CrossRef\]](#)
- Zhao, X.; Tao, R.; Li, W.; Li, H.-C.; Du, Q.; Liao, W.; Philips, W. Joint Classification of Hyperspectral and Lidar Data Using Hierarchical Random Walk and Deep CNN Architecture. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 7355–7370. [\[CrossRef\]](#)
- Ghamisi, P.; Yokoya, N.; Li, J.; Liao, W.; Liu, S.; Plaza, J.; Rasti, B.; Plaza, A. Advances in Hyperspectral Image and Signal Processing: A Comprehensive Overview of the State of the Art. *IEEE Geosci. Remote Sens. Mag.* **2017**, *5*, 37–78. [\[CrossRef\]](#)
- Dian, R.W.; Li, S.T.; Guo, A.J.; Fang, L.Y. Deep Hyperspectral Image Sharpening. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 5345–5355. [\[CrossRef\]](#) [\[PubMed\]](#)
- Zhang, L.; Nie, J.T.; Wei, W.; Li, Y.; Zhang, Y.N. Deep Blind Hyperspectral Image Super-Resolution. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 2388–2400. [\[CrossRef\]](#) [\[PubMed\]](#)
- Lin, C.-H.; Bioucas-Dias, J.M. Nonnegative Blind Source Separation for Ill-Conditioned Mixtures Via John Ellipsoid. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 2209–2223.
- He, W.; Zhang, H.Y.; Zhang, L.P.; Shen, H.F. Total-Variation-Regularized Low-Rank Matrix Factorization for Hyperspectral Image Restoration. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 176–188. [\[CrossRef\]](#)
- Wang, Y.; Peng, J.J.; Zhao, Q.; Leung, Y.; Zhao, X.L.; Meng, D.Y. Hyperspectral Image Restoration Via Total Variation Regularized Low-Rank Tensor Decomposition. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **2018**, *11*, 1227–1243. [\[CrossRef\]](#)
- Zhou, L.; Ma, X.; Wang, X.; Hao, S.; Ye, Y.; Zhao, K. Shallow-to-Deep Spatial-Spectral Feature Enhancement for Hyperspectral Image Classification. *Remote Sens.* **2023**, *15*, 261. [\[CrossRef\]](#)
- Deng, C.; Chen, Y.; Zhang, S.; Li, F.; Lai, P.; Su, D.; Hu, M.; Wang, S. Robust Dual Spatial Weighted Sparse Unmixing for Remotely Sensed Hyperspectral Imagery. *Remote Sens.* **2023**, *15*, 4056. [\[CrossRef\]](#)
- Pan, H.; Jing, Z.; Leung, H.; Peng, P.; Zhang, H. Multiband Image Fusion Via Regularization on a Riemannian Submanifold. *Remote Sens.* **2023**, *15*, 4370. [\[CrossRef\]](#)
- Deng, Y.-J.; Li, H.-C.; Tan, S.-Q.; Hou, J.; Du, Q.; Plaza, A. T-Linear Tensor Subspace Learning for Robust Feature Extraction of Hyperspectral Images. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5501015. [\[CrossRef\]](#)
- Sun, S.; Bao, W.; Qu, K.; Feng, W.; Zhang, X.; Ma, X. Hyperspectral Image Super-Resolution Algorithm Based on Graph Regular Tensor Ring Decomposition. *Remote Sens.* **2023**, *15*, 4983. [\[CrossRef\]](#)
- Ji, L.; Geng, X. Hyperspectral Target Detection Methods Based on Statistical Information: The Key Problems and the Corresponding Strategies. *Remote Sens.* **2023**, *15*, 3835. [\[CrossRef\]](#)
- Chang, Y.; Yan, L.; Fang, H.; Zhong, S.; Liao, W. HSI-DeNet: Hyperspectral Image Restoration Via Convolutional Neural Network. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 667–682. [\[CrossRef\]](#)
- Cao, X.-Y.; Fu, X.-Y.; Xu, C.; Meng, D.-Y. Deep Spatial-Spectral Global Reasoning Network for Hyperspectral Image Denoising. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5504714. [\[CrossRef\]](#)
- Chen, Y.; Cao, X.; Zhao, Q.; Meng, D.; Xu, Z. Denoising Hyperspectral Image with Non-i.i.d. Noise Structure. *IEEE Trans. Cybern.* **2018**, *48*, 1054–1066. [\[CrossRef\]](#)
- Peng, J.; Xie, Q.; Zhao, Q.; Wang, Y.; Yee, L.; Meng, D. Enhanced 3DTV Regularization and Its Applications on HSI Denoising and Compressed Sensing. *IEEE Trans. Image Process.* **2020**, *29*, 7889–7903. [\[CrossRef\]](#)
- Zhuang, L.; Bioucas-Dias, J.M. Fast Hyperspectral Image Denoising and Inpainting Based on Low-Rank and Sparse Representations. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **2018**, *11*, 730–742. [\[CrossRef\]](#)
- Xie, Q.; Zhao, Q.; Meng, D.; Xu, Z.; Gu, S. Multispectral Images Denoising by Intrinsic Tensor Sparsity Regularization. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 27–30 June 2016; pp. 1692–1700.
- Zheng, Y.-B.; Huang, T.-Z.; Zhao, X.-L.; Jiang, T.-X.; Ma, T.-H.; Ji, T.-Y. Mixed Noise Removal in Hyperspectral Image Via Low-Fibered-Rank Regularization. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 734–749. [\[CrossRef\]](#)
- Lin, J.; Huang, T.-Z.; Zhao, X.-L.; Ma, T.-H.; Jiang, T.-X.; Zheng, Y.-B. A Novel Non-Convex Low-Rank Tensor Approximation Model for Hyperspectral Image Restoration. *Appl. Math. Comput.* **2021**, *408*, 126342. [\[CrossRef\]](#)
- Chang, Y.; Yan, L.-X.; Fang, H.-Z.; Liu, H. Simultaneous Destriping and Denoising for Remote Sensing Images with Unidirectional Total Variation and Sparse Representation. *IEEE Geosci. Remote Sens. Lett.* **2014**, *11*, 1051–1055. [\[CrossRef\]](#)
- Fan, H.-Y.; Li, C.; Guo, Y.-L.; Kuang, G.-Y.; Ma, J.-Y. Spatial-Spectral Total Variation Regularized Low-Rank Tensor Decomposition for Hyperspectral Image Denoising. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 6196–6213. [\[CrossRef\]](#)
- Jiang, T.-X.; Zhuang, L.-N.; Huang, T.-Z.; Zhao, X.-L.; Bioucas-Dias, J.M. Adaptive Hyperspectral Mixed Noise Removal. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5511413. [\[CrossRef\]](#)
- Zhao, B.; Ulfarsson, M.O.; Sveinsson, J.R.; Chanussot, J. Hyperspectral Image Denoising Using Spectral-Spatial Transform-Based Sparse and Low-Rank Representations. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5522125. [\[CrossRef\]](#)
- Chang, Y.; Yan, L.-X.; Zhao, X.-L.; Fang, H.-Z.; Zhang, Z.-J.; Zhong, S. Weighted Low-Rank Tensor Recovery for Hyperspectral Image Restoration. *IEEE Trans. Cybern.* **2020**, *50*, 4558–4572. [\[CrossRef\]](#)

29. Wang, M.; Wang, Q.; Chanussot, J.; Hong, D. L0-L1 Hybrid Total Variation Regularization and Its Applications on Hyperspectral Image Mixed Noise Removal and Compressed Sensing. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 7695–7710. [\[CrossRef\]](#)
30. Cao, C.-H.; Yu, J.; Zhou, C.-Y.; Hu, K.; Xiao, F.; Gao, X.-P. Hyperspectral Image Denoising Via Subspace-Based Nonlocal Low-Rank and Sparse Factorization. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **2019**, *12*, 973–988. [\[CrossRef\]](#)
31. Zheng, Y.-B.; Huang, T.-Z.; Zhao, X.-L.; Chen, Y.; He, W. Double-Factor-Regularized Low-Rank Tensor Factorization for Mixed Noise Removal in Hyperspectral Image. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 8450–8464. [\[CrossRef\]](#)
32. Yao, J.; Meng, D.; Zhao, Q.; Cao, W.; Xu, Z. Nonconvex-Sparsity and Nonlocal-Smoothness-Based Blind Hyperspectral Unmixing. *IEEE Trans. Image Process.* **2019**, *28*, 2991–3006. [\[CrossRef\]](#) [\[PubMed\]](#)
33. Dian, R.-W.; Li, S.-T.; Sun, B.; Guo, A.-J. Recent Advances and New Guidelines on Hyperspectral and Multispectral Image Fusion. *Inf. Fusion* **2021**, *69*, 40–51. [\[CrossRef\]](#)
34. Miao, Y.-C.; Zhao, X.-L.; Fu, X.; Wang, J.-L.; Zheng, Y.-B. Hyperspectral Denoising Using Unsupervised Disentangled Spatiospectral Deep Priors. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5513916. [\[CrossRef\]](#)
35. Zha, Z.-Y.; Wen, B.-H.; Yuan, X.; Zhang, J.-C.; Zhou, J.-T.; Lu, Y.-L.; Zhu, C. Nonlocal Structured Sparsity Regularization Modeling for Hyperspectral Image Denoising. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5510316. [\[CrossRef\]](#)
36. Wang, S.; Zhu, Z.-B.; Liu, Y.-F.; Zhang, B.-X. Weighted Group Sparse Regularized Tensor Decomposition for Hyperspectral Image Denoising. *Appl. Sci.* **2023**, *13*, 10363. [\[CrossRef\]](#)
37. Han, J.; Pan, C.; Ding, H.-Y.; Zhang, Z.-C. Double-Factor Tensor Cascaded-Rank Decomposition for Hyperspectral Image Denoising. *Remote Sens.* **2024**, *16*, 109. [\[CrossRef\]](#)
38. Kolda, T.G.; Bader, B.W. Tensor Decompositions and Applications. *SIAM Rev.* **2009**, *51*, 455–500. [\[CrossRef\]](#)
39. Beck, A.; Teboulle, M. Fast Gradient-Based Algorithms for Constrained Total Variation Image Denoising and Deblurring Problems. *IEEE Trans. Image Process.* **2009**, *18*, 2419–2434. [\[CrossRef\]](#)
40. Xu, L.; Zheng, S.-C.; Jia, J.-Y. Unnatural L0 Sparse Representation for Natural Image Deblurring. In Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 1107–1114.
41. Chen, Y.; Cao, W.-F.; Pang, L.; Cao, X.-Y. Hyperspectral Image Denoising with Weighted Nonlocal Low-Rank Model and Adaptive Total Variation Regularization. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 15. [\[CrossRef\]](#)
42. Xie, Y.; Qu, Y.-Y.; Tao, D.-C.; Wu, W.-W.; Yuan, Q.-Q.; Zhang, W.-S. Hyperspectral Image Restoration Via Iteratively Regularized Weighted Schatten P-Norm Minimization. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 4642–4659. [\[CrossRef\]](#)
43. Bioucas-Dias, J.M.; Plaza, A.; Dobigeon, N.; Parente, M.; Du, Q. Hyperspectral Unmixing Overview: Geometrical, Statistical, and Sparse Regression-Based Approaches. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2012**, *5*, 354–379. [\[CrossRef\]](#)
44. He, W.; Yao, Q.-M.; Li, C.; Yokoya, N.; Zhao, Q.-B. Non-Local Meets Global: An Iterative Paradigm for Hyperspectral Image Restoration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 2089–2107. [\[CrossRef\]](#)
45. Zhuang, L.-N.; Bioucas-Dias, J.M. Hyperspectral Image Denoising Based on Global and Non-Local Low-Rank Factorizations. In Proceedings of the 24th IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 1900–1904.
46. Zhuang, L.-N.; Ng, M.K. FastHyMix: Fast and Parameter-Free Hyperspectral Image Mixed Noise Removal. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *34*, 4702–4716. [\[CrossRef\]](#)
47. Bioucas-Dias, J.M.; Nascimento, J.M.P. Hyperspectral Subspace Identification. *IEEE Trans. Geosci. Remote Sens.* **2008**, *46*, 2435–2445. [\[CrossRef\]](#)
48. He, W.; Yao, Q.-M.; Li, C.; Yokoya, N.; Zhao, Q.-B. Non-Local Meets Global: An Integrated Paradigm for Hyperspectral Denoising. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 6861–6870.
49. Chen, Y.; He, W.; Yokoya, N.; Huang, T.-Z. Hyperspectral Image Restoration Using Weighted Group Sparsity-Regularized Low-Rank Tensor Decomposition. *IEEE Trans. Cybern.* **2020**, *50*, 3556–3570. [\[CrossRef\]](#)
50. Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; Eckstein, J. Distributed Optimization and Statistical Learning Via the Alternating Direction Method of Multipliers. *Found. Trends Mach. Learn.* **2010**, *3*, 1–122. [\[CrossRef\]](#)
51. Wu, C.-L.; Tai, X.-C. Augmented Lagrangian Method, Dual Methods, and Split Bregman Iteration for ROF, Vectorial TV, and High Order Models. *SIAM J. Imag. Sci.* **2010**, *3*, 300–339. [\[CrossRef\]](#)
52. Donoho, D.L. De-Noising by Soft-Thresholding. *IEEE Trans. Inf. Theory* **1995**, *41*, 613–627. [\[CrossRef\]](#)
53. Zhang, H.-Y.; He, W.; Zhang, L.-P.; Shen, H.-F.; Yuan, Q.-Q. Hyperspectral Image Restoration Using Low-Rank Matrix Recovery. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 4729–4743. [\[CrossRef\]](#)
54. Chen, Y.; Huang, T.-Z.; He, W.; Zhao, X.-L.; Zhang, H. Hyperspectral Image Denoising Using Factor Group Sparsity-Regularized Nonconvex Low-Rank Approximation. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 3110769. [\[CrossRef\]](#)
55. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [\[CrossRef\]](#) [\[PubMed\]](#)
56. Zhang, L.; Zhang, L.; Mou, X.-Q.; Zhang, D. FSIM: A Feature Similarity Index for Image Quality Assessment. *IEEE Trans. Image Process.* **2011**, *20*, 2378–2386. [\[CrossRef\]](#) [\[PubMed\]](#)

-
57. Wang, P.; Wang, Y.-L.; Huang, B.; Wang, L.-G.; Zhang, X.-W.; Leung, H.; Chanussot, J. Poissonian Blurred Hyperspectral Imagery Denoising Based on Variable Splitting and Penalty Technique. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5505414. [[CrossRef](#)]
 58. Palud, P.; Thouvenin, P.-A.; Chainais, P.; Bron, E.; Le Petit, F. Efficient Sampling of Non Log-Concave Posterior Distributions with Mixture of Noises. *IEEE Trans. Signal Process.* **2023**, *71*, 2491–2501. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.