Hyperspectral Denoising Using Unsupervised Disentangled Spatio-Spectral Deep Priors

Yu-Chun Miao, Xi-Le Zhao*, Xiao Fu*, Jian-Li Wang, and Yu-Bang Zheng

***Abstract*—Image denoising is often empowered by accurate prior information. In recent years, data-driven neural network priors have shown promising performance for RGB natural image denoising. Compared to classic handcrafted priors (e.g., sparsity and total variation), the "deep priors" are learned using a large number of training samples—which can accurately model the complex image generating process. However, data-driven priors are hard to acquire for hyperspectral images (HSIs) due to the lack of training data. A remedy is to use the so-called *unsupervised deep image prior* (DIP). Under the unsupervised DIP framework, it is hypothesized and empirically demonstrated that *proper* neural network *structures* are reasonable priors of certain types of images, and the network weights can be learned without training data. Nonetheless, the most effective unsupervised DIP structures were proposed for natural images instead of HSIs. The performance of unsupervised DIP-based HSI denoising is limited by a couple of serious challenges, namely, network structure design and network complexity. This work puts forth an unsupervised DIP framework that is based on the classic spatio-spectral decomposition of HSIs. Utilizing the so-called *linear mixture model* of HSIs, two types of unsupervised DIPs, i.e., U-Net-like network and fully-connected networks, are employed to model the abundance maps and endmembers contained in the HSIs, respectively. This way, empirically validated unsupervised DIP structures for natural images can be easily incorporated for HSI denoising. Besides, the decomposition also substantially reduces network complexity. An efficient alternating optimization algorithm is proposed to handle the formulated denoising problem. Simulated and real data experiments are employed to showcase the effectiveness of the proposed approach.**

***Index Terms*—Hyperspectral image denoising, unsupervised deep image prior, spatio-spectral decomposition**

I. Introduction

HYPERSPECTRAL images (HSIs) contain rich spectral and spatial information of areas/objects of interest. HSIs have been widely used across many disciplines, e.g., biology, ecology, geoscience, and food/medicine science [1]. However, the acquired HSIs are often corrupted by various types of noise. Heavy noise may affect the performance of downstream analytical tasks (e.g., hyperspectral pixel classification). In the past two decades, a plethora of HSI denoising techniques were proposed to address this challenge; see [2]–[5].

At a high level, the idea of many HSI denoising methods is to fit the acquired image using an estimated image with prior information-induced priors. The rationale is that noise does not obey the HSI priors, and thus such a fitting process can effectively extract the "clean" HSI from the noisy version. Under this principle, early HSI denoising methods used spatial priors such as sparsity [6]–[8] and total variation (TV) [9]. Methods that exploit spectral priors were also proposed; see [10]–[12]. A number of denoising methods incorporated with *implicit* priors such as low matrix/tensor rank that is a result of multi-dimensional correlations; some examples can be found in [2]–[4], [13]–[18].

More recently, data-driven priors have drawn much attention in the vision and imaging communities [19]. In a nutshell, deep neural networks are used to learn a generative model of images from a large number of training samples. Deep generative models have been successful in computer vision, see, e.g., [20]–[22]. In particular, these models are able to map low-dimensional random vectors to visually authentic images—which means that they capture the essence of the image generating process. Hence, the learned generative network is naturally a good prior of clean images. This idea has also been used in HSI denoising; see, e.g., [23]–[27].

Although the methods mentioned above have attained satisfactory results for HSI denoising, these models' expressive ability is limited by the training data's adversity and quantity. That is, there is a lack of training data for HSIs [28]. This is because HSIs are, in general, much more costly to acquire relative to natural RGB images. In addition, different hyperspectral sensors often admit largely diverse specifications (e.g., the frequency band used, the spectral resolution, and the spatial resolution)—data acquired from one sensor may not be useful for training deep priors for images from other sensors.

Recently, Ulyanov *et al.* proposed an unsupervised image restoration framework, namely, *deep image prior* (DIP) [29]. DIP directly learns a generator network from a *single* noisy image—instead of learning the generator from a large number of training samples. The work in [29] showed that proper deep neural network architectures, without training on any samples, can already "encode" much critical information in the natural image generating process. This discovery has helped design *unsupervised* DIPs for tasks such as image

*Corresponding authors: Xi-Le Zhao and Xiao Fu.

This research of Xi-Le Zhao is supported by NSFC (No. 61876203, 61772003), the Key Project of Applied Basic Research in Sichuan Province (No. 2020YJ0216), the Applied Basic Research Project of Sichuan Province (No. 2021YJ0107) and National Key Research and Development Program of China (No. 2020YFA0714001). The work of X. Fu is supported by NSF ECCS-2024058 and NSF ECCS-1808159.

Y.-C. Miao, X.-L. Zhao, and J.-L. Wang are with the Research Center for Image and Vision Computing, School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu 611731, P.R.China (e-mails: szmyc1@163.com; xlzhao122003@163.com; wangjianli_123@163.com).

X. Fu is with the School of Electrical Engineering and Computer Science, Oregon State University (OSU), Corvallis, OR 97331, United States (e-mail: xiao.fu@oregonstate.edu).

Y.-B. Zheng is with the School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu 611731, China, and also with the Tensor Learning Team, RIKEN Center for Advanced Intelligence Project, Tokyo 103-0027, Japan (e-mail: zhengyubang@163.com).

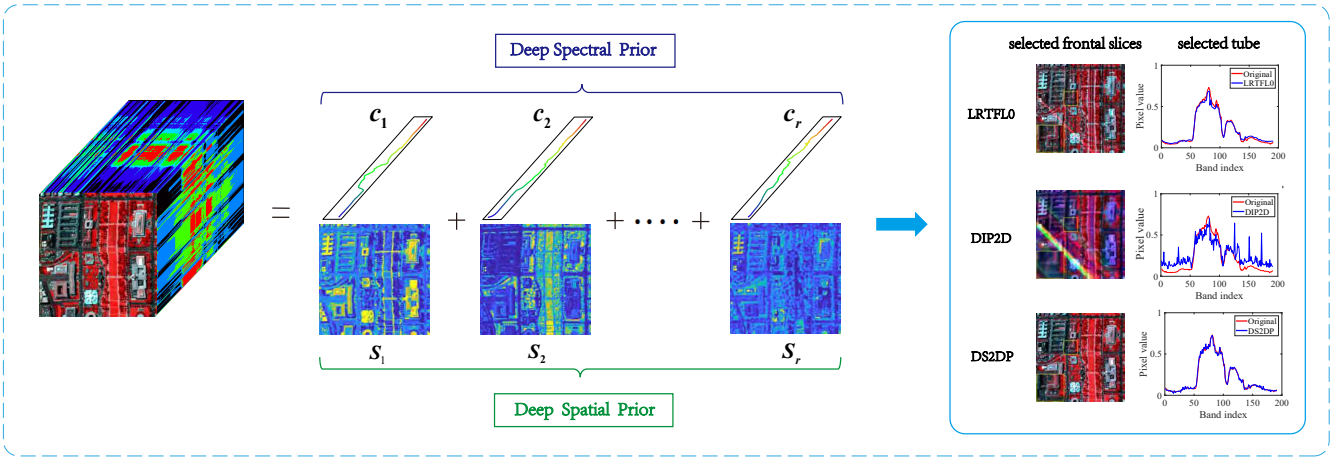


Fig. 1. The LMM for HSI and the proposed unsupervised disentangled spatio-spectral deep priors (DS2DP).

denoising, inpainting, and super-resolution. This work has thus attracted much attention. Since the DIP approach does not use any training data, it is particularly suitable for data-starved applications like hyperspectral imaging. Indeed, Sidorov *et al.* [30] extended the DIP idea to HSI denoising and observed positive results.

Nonetheless, capitalizing on the power of DIP for HSI denoising still faces a series of challenges. Unlike RGB images that only have three spectral channels, HSIs are often measured over hundreds of spectral channels. Therefore, directly using the DIP method that is originally proposed for RGB images to handle HSIs may not be as promising. First, it is unclear if the network structures used in [29] are still effective for HSIs. Second, due to the large size of HSIs, the scalability challenge is much more severe compared to the natural image cases. Indeed, as one will see in Sec. V, the two neural network structures used in [30] for modeling the generator of a standard HSI induce 2.150 and 2.342 million parameters, respectively—which makes the learning process challenging. Third, due to the special data acquisition process of HSIs, outlying pixels and structured noise (other than Gaussian noise) often arise. The DIP denoising loss function used in [29], [30] did not take these aspects into consideration.

Contributions. In this work, our interest lies in an unsupervised DIP-based denoising framework tailored for HSIs. Our detailed contributions are summarized as follows:

- **Disentangled Spatio-Spectral Deep Prior for HSI.** We propose an unsupervised DIP structure that is inspired by the well-established *linear mixture model* (LMM) for HSIs [31]; see Fig. 1. The LMM views every hyperspectral pixel as a linear combination of spectral signatures of a number of materials (*endmembers*). The linear combination coefficients of different endmembers across the image give rise to the *abundance maps* of the endmembers [32]. Using LMM, the spatial and spectral information embedded in the HSI can be "disentangled". This way, the spectral and spatial priors can be designed and modeled *individually*. As a result, the modeling and computational complexities can be substantially reduced—which often leads to improved accuracy. By our design, empirically validated unsupervised DIP structures for natural images can be much more easily capitalized for HSI denoising.
- **Structured Noise-robust Optimization.** We propose a training loss that models the structured noise (e.g., stripe-shaped or deadlines) as sparse outliers. We use an alternating optimization process to handle the formulated structured-noise robust deep prior-based denoising method, and admits simple lightweight updates.
- **Extensive Experiments.** We test the proposed approach on a large variety of simulated and real datasets. The experiments support our design—we observe substantially improved denoising performance relative to classic methods and more recent neural prior-based methods over all the datasets under test. In particular, due to our disentangled network design, the proposed method outperforms the existing unsupervised DIP-based HSI denoising methods in [30] in terms of both accuracy and memory/computational efficiency.

Notation. A scalar, a vector, a matrix, and a tensor are denoted as x, $\boldsymbol{x}$, $\boldsymbol{X}$, and $\underline{\boldsymbol{X}}$, respectively. $[\boldsymbol{x}]_i$, $[\boldsymbol{X}]_{i,j}$, and $[\underline{\boldsymbol{X}}]_{i,j,k}$ denote the i-th, (i,j)-th, and (i,j,k)-th element of $\boldsymbol{x} \in \mathbb{R}^I$, $\boldsymbol{X} \in \mathbb{R}^{I\times J}$, and $\underline{\boldsymbol{X}} \in \mathbb{R}^{I\times J\times K}$, respectively. The Frobenius norms of $\boldsymbol{X}$ and $\underline{\boldsymbol{X}}$ are denoted as $\|\boldsymbol{X}\|_F = \sqrt{\sum_{i,j}[\boldsymbol{X}]_{i,j}^2}$ and $\|\underline{\boldsymbol{X}}\|_F = \sqrt{\sum_{i,j,k}[\underline{\boldsymbol{X}}]_{i,j,k}^2}$, respectively. Given $\boldsymbol{y} \in \mathbb{R}^N$ and a matrix $\boldsymbol{X} \in \mathbb{R}^{I\times J}$, the outer product is defined as $\boldsymbol{X} \circ \boldsymbol{y}$. In particular, $\boldsymbol{X}\circ\boldsymbol{y} \in \mathbb{R}^{I\times J\times N}$ and $[\boldsymbol{X}\circ\boldsymbol{y}]_{i,j,n} = [\boldsymbol{X}]_{i,j}[\boldsymbol{y}]_n$. The matrix unfolding operator for a tensor is defined as $\mathrm{mat}(\underline{\boldsymbol{X}})$, which denotes the mode-3 unfolding of $\underline{\boldsymbol{X}}$ (see details of the unfolding of HSI in [33]). The $\mathrm{vec}(\boldsymbol{X})$ operator represents $\mathrm{vec}(\boldsymbol{X}) = [[\boldsymbol{X}]_{:,1}^T, \ldots, [\boldsymbol{X}]_{:,J}^T]^T$.

II. Preliminaries

In this section, we briefly review pertinent background information.

A. HSI Denoising

The acquired HSIs are three-dimensional arrays (i.e., tensors [34]). Denote $\underline{\boldsymbol{X}} \in \mathbb{R}^{I\times J\times K}$ as the HSI captured by a remotely deployed hyperspectral sensor, where $I \times J$ is the number of pixels presenting in the 2D spatial domain, and K is the number of spectral bands. Unlike natural images that are measured with the R, G, and B channels (i.e., $K = 3$), HSIs are measured over tens or hundreds of frequency bands, depending on the specifications of the employed sensors.

In general, $\underline{\boldsymbol{X}}$ is a noise-contaminated version of the underlying "clean" HSI (denoted by $\underline{\boldsymbol{X}}_\natural$). There are many factors contributing to noise in the hyperspectral acquisition process, i.e., thermal electronics, dark current, and stochastic error of photon-counting. If the noise is additive, we have

$$\underline{\boldsymbol{X}} = \underline{\boldsymbol{X}}_\natural + \underline{\boldsymbol{V}}, \tag{1}$$

where $\underline{\boldsymbol{V}} \in \mathbb{R}^{I\times J\times K}$ denotes the noise. The objective of HSI denoising is to "extract" $\underline{\boldsymbol{X}}_\natural$ from $\underline{\boldsymbol{X}}$.

B. Prior-Regularization Based HSI Denoising

Note that even under the additive noise model in (1), this problem is ill-posed—this is essentially a disaggregation problem which admits an infinite number of solutions. To overcome such ambiguity, prior information of the HSI is used to confine the solution space. A generic formulation can be summarized as follows:

$$\widehat{\underline{\boldsymbol{X}}} = \arg\min_{\underline{\boldsymbol{M}}} \ \|\underline{\boldsymbol{X}} - \underline{\boldsymbol{M}}\|_F^2 + \lambda R\left(\underline{\boldsymbol{M}}\right), \tag{2a}$$

$$\text{subject to } \underline{\boldsymbol{M}} \in \mathcal{M}, \tag{2b}$$

where $\widehat{\underline{\boldsymbol{X}}}$ denotes the estimate for $\underline{\boldsymbol{X}}_\natural$ using the above estimator, $\underline{\boldsymbol{M}}$ represents the optimization variable, $\mathcal{M}$ and $R(\cdot) : \mathbb{R}^{I\times J\times K} \to \mathbb{R}_+$ are the constraint set and regularization function imposed according to prior knowledge about the clean image $\underline{\boldsymbol{X}}_\natural$, respectively, and $\lambda \geq 0$ is the regularization parameter that balances the data fidelity term (i.e., the first term in (2a)) and the regularization.

1) From Analytical Priors to Data-Driven Priors: A variety of regularization/constraints have been considered in the literature. For example, in [2], [35],

$$R(\cdot) = \|\cdot\|_{\text{TV}}$$

is the TV across the two spatial dimensions, since image data exhibits certain slow changing properties over the space. In [36], [37], $\mathcal{M}$ represents the nonnegative orthant, since HSIs are always nonnegative. In [13], [38]–[42], low tensor and matrix rank constraints are added to $\underline{\boldsymbol{M}}$ through low-rank parameterization, respectively. Such prarameterization-based regularization can be written as

$$\widehat{\boldsymbol{z}} = \arg\min_{\boldsymbol{z}} \ \|\underline{\boldsymbol{X}} - \mathcal{G}\left(\boldsymbol{z}\right)\|_F^2, \tag{3}$$

where $\mathcal{G} : \mathbb{R}^N \to \mathbb{R}^{I\times J\times K}$ is a pre-specified parameterization function that represents the $I\times J\times K$ HSI using N parameters, i.e., $\boldsymbol{z}$, and $\mathcal{G}\left(\boldsymbol{z}\right)$ represents the estimation for the underlying clean HSI generated by $\mathcal{G}$ with parameters $\boldsymbol{z}$. For example, if $\text{mat}\left(\underline{\boldsymbol{X}}\right)$ is believed to be a low-rank matrix, $\text{mat}\left(\mathcal{G}\left(\boldsymbol{z}\right)\right) = \boldsymbol{A}\boldsymbol{B}^T$ and $\boldsymbol{z} = [\text{vec}(\boldsymbol{A})^T, \text{vec}(\boldsymbol{B})^T]^T$. After estimating the parameters $\boldsymbol{z}$, the clean image can be simply estimated via

$$\widehat{\underline{\boldsymbol{X}}} = \mathcal{G}(\widehat{\boldsymbol{z}}).$$

Classic priors are useful but often insufficient to capture the complex nature of the underlying structure of HSIs.

A number of works used deep neural networks to parameterize the regularization—i.e., these works use a deep neural network $\mathcal{G}_{\boldsymbol{\theta}}(\cdot) : \mathbb{R}^N \to \mathbb{R}^{I\times J\times K}$ whose network weights are collected in $\boldsymbol{\theta} \in \mathbb{R}^D$ to act as the regularization in (2a) [23]–[27]. Instead of having an analytical expression, such regularizers are "trained" using a large number of training samples. As deep neural networks are universal function approximators, such learned "deep priors" are believed to be able to approximate complex generative processes of HSIs and thus are more effective priors for denoising.

$$\widehat{\boldsymbol{z}} = \arg\min_{\boldsymbol{z}} \ \|\underline{\boldsymbol{X}} - \mathcal{G}_{\boldsymbol{\theta}}\left(\boldsymbol{z}\right)\|_F^2, \tag{4}$$

However, unlike natural RGB images that have tens of thousands of training samples for learning $\mathcal{G}_{\boldsymbol{\theta}}$, HSI (especially remotely sensed HSI) datasets are relatively rare due to their costly acquisition process. Without a large amount of (diverse) HSIs, training such a regularizer may be out of reach.

2) Unsupervised Deep Image Prior: Very recently, Ulyanov *et al.* proposed the so-called DIP [29] to circumvent the lack of training samples. The major discovery in [29] is that a proper neural network *architecture* (without knowing the neural network weights $\boldsymbol{\theta}$) can already encode much prior information of images. As a result, tasks such as image denoising can be done by learning a neural network $\mathcal{G}_{\boldsymbol{\theta}}(\boldsymbol{z})$ to fit $\underline{\boldsymbol{X}}$ with a *random* but *known* $\boldsymbol{z}$.

With this idea, the denoising problem can be formulated as follows:

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \ \|\underline{\boldsymbol{X}} - \mathcal{G}_{\boldsymbol{\theta}}\left(\boldsymbol{z}\right)\|_F^2, \tag{5}$$

and the denoised image can be estimated via

$$\widehat{\underline{\boldsymbol{X}}} = \mathcal{G}_{\widehat{\boldsymbol{\theta}}}\left(\boldsymbol{z}\right). \tag{6}$$

The idea of DIP is quite different compared to the supervised deep prior-based approaches such as those in [23]–[26] [cf. Eq. (4)]. In DIP, the network weights $\boldsymbol{\theta}$ is learned from a single degraded image in an unsupervised manner, and $\boldsymbol{z}$ is given instead of learned.

At first glance, it may be surprising that an untrained neural network can be used for image denoising (and also inpainting and super-resolution as revealed in [29]). The key rationale behind this approach may be understood as follows: First, some carefully designed neural network structures (e.g., convolutional neural network with proper modifications) are able to capture much information in the generating process of some types of images of interest. That is, *not all* neural network structures could work well for all types of images. Different structures may need to be carefully *handcrafted* for different types of images. The handcrafted neural network structure is analogous to the handpicked priors such as the L_1 norm, Tikhonov regularization, and TV regularization—which are also not learned from training samples. In the

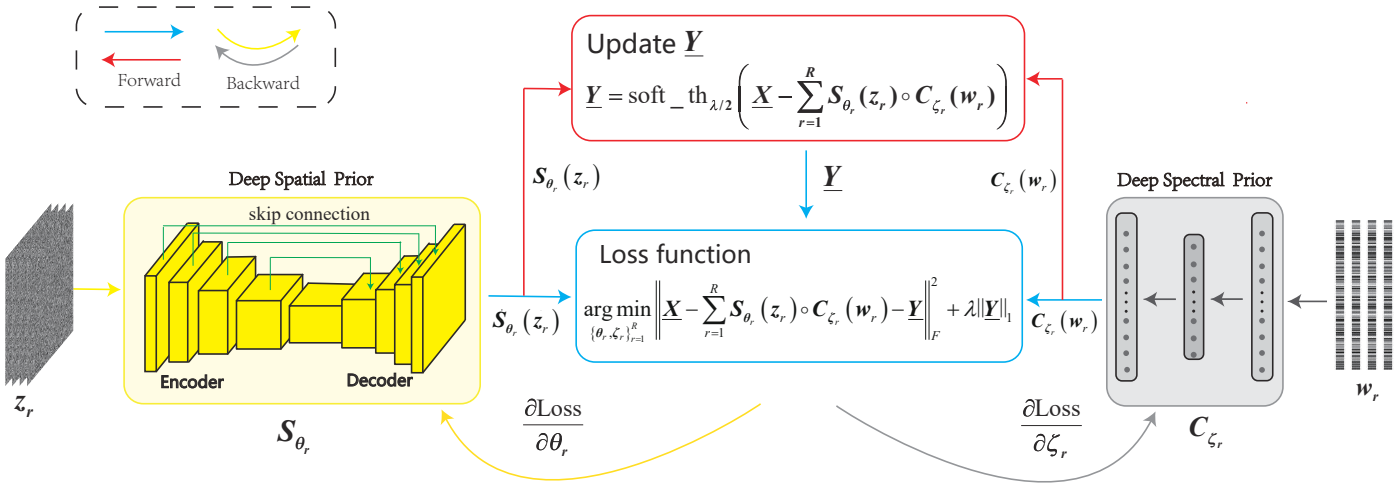


Fig. 2. Illustration of the proposed DS2DP. The generative networks $\mathcal{C}_{\zeta_r}$ and $\mathcal{S}_{\theta_r}$ are applied to capture the deep spectral prior of the spectral signatures and the deep spatial prior of the abundance matrices, respectively.

original paper [29], the U-Net-like "hourglass" architecture was shown to be powerful in natural RGB image restoration tasks under the DIP framework. In [30], various network structures (namely, DIP2D and DIP3D) were experimented for HSI denoising—and the results can be quite different, as one will also see in Sec. IV. Second, in image restoration tasks, the degraded (noisy) $\underline{\boldsymbol{X}}$ still contains much information in the underlying image. Hence, the fitting loss in (5) also "forces" the $\mathcal{G}_{\boldsymbol{\theta}}$ to faithfully capture the essential information in $\underline{\boldsymbol{X}}$. In particular, since $\mathcal{G}_{\boldsymbol{\theta}}$ has a structured underlying generative process (by construction), the learned $\mathcal{G}_{\boldsymbol{\theta}}$ is more likely to capture the "structured signal part" (i.e., the clean image $\underline{\boldsymbol{X}}_{\natural}$) in $\underline{\boldsymbol{X}}$ other than the random noise part.

Since the DIP procedure does not use any training examples, it is particularly attractive to data-starved applications such as hyperspectral imaging. In addition, although it involves careful structure handcrafting, DIP still inherits many good properties of neural networks, e.g., being capable of modeling complex generative processes. Consequently, it often exhibits more appealing image restoration performance compared to classic regularizer/parameterization based methods (e.g., TV and low matrix/tensor rank); see [29], [30].

C. Challenges

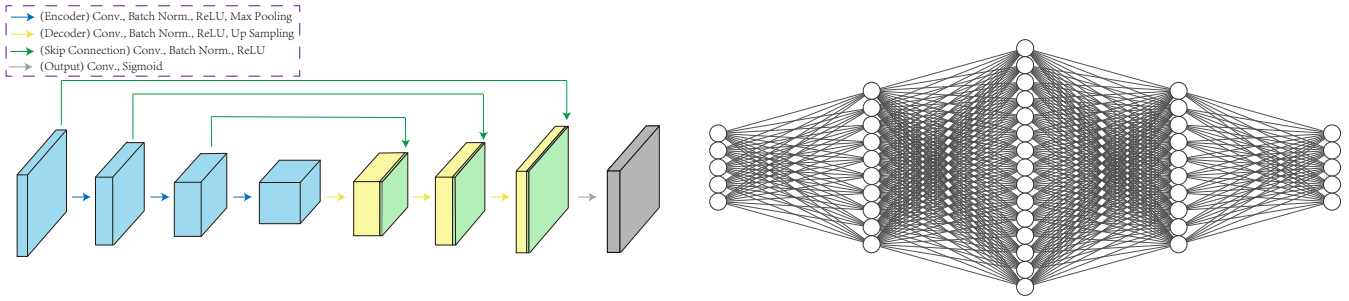
The unsupervised DIP-based approaches are attractive since they are effective without using any training data. However, finding a proper network structure to serve as prior of HSIs and learning the corresponding $\boldsymbol{\theta}$ is by no means a trivial task. A couple of notable new challenges that arise in the domain of hyperspectral imaging are as follows:

1) Challenge - 1 Integrating Unsupervised DIP and HSI: Since HSIs are quite different compared to natural RGB images (in terms of sensors, sensing processes, resolutions, and frequency bands used), directly using the neural network structure in [29] in hyperspectral imaging may not be best practice. The work in [30] proposed two structures crafted for this, but it is not clear if these two structures are "optimal" due to the lack of extensive experiments. In fact, as we will show in Sec. IV, these two unsupervised DIP structures are sometimes not as promising as some classic models (e.g., low-rank tensor decomposition-based denoising) in terms of denoising performance. Hence, the first challenge lies in if we could *circumvent* designing a new unsupervised DIP network architecture from scratch—which could involve much trial-and-error and time/resource consuming. In particular, can we leverage some underlying structures of the HSIs to avoid exhaustively searching through ad-hoc DIP architectures, but utilize some existing DIP network structures (e.g., those in [29]) to effectively assist our HSI denoising task? We will answer this question.

2) Challenge - 2 Network Size: Another challenge that arises in unsupervised DIP-based HSI denoising is that the HSIs are large-scale images due to the large number of spectral bands contained in the pixels. Directly modeling the generative process of a large-scale 3D image (or a third-order tensor) inevitably leads to an overly sized neural network $\mathcal{G}_{\boldsymbol{\theta}}$. Although the work in [30] employed a number of tricks for network size reduction, the final constructions still yield a large number of network parameters. This leads to a computationally heavy optimization problem [cf. Eq. (5)]. Since the problem is already nonconvex and challenging, the excessive scale of the optimization problem only makes the denoising procedure less efficient. The challenging nature of numerical optimization may also affect the denoising performance since "bad" local minima may be easier to happen.

III. Proposed Approach

To circumvent the challenges, we will leverage the well-established LMM of HSI to come up with our customized unsupervised DIPs in the next section. As will be seen, using the LMM to disentangle the spatial and spectral modalities of the HSIs allows us to use well-established/simple DIP structures to model each modality, which spares the agnostic pain of searching for a new DIP to model the high-dimensional



(a) (b)

Fig. 3. The detailed network structures. The two figures conrespond to: (a) the unsupervised DIP for abundance maps; and (b) the unsupervised DIP for endmembers

hyperspectral data. The disentanglement also effectively reduces the model complexity. To this end, we briefly review the main idea of LMM.

A. Linear Mixture Model of HSI

The LMM of $\underline{\boldsymbol{X}}$ is as follows (when the noise is absent):

$$\underline{\boldsymbol{X}} = \sum_{r=1}^{R} \boldsymbol{S}_r \circ \boldsymbol{c}_r, \tag{7}$$

where $\boldsymbol{S}_r \in \mathbb{R}^{I\times J}$ and $\boldsymbol{c}_r \in \mathbb{R}^{K}$ represent the r-th endmember's abundance map and the spectral signature, respectively, and R is the number of endmembers contained in the HSI. The LMM can also be expressed as

$$[\underline{\boldsymbol{X}}]_{i,j,k} = \sum_{r=1}^{R} [\boldsymbol{S}_r]_{i,j}[\boldsymbol{c}_r]_k;$$

see [1], [31]. Physically, it means that every pixel is a non-negative combination of the spectral signatures of the constituting endmembers in the HSI. Note that

$$\boldsymbol{S}_r \geq \boldsymbol{0},\ \boldsymbol{c}_r \geq \boldsymbol{0}$$

according to their physical meanings—and thus the model in (7) is often related to non-negative matrix factorization (NMF) [43]. An illustration of the LMM can be found in Fig. 1. The LMM model with a relatively small R can often capture around 98% of the energy of the HSI [44]. Hence, it is a reliable model for HSIs. Indeed, the LMM has been utilized for a large variety of hyperspectral imaging tasks, e.g., hyperspectral unmixing [1], [32], [45]–[48], hyperspectral super-resolution [49], pansharpening [50], compression and recovery [51], and denoising [52], just to name a few. In this work, we propose to use the LMM to help design unsupervised DIP neural network structures and denoising algorithms.

B. LMM-Aided Unsupervised DIP for HSI

Notably, the LMM disentangles the spectral and spatial information into two sets of latent factors, i.e., $\{\boldsymbol{S}_r\}_{r=1}^R$ and $\{\boldsymbol{c}_r\}_{r=1}^R$. Our motivations for using the LMM representation to design unsupervised DIP for HSIs are as follows:

- First, LMM disentanglement allows using known effective DIP structures for natural images for HSI. The physical meaning of the latent factors entails the opportunity to employ known effective neural network structures of unsupervised DIP. The abundance matrix $\boldsymbol{S}_r$ can be understood as how the material r spreads over space. The hypothesis is that the abundance maps exhibit similar properties to natural images that focus on capturing and conveying spatial information. Under this hypothesis, it is reasonable to use unsupervised DIP neural network structures that are known to work well for natural images to model $\boldsymbol{S}_r$. Moreover, the $\boldsymbol{c}_r$ vector can be understood as the spectral signature of the r-th material, which is the variation of reflectance or emittance of material over different wavelengths. It is known that fully connected neural networks (FCNs) can approximate such relatively simple 1-D continuous smooth functions well.
- Second, LMM disentanglement effectively reduces network complexity. By disentanglement and LMM, the model size of the HSI is substantially reduced. Instead of directly imposing unsupervised DIP on the whole HSI, we employ two types of unsupervised DIPs (i.e., the deep spatial and spectral priors) to model abundance maps and spectral signatures, respectively. Since the number of endmembers is often not large, the computational complexity is substantially reduced.

Following the above argument, we model the HSI using the following:

$$\underline{\boldsymbol{X}} = \sum_{r=1}^{R} \mathcal{S}_{\boldsymbol{\theta}_r}(\boldsymbol{z}_r) \circ \mathcal{C}_{\boldsymbol{\zeta}_r}(\boldsymbol{w}_r), \tag{8}$$

where $\mathcal{S}_{\boldsymbol{\theta}_r}(\cdot) : \mathbb{R}^{N_a} \to \mathbb{R}^{I\times J}$ is the unsupervised DIP neural network of the r-th endmember's abundance map, and $\boldsymbol{\theta}_r$ collects all the corresponding network weights; similarly, $\mathcal{C}_{\boldsymbol{\zeta}_r}(\cdot) : \mathbb{R}^{N_s} \to \mathbb{R}^{K}$ and $\boldsymbol{\zeta}_r$ denote the unsupervised DIP of the r-th endmember and its corresponding network weights, respectively; the vectors $\boldsymbol{z}_r \in \mathbb{R}^{N_a}$ and $\boldsymbol{w}_r \in \mathbb{R}^{N_s}$ are low-dimensional random vectors that are responsible for generating the r-th abundance map and endmember, respectively. Our detailed design for $\mathcal{S}_{\boldsymbol{\theta}_r}$ and $\mathcal{C}_{\boldsymbol{\zeta}_r}$ are as follows:

1) Unsupervised DIP for Abundance Maps: As mentioned, the abundance maps capture the spatial information of the corresponding materials. We propose to employ the U-Net-like "hourglass" architecture in [29] for modeling $\mathcal{S}_{\boldsymbol{\theta}_r}$. Note that this network architecture was shown to be able to capture the spatial prior of nature images. The U-Net is an asymmetric

autoencoder [53] with skip connections, whose structure is shown in Fig. 3 (left).

2) Unsupervised DIP for Endmembers: The endmembers are relatively simple to model—since they can be understood as one-dimensional smooth functions. Hence, we employ FCNs as the unsupervised DIP for $\mathcal{C}_{\boldsymbol{\zeta}_r}$. We use FCNs with three layers; also see Fig. 3 (right).

Besides the above unsupervised DIP design, in this work, we also take into consideration of impulsive noise and grossly corrupted pixels (outliers) that often arise in HSIs. Unlike natural images whose sensing environment can be well controlled, remotely sensed HSIs often suffer from heavily corrupted pixels or spectral bands due to various reasons; see [39], [40]. If not accounted for, the HSI denoising performance could be severely hindered by such noise. To this end, we consider a noisy data acquisition model as follows:

$$\underline{\boldsymbol{X}} = \underbrace{\sum_{r=1}^{R} \mathcal{S}_{\boldsymbol{\theta}_r}(\boldsymbol{z}_r) \circ \mathcal{C}_{\boldsymbol{\zeta}_r}(\boldsymbol{w}_r)}_{\underline{\boldsymbol{X}}_\natural} + \underline{\boldsymbol{Y}} + \underline{\boldsymbol{V}}, \tag{9}$$

where $\underline{\boldsymbol{V}}$ represents ubiquitous noise, e.g., the Gaussian noise, and $\underline{\boldsymbol{Y}}$ denotes the impulsive noise or outliers. Accordingly, We propose the following denoising criterion:

$$\underset{\{\boldsymbol{\theta}_r,\boldsymbol{\zeta}_r\}_{r=1}^R}{\arg\min} \left\| \underline{\boldsymbol{X}} - \sum_{r=1}^{R} \boldsymbol{S}_{\boldsymbol{\theta}_r}(\boldsymbol{z}_r) \circ \boldsymbol{C}_{\boldsymbol{\zeta}_r}(\boldsymbol{w}_r) - \underline{\boldsymbol{Y}} \right\|_F^2 + \lambda \|\underline{\boldsymbol{Y}}\|_1, \tag{10}$$

where $\lambda \geq 0$ and $\|\underline{\boldsymbol{Y}}\|_1 = \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K} |[\underline{\boldsymbol{Y}}]_{i,j,k}|$ is used for imposing the sparsity prior on $\underline{\boldsymbol{Y}}$, since outliers happen sparsely.

C. Optimization Algorithm

Let us denote the objective function in (10) using the following shorthand notation:

$$\underset{\{\boldsymbol{\theta}_r,\boldsymbol{\zeta}_r\}_{r=1}^R,\underline{\boldsymbol{Y}}}{\arg\min} \ \mathsf{Loss}\left(\{\boldsymbol{\theta}_r,\boldsymbol{\zeta}_r\}_{r=1}^R, \underline{\boldsymbol{Y}}\right). \tag{11}$$

We propose the following algorithmic structure:

$$\{\boldsymbol{\theta}_r^{t+1},\boldsymbol{\zeta}_r^{t+1}\}_{r=1}^R \leftarrow \underset{\{\boldsymbol{\theta}_r,\boldsymbol{\zeta}_r\}_{r=1}^R}{\widetilde{\arg\min}} \ \mathsf{Loss}\left(\{\boldsymbol{\theta}_r,\boldsymbol{\zeta}_r\}_{r=1}^R, \underline{\boldsymbol{Y}}^t\right) \tag{12}$$

$$\underline{\boldsymbol{Y}}^{t+1} \leftarrow \underset{\underline{\boldsymbol{Y}}}{\arg\min} \ \mathsf{Loss}\left(\{\boldsymbol{\theta}_r^{t+1},\boldsymbol{\zeta}_r^{t+1}\}_{r=1}^R, \underline{\boldsymbol{Y}}\right), \tag{13}$$

where the superscript "t" is the iteration index. In (12), we use $\widetilde{\min}$ to denote *inexact* minimization since exactly solving the subproblem w.r.t. the network parameters may not be possible—due to its large size and nonconvexity.

1) Solution for (12): Note that the subproblem w.r.t. $\{\boldsymbol{\theta}_r,\boldsymbol{\zeta}_r\}_{r=1}^R$ is nothing but a regression problem using neural models. Hence, any off-the-shelf neural network optimizer can be employed for updating $\{\boldsymbol{\theta}_r,\boldsymbol{\zeta}_r\}_{r=1}^R$. In this work, we use the (sub-)gradient descent[1] algorithm with momentum that has been proven effective in complex network learning problems [54]:

$$\boldsymbol{\theta}_r^{t+1} \leftarrow \boldsymbol{\theta}_r^t - \alpha^t \nabla_{\boldsymbol{\theta}_r} \ \mathsf{Loss}\left(\{\boldsymbol{\theta}_r,\boldsymbol{\zeta}_r^t\}_{r=1}^R, \underline{\boldsymbol{Y}}^t\right) \tag{14a}$$

$$\boldsymbol{\zeta}_r^{t+1} \leftarrow \boldsymbol{\zeta}_r^t - \alpha^t \nabla_{\boldsymbol{\zeta}_r} \ \mathsf{Loss}\left(\{\boldsymbol{\theta}_r^t,\boldsymbol{\zeta}_r\}_{r=1}^R, \underline{\boldsymbol{Y}}^t\right), \tag{14b}$$

for all $r = 1, \ldots, R$. Note that the gradient w.r.t. $\boldsymbol{\theta}_r$ and $\boldsymbol{\zeta}_r$ can be computed by the standard back-propagation algorithm [55]. Here, α^t is the step size (i.e., learning rate) of iteration t. There are multiple ways of determining α^t. In this work, we use the step size rule advocated in the Adam algorithm [54].

2) Solution for (13) : The subproblem (13) is convex—whose solution is the well-known soft-thresholding proximal operator [56]. Hence, the update of $\underline{\boldsymbol{Y}}$ can be expressed as

$$\underline{\boldsymbol{Y}}^{t+1} = \mathrm{soft\_th}_{\lambda/2}\left(\underline{\boldsymbol{X}} - \sum_{r=1}^{R} \widehat{\boldsymbol{S}}_r^{t+1} \circ \widehat{\boldsymbol{c}}_r^{t+1}\right). \tag{15}$$

where

$$\widehat{\boldsymbol{S}}_r^{t+1} = \mathcal{S}_{\boldsymbol{\theta}_r^{t+1}}(\boldsymbol{z}_r), \ \widehat{\boldsymbol{c}}_r^{t+1} = \mathcal{C}_{\boldsymbol{\zeta}_r^{t+1}}(\boldsymbol{w}_r)$$

and $\mathrm{soft\_th}_{\lambda/2}(\cdot)$ applies soft-thresholding to every entry of its input, in which the entry-wise thresholding is defined as

$$\mathrm{soft\_th}_{\delta}(x) = \mathrm{sgn}(x)\max(|x| - \delta, 0). \tag{16}$$

Algorithm 1 DS2DP for HSI Denoising.

Input: the HSI $\underline{\boldsymbol{X}} \in \mathbb{R}^{I\times J\times K}$, the regularization parameter λ, and the number of endmembers R.

1: sample random $\boldsymbol{z}_r$ and $\boldsymbol{w}_r$ from uniform distribution;
2: **for** $t = 1$ to T **do** (repeat until convergence)
3: $\quad \widehat{\boldsymbol{S}}_r = \mathcal{S}_{\boldsymbol{\theta}_r^{t-1}}(\boldsymbol{z}_r)$, $\widehat{\boldsymbol{c}}_r = \mathcal{C}_{\boldsymbol{\zeta}_r^{t-1}}(\boldsymbol{w}_r)$;
4: $\quad$ update $\boldsymbol{\theta}_r, \boldsymbol{\zeta}_r$ for all r; using the Adam [54];
5: $\quad$ update $\underline{\boldsymbol{Y}}$ according to (13);
6: **end for**
7: $\widehat{\underline{\boldsymbol{X}}} = \sum_{r=1}^{R} \widehat{\boldsymbol{S}}_r \circ \widehat{\boldsymbol{c}}_r$;

Output: the denoising HSI $\widehat{\underline{\boldsymbol{X}}}$.

The algorithm is summarized in Algorithm 1, which we name as the *unsupervised disentangled spatio-spectral deep prior* (DS2DP) algorithm. The algorithm falls into the category of inexact block coordinate descent [57]. Under some relatively mild conditions, the algorithm produces a solution sequence that converges to a stationary point of the optimization problem in (10); see detailed discussions in [57].

IV. Experiments

In this section, we use simulated and real data to demonstrate the effectiveness of the proposed approach.

A. Baselines

To thoroughly evaluate the performance of DS2DP, we implemented five state-of-the-art methods as the baselines. These methods include two unsupervised methods, i.e., *deep image prior based on 2D convolution* (DIP2D) [30] and *deep image prior based on 3D convolution* (DIP3D) [30], a matrix optimization-based method, i.e., *hyperspectral image restoration using low-rank matrix recovery* (LRMR) [38], and

[1]Since the ReLU activation functions used in the U-Net and the FCN are not differentiable at one point, the algorithm is subgradient based. Nonetheless, we use ∇ (usually for denoting gradient) to denote subgradient for notation simplicity.

TABLE I

QUANTITATIVE COMPARISON OF THE DENOISING RESULTS BY DIFFERENT METHODS. THE **BEST** AND SECOND BEST VALUES ARE HIGHLIGHTED IN BOLD AND UNDERLINED, RESPECTIVELY.

Case		Case 1			Case 2			Case 3			Case 4			Case 5			Case 6		
Dataset	Method	PSNR	SSIM	SAM	PSNR	SSIM	SAM	PSNR	SSIM	SAM	PSNR	SSIM	SAM	PSNR	SSIM	SAM	PSNR	SSIM	SAM
WDC Mall	DIP2D	30.408	0.871	0.122	26.540	0.770	0.163	24.043	0.708	0.228	22.679	0.678	0.271	23.366	0.696	0.227	21.759	0.594	0.282
	DIP3D	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
	LRMR	34.954	0.951	0.130	34.954	0.951	0.130	32.422	0.933	0.156	32.058	0.925	0.148	32.358	0.920	0.159	29.815	0.907	0.210
	LRTDTV	35.293	0.952	0.106	35.087	0.950	0.106	33.307	0.925	0.148	33.024	0.919	0.136	33.464	0.914	0.113	31.691	0.894	0.136
	LRTFL0	36.043	0.964	0.112	35.796	0.961	0.111	34.151	0.948	0.133	35.278	0.941	0.115	34.296	0.949	0.123	33.224	0.943	0.163
	DS2DP	**36.439**	**0.965**	**0.102**	**35.926**	**0.969**	**0.093**	**34.562**	**0.951**	**0.116**	**35.887**	**0.954**	**0.100**	**35.087**	**0.962**	**0.100**	**34.352**	**0.967**	**0.116**
Pavia Centre	DIP2D	31.965	0.897	0.068	29.603	0.876	0.072	25.319	0.758	0.186	23.587	0.728	0.232	24.885	0.768	0.164	22.175	0.551	0.180
	DIP3D	26.969	0.694	0.075	26.338	0.691	0.078	25.421	0.651	0.094	23.445	0.637	0.104	24.173	0.672	0.091	23.039	0.627	0.131
	LRMR	33.293	0.926	0.090	33.293	0.926	0.090	30.398	0.816	0.052	32.398	0.916	0.142	31.409	0.901	0.106	24.667	0.742	0.724
	LRTDTV	33.511	0.921	0.095	33.608	0.923	0.065	31.465	0.901	0.104	33.096	0.903	0.147	31.415	0.881	0.104	31.882	0.894	0.101
	LRTFL0	33.833	0.923	0.088	33.310	0.935	0.089	31.751	0.917	0.096	32.756	0.927	0.089	32.676	0.928	0.090	32.003	0.920	0.101
	DS2DP	**35.211**	**0.947**	**0.062**	**34.336**	**0.941**	**0.058**	**32.545**	**0.926**	**0.094**	**33.682**	**0.934**	**0.066**	**33.836**	**0.936**	**0.064**	**32.523**	**0.924**	**0.086**
Pavia University	DIP2D	33.103	0.852	0.107	25.818	0.770	0.177	25.157	0.727	0.223	24.047	0.714	0.269	24.024	0.719	0.283	21.549	0.574	0.382
	DIP3D	30.070	0.804	0.111	24.968	0.705	0.151	25.307	0.701	0.156	24.198	0.683	0.166	24.265	0.701	0.166	23.509	0.640	0.173
	LRMR	33.063	0.862	0.113	31.582	0.787	0.149	31.155	0.860	0.119	31.858	0.861	0.115	31.385	0.829	0.139	27.615	0.747	0.240
	LRTDTV	33.136	0.875	0.108	32.223	0.861	0.110	31.497	0.841	0.151	32.190	0.866	0.112	32.123	0.851	0.136	31.027	0.830	0.187
	LRTFL0	34.312	0.890	0.092	33.724	0.879	0.099	32.972	0.867	0.123	33.642	0.877	0.103	33.146	0.863	0.124	32.735	0.858	0.126
	DS2DP	**35.202**	**0.928**	**0.068**	**34.600**	**0.917**	**0.073**	**33.916**	**0.915**	**0.085**	**34.600**	**0.917**	**0.073**	**34.467**	**0.918**	**0.074**	**33.795**	**0.915**	**0.081**
CAVE	DIP2D	29.643	0.636	0.339	23.839	0.589	0.421	23.204	0.562	0.449	21.955	0.526	0.506	22.416	0.538	0.484	22.416	0.539	0.484
	DIP3D	28.960	0.709	0.332	23.397	0.571	0.447	23.377	0.566	0.449	22.157	0.534	0.471	22.435	0.549	0.460	21.405	0.509	0.501
	LRMR	30.633	0.661	0.418	30.633	0.661	0.418	27.724	0.607	0.466	31.809	0.807	0.334	29.015	0.680	0.445	26.404	0.659	0.536
	LRTDTV	35.529	0.883	0.165	34.769	0.877	0.210	32.792	0.843	0.260	34.036	0.862	0.232	31.779	0.772	0.361	31.063	0.773	0.430
	LRTFL0	33.241	0.877	0.233	33.191	0.891	0.262	32.978	0.846	0.209	33.743	0.852	0.264	32.139	0.781	0.352	30.956	0.855	0.301
	DS2DP	**36.043**	**0.923**	**0.142**	**35.603**	**0.907**	**0.146**	**33.892**	**0.956**	**0.165**	**35.682**	**0.914**	**0.155**	**32.775**	**0.862**	**0.187**	**32.588**	**0.848**	**0.213**

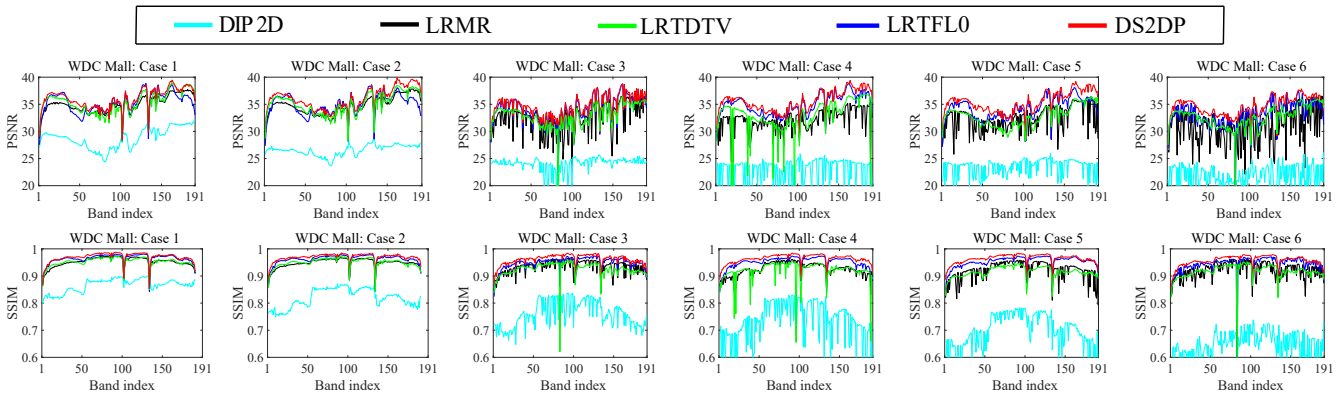


Fig. 4. PSNR and SSIM values of all bands obtained by different methods on HSI WDC Mall for Cases 1-6.

two tensor optimization-based methods, i.e., *TV-regularized low-rank tensor decomposition* (`LRTDTV`) [39] and *hyperspectral restoration via L_0 gradient regularized low-rank tensor factorization* (`LRTFL0`) [40].

For `DIP2D` and `DIP3D`, we set the maximum number of iterations to be 6,000 and report the best results during the iterations. For the proposed `DS2DP`, we set the maximum number of iterations to be 6,000 and report the results at the 6,000th iteration. For `LRMR`, `LRTDTV`, and `LRTFL0`, their parameters are set as suggested in [38]–[40]—with parameter fine-tuning effort to uplift its performance in some cases. The experiments of `DIP2D`, `DIP3D`, and `DS2DP` are executed using `Python`

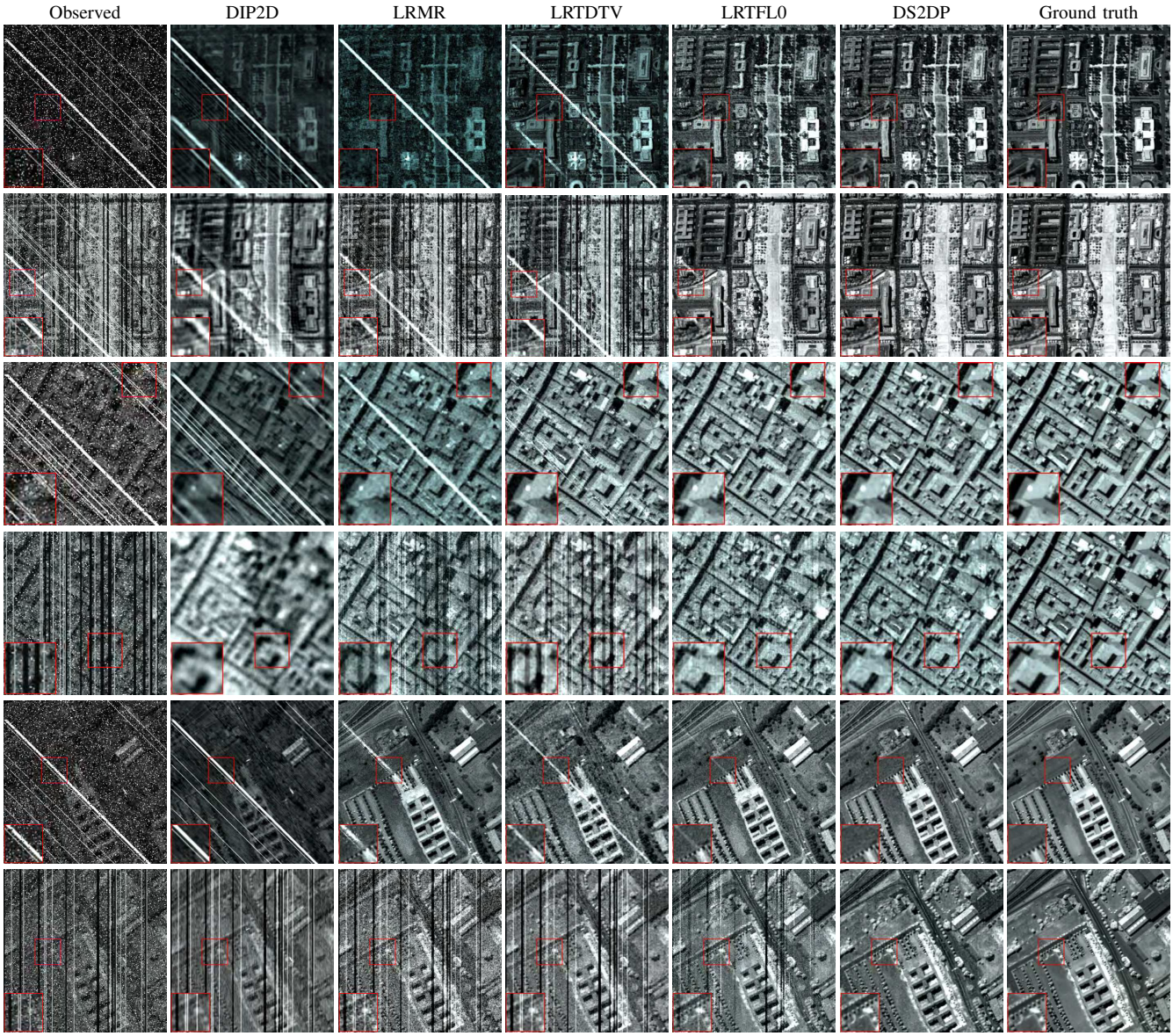


Fig. 5. Denoising results obtained by different methods. From left to right: the observed image, the denoising results by DIP2D, LRMR, LRTDTV, LRTFL0, DS2DP (proposed), and the ground truth, respectively. The first two rows are the denoising results on WDC Mall for Cases 4 and 6, respectively. The second two rows are the denoising results on Pavia Centre for Cases 4 and 6, respectively. The last two rows are the denoising results on Pacia University for Cases 4 and 6, respectively.

on a computer with a six-core Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz, 32.0 GB of RAM, and an NVIDIA GeForce RTX 2070 GPU. The experiments of LRMR, LRTDTV, and LRTFL0 are implemented in Matlab (2019a) on the same computer.

B. Simulated Data Experiments

Evaluation Metrics. We adopt three frequently used evaluation metrics, namely, peak signal-to-noise ratio (PSNR), structure similarity (SSIM), and spectral angle mapper (SAM) [40]. Generally, better-restored denoising performance is reflected by higher PSNR and SSIM values and lower SAM values.

Simulated Data. For simulated data, we use a number of HSIs to serve as our ground truth, which include Washington DC Mall (WDC Mall) [2] of size 256 × 256 × 191, Pavia Centre [2] of size 200 × 200 × 80 that is clipped into 192 × 192 × 80, and Pavia University [2] of size 256 × 256 × 87. The multispectral images (MSIs) in the CAVE dataset [3] of size 256 × 256 × 31 are also used to serve as our clean data $\underline{\boldsymbol{X}}_{\natural}$.

Scenarios. We consider a series of scenarios with various types of noise:

Case 1 *(Gaussian Noise)*: In this basic scenario, the i.i.d. zero-mean Gaussian noise is added to all bands with the variance set to be 0.1. The signal-to-noise ratios (SNRs) (see definition in [58]) associated with different datasets can be found in Table II. One can see the noise levels in different datasets are similar. Note that the HSIs with SNR being 6dB to 8dB are considered

[2] http://lesun.weebly.com/hyperspectral-data-set.html
[3] https://www.cs.columbia.edu/CAVE/databases/multispectral/

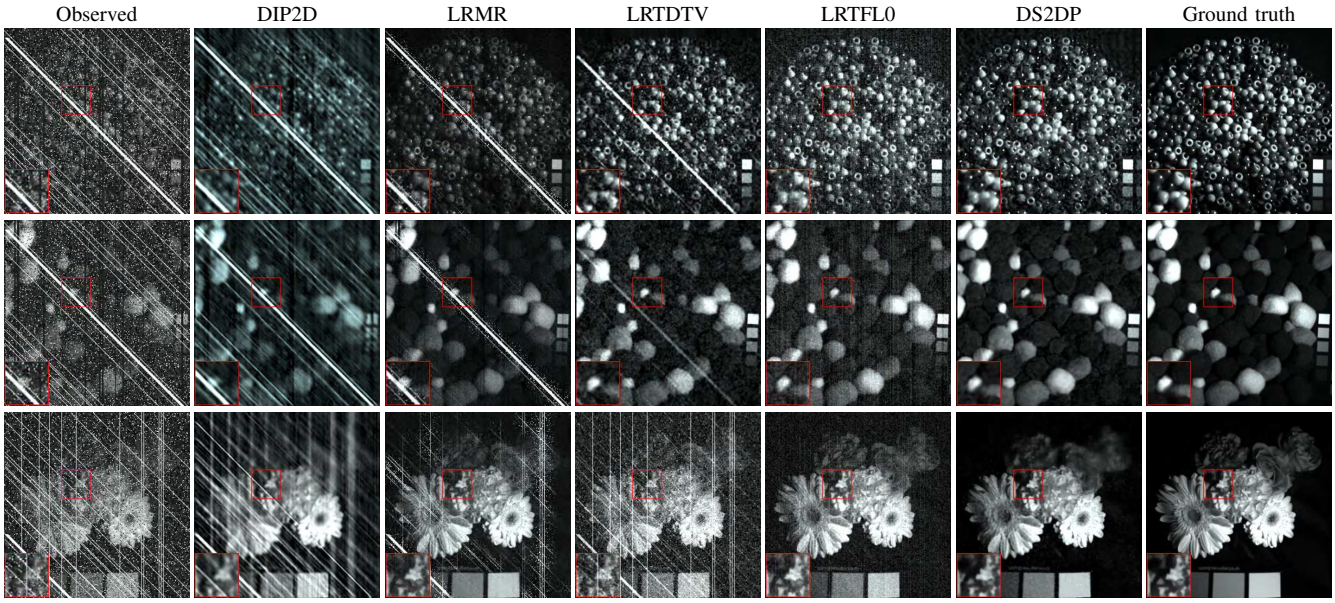


Fig. 6. Denoising results obtained by different methods for Case 6. From top to bottom: the band 4 of Beads, the band 4 of Pompoms, and the band 31 of Flowers, respectively. From left to right: the observed image, the denoising results by DIP2D, LRMR, LRTDTV, LRTFL0, DS2DP, and the ground truth, respectively.

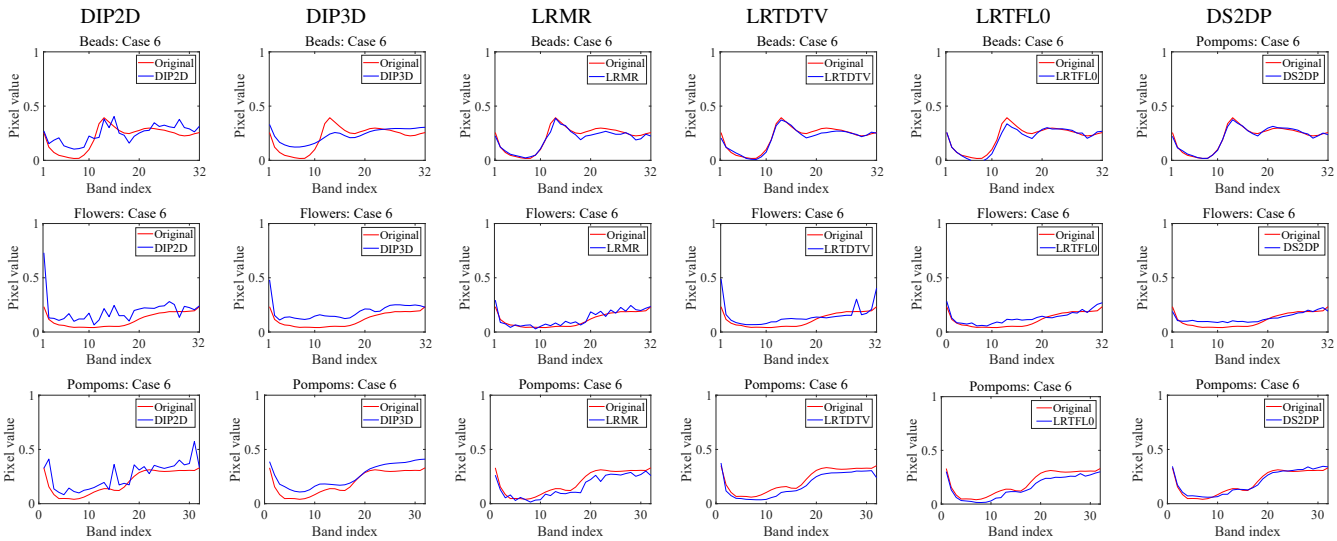


Fig. 7. Spectral curves of the denoising results by different compared methods for Case 6. From left to right: the results by DIP2D, DIP3D, LRMR, LRTDTV, LRTFL0, and DS2DP, respectively. From top to bottom: the results at spatial location (55, 60) of MSI Beads, the results at spatial location (90, 45) of the MSI Flowers, and the results at spatial location (200, 200) of the MSI Pompoms, respectively.

as severely corrupted data.

TABLE II
THE SNR OF THE DEGRADED IMAGES FOR CASE 1.

Data	WDC Mall	Pavia Centre	Pavia University	CAVE
SNR	7.196	7.691	6.297	6.318

Case 2 *(Gaussian Noise + Impulse Noise)*: In this case, the Gaussian noise for Case 1 is kept. We also additionally consider impulse noise that often happens in real HSI analysis. The impulsive noise is also added to each band. Such noise is generated following the i.i.d. zero-mean Laplacian distribution with the density parameter being 0.1.

Case 3 *(Gaussian Noise + Impulse Noise + Deadlines)*: To make the case more challenging, we include deadlines on top of Case 2; see Fig. 5 for illustration of deadlines. The deadlines are generated by nullifying some selected pixels and bands. We assume that the deadlines randomly affect 30% of the bands. Moreover, for each selected band, the number of deadlines is randomly generated from 10 to 15, and the spatial width of the deadlines is randomly selected from 1 to 3 pixels.

Case 4 *(Gaussian Noise + Impulse Noise + Diagonal Stripes)*: In this case, we replace the deadlines for Case 3 by diagonal stripes; see Fig. 5 for illustration. The elements of the diagonal stripes are all ones, which are used to simulate the constant brightness. As before, we assume that the stripes effect 30% of the bands. Moreover, for each selected band, the number

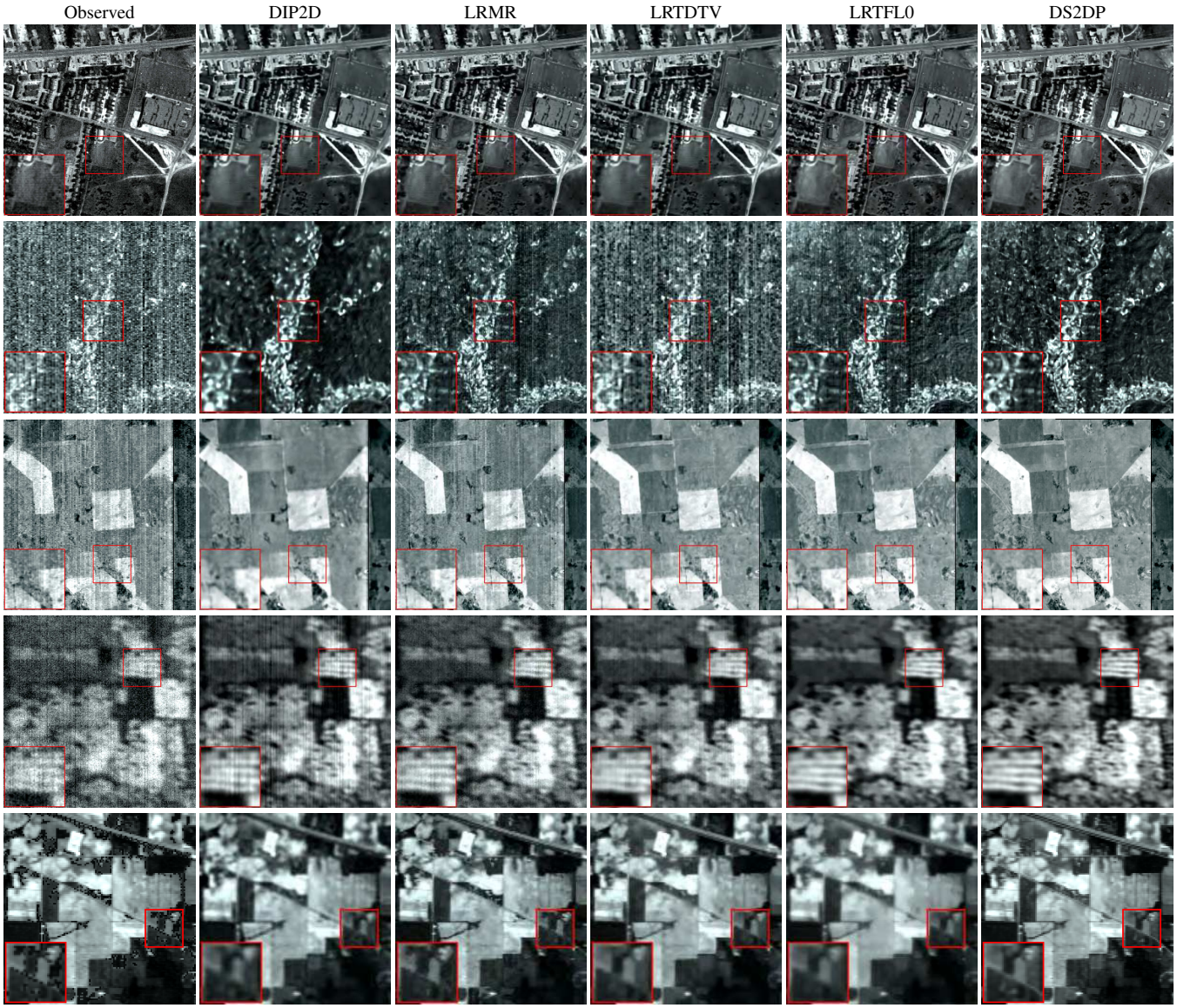


Fig. 8. Denoising results by different methods on Urban dataset, EO-1, Australian dataset, WHU HongHu dataset, and Indian Pines dataset. From top to bottom: the band 203 of Urban dataset, the band 132 of EO-1 dataset, the band 87 of Australian dataset, the band 37 of WHU HongHu dataset, and the band 24 of the Indian Pines dataset, respectively. From left to right: the observed image, the results by DIP2D, LRMR, LRTDTV, LRTFL0, and DS2DP, respectively.

of diagonal stripes is randomly generated from 15 to 30.

Case 5 *(Gaussian Noise + Impulse Noise + Vertical Stripes)*: In this case, we use the setting as for Case 4, except that vertical (other than diagonal) stripes are added; see Fig. 5. For each affected band, the number of vertical stripes is randomly generated from 10 to 15. In this case, the elements of each vertical stripe are set to a certain value randomly generated from the range of [0.6, 0.8], to diversify our simulated scenarios.

Case 6 *(Gaussian Noise + Impulse Noise + Deadlines + Diagonal Stripes + Vertical Stripes)*: To create an extra challenging case, Gaussian noise, impulse noise, and deadlines are added as for Case 3. Moreover, diagonal stripes and vertical stripes are added as for Case 4 and Case 5, respectively.

Parameter Setting. In DS2DP, there are two parameters to be manually tuned, namely, λ and R. For the parameter λ, we generally set it as $i \times 10^j$ $(i = 2, 5, 8;\ j = -6, -5, -4, -3, -2)$ for Cases 1-6. Regarding the parameter R, which is the number of endmembers in the HSI and can be determined by many existing algorithms, e.g., [32], [44], [59].

Quantitative Comparison. Table I lists the quantitative comparisons of the competing methods for Cases 1-6. The symbol "*" in Table I means that the corresponding methods have exhausted the computational resources (memory or time) but still could not produce sensible results. For the CAVE dataset, we report the averaged evaluation results from 32 images. From Table I, it is easy to see that DS2DP outperforms the state-of-the-art approaches in most cases, in terms of PSNR, SSIM, and SAM. For example, for Case 1, DS2DP achieves around 1.4 dB gain in PSNR compared to the second-best method (LRTFL0) on Pavia Centre. For Case 5, when the clean image is corrupted by Gaussian noise, impulse noise, and vertical stripes, the proposed method also achieves around 1.2 dB gain in PSNR against the same second-best method

(LRTFL0).

To test our method's performance on every band, each band's PSNR and SSIM values on WDC Mall for Cases 1-6 are shown in Fig. 4. As observed, DS2DP achieves the highest SSIM and PSNR values on most bands in all cases.

Visual Comparison. Figs. 5 and 6 show denoising results on HSIs and MSIs by different methods, respectively. The low-rank matrix model-based approach LRMR cannot effectively remove the stripes and deadlines. Additionally, LRTDTV achieves noise removal in partial bands but fails to remove the stripes and deadlines in all bands. Besides, LRTFL0 removes almost all of the noise but fails to capture the detailed information. Although there is some residual structured noise remaining in the result produced by DS2DP, the overall visual perception largely outperforms the baselines. We conjecture that such performance boost is mainly due to the deep spatial prior's ability to preserve the local spatial details—empowered by the expressiveness of appropriately crafted neural network structures.

Fig. 7 visualizes the denoising results by the algorithms in the spectral domain. One can see that, among all algorithms, the DS2DP-produced spectral signatures (on randomly selected pixel) also exhibit the highest visual similarity with those from the ground-truth image. This is consistent with its good performance in the spatial domain.

Compared with hand-crafted prior and deep image prior, the promising results of the proposed DS2DP can be attributed to that unsupervised disentangled spatio-spectral deep priors can characterize the complex scenes finely, which is beneficial to stripe removal.

C. Real Data Experiments

For real-data experiments, we choose five real-world HSI datasets to test the real noise removal, i.e., Urban dataset[4], Earth Observing-1 (EO-1) Hyperion dataset[5], Australian dataset[6], WHU HongHu dataset[7], and Indian Pines dataset[8] More precisely, the size of Urban dataset is 256×256×210, the size of EO-1 dataset is 192×192×166, the size of Australian dataset is 256×256×128, the size of WHU HongHu dataset is 256×256×64, and the size of Indian Pines dataset is 128×128×220. Regarding the proposed DS2SP, the parameters R is set as as 3, 2, 6, 2, and 5 for Urban, EO-1, Australian, WHU HongHu, and Indian Pines respectively. λ is set as 0.01, 0.01, 0.001, 0.1, and 0.000001 for Urban, EO-1, Australian, WHU HongHu, and Indian Pines, respectively.

The denoising results on these real-world datasets are shown in Fig. 8. One can see that all algorithms offer reasonable results on Urban dataset, perhaps because the data is not severely corrupted. Nevertheless, the proposed method produces the visually sharpest results. In particular, in the zoomed-in area, one can see that the proposed method's result does not have horizontal stripes, while such stripes still appear in results given by most of the baselines. For EO-1 dataset, since the selected band was severely damaged by sparse noise, the denoising task is particularly challenging. One can see that traditional methods can hardly produce satisfactory results. Nonetheless, DS2DP removes almost all of the noise—with the price of blurring the image to a certain extent—and offers the most visually pleasing result. For Australian, WHU HongHu, and Indian Pines datasets, we can see that there exists visible sparse noise in the observed images. The proposed DS2DP preserves more details while removing such noise, compared with other competing methods.

[4]https://sites.google.com/site/feiyunzhuhomepage/datasets-ground-truths

[5]http://www.lmars.whu.edu.cn/profweb/zhanghongyan/resource/noiseEOI.zip

[6]http://remote-sensing.nci.org.au/u39/public/html/index.shtml

[7]http://rsidea.whu.edu.cn/resource_WHUHi_sharing.htm

[8]http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes

V. Further Discussions

A. Analysis of Algorithm Complexity

In this part, we analyze the algorithm complexity of the proposed method on HSI WDC Mall and MSI Superballs for Case 6. DIP2D and DIP3D are selected as the baseline models since they stand for the unsupervised HSI denoising models. For DIP2D and DIP3D, we select the network structure with the best performance according to the original implementation.

For a fair comparison, the network structure utilized in DS2DP, which is expected to capture the spatial prior information, is simply designed as U-Net-like "hourglass" architecture. Moreover, we do not focus on meticulous designs on reducing the model scale in this work, i.e., depth-wise separable convolution, model pruning, and model compression [60]. These techniques may be used to reduce the network complexity of all methods (including ours), but this is beyond the scope of this work. Table III lists the scale of parameters of different methods on HSI WDC Mall and MSI Superballs. Moreover, the corresponding values of PSNR, SSIM, and the execution time (in minutes) are also reported in Table III.

TABLE III
THE RELEVANT INDICATORS OF DIP3D, DIP2D, AND DS2DP ON HSI WDC MALL AND MSI SUPERBALLS FOR CASE 6. THE **BEST** AND SECOND-BEST VALUES ARE HIGHLIGHTED IN BOLD AND UNDERLINED, RESPECTIVELY.

Data	Methods	Params	PSNR	SSIM	Time
HSI: WDC Mall (256 × 256 × 191)	DIP3D	6.275M	*	*	*
	DIP2D	2.342M	21.759	0.594	15.625
	DS2DP	2.150M	**34.352**	**0.967**	19.544
	DS2DP*	**0.574M**	32.710	0.942	**12.353**
MSI: Superballs (256 × 256 × 32)	DIP3D	6.275M	20.705	0.399	16.091
	DIP2D	2.138M	20.901	0.408	3.314
	DS2DP	2.150M	**35.037**	**0.881**	4.677
	DS2DP*	**0.574M**	34.779	0.871	**2.181**

As shown in Table III, the proposed DS2DP achieves significantly better performance with roughly equal parameters and slightly longer execution time compared with the baseline models. More precisely, DS2DP outperforms DIP2D by 12.593 dB and 14.136 dB in terms of PSNR on HSI WDC Mall and MSI Superballs, respectively. DS2DP achieves performance gains over DIP3D with about 14 dB on MSI Superballs.

In our original implementation, to push DS2DP to attain the (empirically) achievable "best" performance, we use several

parallel networks with the same architecture to generate the abundance maps. To reduce the number of the parameters, we let the parameters be shared between several parallel networks. This method is denoted by DS2DP* and its performance is also shown in Table III. This way, the parameter amount reduces by 3/4 and the execution time reduces by 2/5, while the PSNR is essentially unaffected.

B. Effectiveness of the Deep Spectral and Spatial Priors

In this part, we take a deeper look at the deep spectral and spatial priors in DS2DP. To verify these two priors' effectiveness, we conduct ablation studies for Case 6 using the WDC Mall data. The impacts of our designed priors in spectral and spatial domains are shown in Fig. 9 and Fig. 10, respectively.

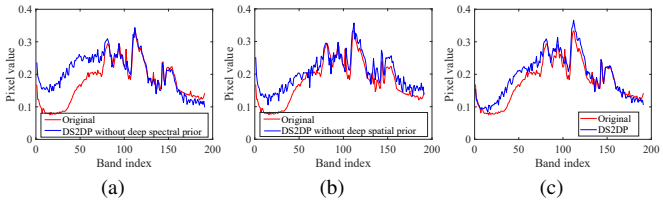


Fig. 9. Effectiveness of the deep priors in the spectral domain. The red curve is the ground truth of a selected pixel at spatial location (120, 120) for illustration. The blue curves correspond to: (a) the estimated spectrum by DS2DP without deep spectral prior; (b) the estimated spectrum by DS2DP without deep spatial prior; and (c) the estimated spectrum by the proposed DS2DP.

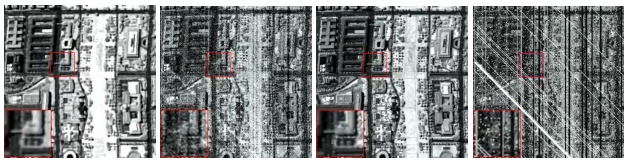


Fig. 10. Effectiveness of the deep priors in the spatial domain. The four figures correspond to: (a) the denoising result by DS2DP without deep spectral prior; (b) the denoising result by DS2DP without deep spatial prior; (c) the denoising result by DS2DP; and (d) the observed image.

Fig. 9 (a) shows that when only employing deep spatial prior in DS2DP without the deep spectral prior, the estimated spectrum of the selected pixel is not accurate. In contrast, when considering both types of priors in DS2DP, the results become much more promising; see (c). Besides, DS2DP without the deep spatial prior and the complete DS2DP both achieve satisfactory performance on most bands. This supports our idea for disentangling the spatial and spectral information and modeling them individually.

Fig. 10 shows similar effects in the spatial domain. One can see that there is obviously visible noise in the results when only employing the deep spectral prior. However, when considering the two priors, the performance is clearly much more visually pleasing. In addition, Fig. 10 (c) also clearly demonstrates the disentanglement between the spatial and spectral effects.

Moreover, the quantitative comparisons of the denoising results by DS2DP without deep spectral prior, DS2DP without deep spatial prior, and DS2DP are shown in Table IV.

TABLE IV
QUANTITATIVE COMPARISON OF DS2DP WITHOUT DEEP SPECTRAL PRIOR, DS2DP WITHOUT DEEP SPATIAL PRIOR, AND DS2DP. THE **BEST** AND SECOND-BEST VALUES ARE HIGHLIGHTED IN BOLD AND UNDERLINED, RESPECTIVELY.

Method	PSNR	SSIM	SAM
DS2DP without deep spectral prior	27.316	0.837	0.167
DS2DP without deep spatial prior	31.927	0.903	0.181
DS2DP	**34.352**	**0.967**	**0.116**

C. Effectiveness of the Sparsity Regularization

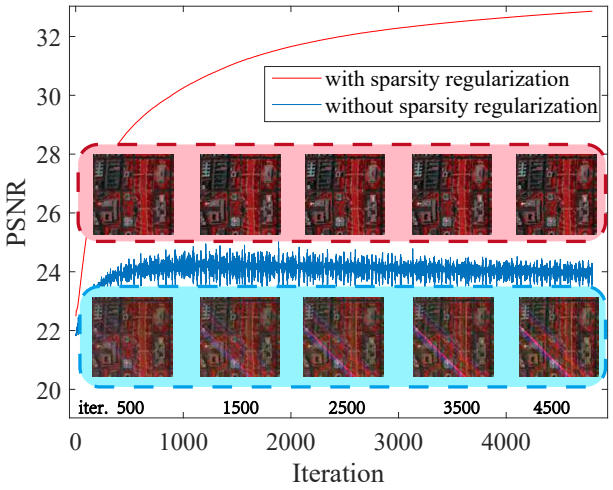


Fig. 11. The history of PSNR values and the corresponding denoising results by DS2DP with and without sparsity regularization.

To verify the sparsity regularization's effect, we design a comparative experiment, also using Case 6 and the WDC Mall data. The result is shown in Fig. 11. One can see that when the sparsity regularization is not applied, the PSNR first increases and then declines slowly as the number of iterations increases. In contrast, when sparsity regularization is employed, the PSNR maintains an upward trend during the iterations—and eventually exhibits a big PSNR improvement relative to the former case.

Fig. 11 also shows the visualization of the algorithm with and without the sparsity regularization in the 1,000th iteration. One can see that the proposed method produces a relatively clean image, which clearly shows an advantage over the case without the L_1 term.

Moreover, to further support the effectiveness of the sparsity regularization, we list the quantitative indexes (i.e., PSNR, SSIM, and SAM) of the denoising results by DS2DP without sparsity regularization and DS2DP in Table V.

D. Sensitivity Analysis and Selection of the Parameters R and λ

1) Parameter R: In this part, we study the parameter sensitivity of the number of materials R on real HSI Indian Pines—which contains 16 materials. Then, we further discuss the selection of the parameter R in real scenarios.

TABLE V
THE QUANTITATIVE INDEXES (I.E., PSNR, SSIM, AND SAM) OF THE DENOISING RESULTS BY DS2DP WITHOUT SPARSITY REGULARIZATION AND DS2DP. THE **BEST** IS HIGHLIGHTED IN BOLD.

Method	PSNR	SSIM	SAM
DS2DP without sparsity regularization	23.762	0.668	0.253
DS2DP	**34.352**	**0.967**	**0.116**

Due to the absence of the ground truth, we display the denoising images with the corresponding number of endmembers R and network parameters in Fig. 12. We can observe that when R is smaller than 3, the visual effect is not satisfactory. When R is larger than 3, the denoising results are visually the same. This observation demonstrates that the proposed DS2DP is robust with respect to the parameter R. Such robustness against underestimated R is a bit surprising at first glance, yet understandable—under the linear mixture model, the range space spanned by the first several principal components may contain most of the energy. Hence, underestimating R may not be very detrimental in many cases. Note that when selecting R, the number of network parameters should also be considered and balanced, since the number of the network parameters increases *substantially* with the increasing value of R. Thus, we select the value of R from a starting number 2 with the increments 1 in practice, which balances between the visual effect and the number of the parameters.

Other than using visual validation, the parameter R could also be selected by the existing effective number of endmember estimation algorithms, e.g., [44], [59].

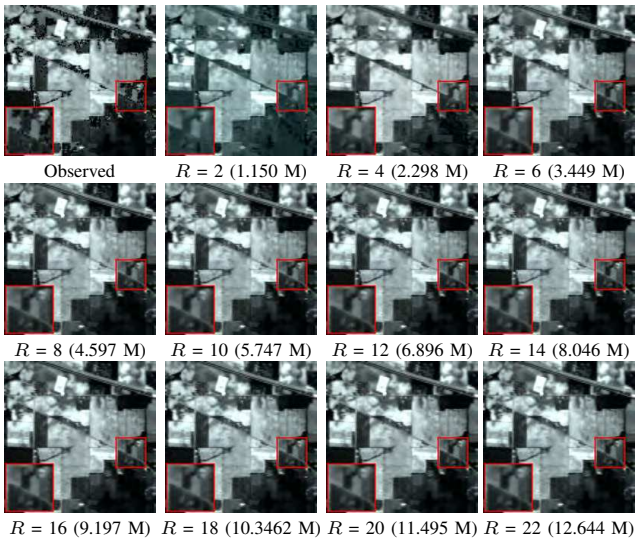


Fig. 12. The band 24 of the denoising results by DS2DP on real-world HSI Indian Pines with the corresponding number of endmembers R and network parameters.

2) Parameter λ: In this subsection, we conduct an empirical sensitivity analysis of the parameter λ, using the real-world HSI Australian, Urban, and WHU HongHu with different corruption levels. Then, we further discuss the selection of the parameter λ in real scenarios.

Fig. 13 shows the denoising results by DS2DP with different λ on these three real HSIs. We can observe that the proposed DS2DP attains the best visual effect when λ = 0.001, 0.01, and 0.1 for real HSIs Australian, Urban, and WHU HongHu, respectively. This observation reflects that the regularization parameter λ is sensitive to the weight of the sparse noise. Thus, we apply (rough) grid search from a range of parameters to select a relatively "good" one in terms of visual effect. For example, we select λ from the candidate set {0.001, 0.01, 0.1} and visually determine which parameter is more plausible.

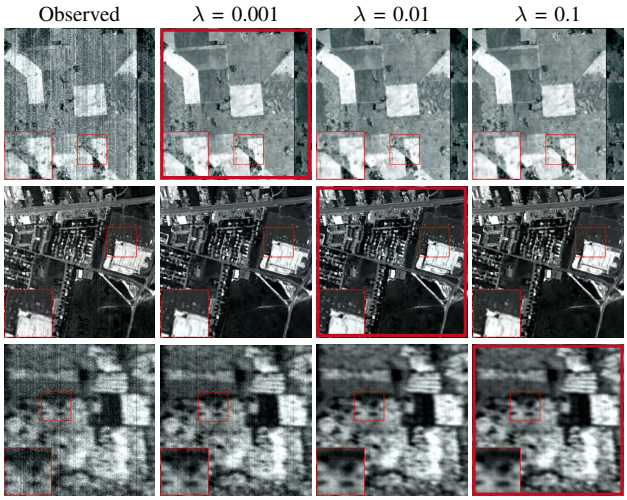


Fig. 13. Denoising results by DS2DP with different λ on real-world HSI Australian, Urban, and WHU HongHu datasets. From top to bottom: the band 87 of Australian dataset, the band 1 of Urban dataset, and the band 37 of WHU HongHu dataset, respectively. From left to right: the observed image, the results with λ = 0.001, 0.01, and 0.1, respectively.

Additionally, note that for images that are more severely contaminated by sparse noise, λ should be larger. Hence, in addition to visual validation, λ could also be selected from a collection of candidates (e.g., {0.001, 0.01, 0.1}) by estimating the corruption level. The level of corruption/noise can be roughly estimated by some hyperspectral noise estimation algorithms, e.g., those in [5], [61]. These noise/outlier estimation methods often use less powerful models relative to our deep prior-based one in terms of expressiveness, but may run fairly fast to yield some initial estimations for the noise level, which can serve our parameter selection purpose.

E. Impact of the Random Input to DS2DP

As illustrated previously, the input of the proposed DS2DP is random but known noise sampled from a uniform distribution. One may wonder if the input $\boldsymbol{z}_r$ has a significant impact on results? The answer is negative. We show this by calculating the means and standard deviations of the algorithm outputs' PSNR for Cases 1-6 on WDC Mall. For each case, we run ten trials with different $\boldsymbol{z}_r$'s that are randomly generated from U(-0.05, 0.05), where U stands for uniform distribution. The results are shown in Table VI. One can see that, perhaps a bit surprisingly, the standard deviations of the results are fairly small—which means the method is essentially not affected by the random input to a good extent.

F. Impact of the Distribution Parameter of the Random Input to DS2DP

We denote the uniform distribution as U(-a, a), where -a and a are the lower boundary and upper boundary,

TABLE VI
THE DENOISING RESULTS' PSNR VALUES (MEAN±STD.DEV) FOR CASES 1-6 ON WDC MALL

Case	Case 1	Case 2	Case 3
PSNR	36.213±0.254	35.636±0.358	34.297±0.276
Case	Case 4	Case 5	Case 6
PSNR	35.511±0.344	34.802±0.297	34.173±0.221

respectively. In our experiment, the parameter a is set as 0.05 following [29]. In this part, we have conducted an empirical sensitivity analysis of parameter a. Fig. 14 presents the PSNR values by the proposed method with different a for Case 6. We can see that, the PSNR value does not fluctuate greatly with different a. Therefore, the denoising result is not sensitive to the value of a.

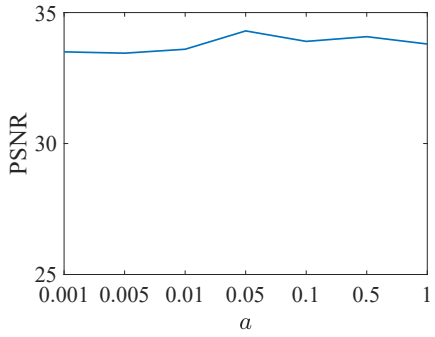


Fig. 14. The influence of a on HSI WDC Mall for Case 6

G. HSI Classification Validation

TABLE VII
QUANTITATIVE COMPARISONS OF CLASSIFICATION ACCURACY OF THE DENOISED HSI BY THE COMPETING METHODS. THE **BEST** AND SECOND-BEST VALUES ARE HIGHLIGHTED IN BOLD AND UNDERLINED, RESPECTIVELY.

Metrics	Original	LRMR	LRTDTV	LRTFL0	DS2DP
Overall accuracy	76.5%	83.6%	83.8%	81.7%	**84.9%**
Kappa coefficient	0.733	0.814	0.816	0.796	**0.828**

To further evaluate the effectiveness of the proposed model, we considered a hyperspectral classification task on the Indian Pine data[9]. We use the observed data and the outputs of different denoising algorithms as the inputs to the classification task—which is performed by a support vector machine (SVM) classifier. For each class, the number of training samples is 40, and the number of test samples ranges from 6 to 2415 for various classes.

Table VII reports the classification performance. The performance is measured by two commonly used metrics for classification tasks, namely, the overall accuracy [62] and the Kappa coefficient [62]. One can see that the denoising methods can improve the classification performance by simply using the observed data. In addition, one can observe that DS2DP achieves around 1.1% and 0.012 gain in terms of the overall accuracy and the Kappa coefficient, respectively, compared to the second-best method (LRTDTV).

[9]http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes

VI. CONCLUSIONS

We proposed an unsupervised deep prior-based HSI denoising framework. Unlike existing methods that directly learn deep generative networks for the entire HSI, our method leverages the classic LMM to disentangle the spatial and spectral information, and learns two types of deep priors for the abundance maps and the spectral signatures of the end-members, respectively. Our design is driven by the challenges that network structures used in deep priors for different types of images (in particular, HSIs) may be hard to search. Using our information-disentangled framework, empirically validated unsupervised deep image prior structures for natural images can be easily incorporated for HSI denoising. Besides, the network complexity can be substantially reduced with proper parameter sharing, making the learning process more affordable than existing approaches. We also proposed a structured noise-robust optimization criterion that is tailored for HSI denoising. We tested our method using extensive experiments with various cases and ablation studies. The numerical results demonstrated promising HSI denoising performance of the proposed approach. A note is that, despite its promising performance, unsupervised DIP research has still been largely empirical—Rigorous analysis has been elusive. Interesting future directions include enhanced theoretical understanding to unsupervised DIP (e.g., in terms of sample complexity and generalization error analysis), which may be of broader interest beyond HSI denoising.

REFERENCES

[1] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 354–379, 2012.
[2] W. He, H. Zhang, L. Zhang, and H. Shen, "Total-variation-regularized low-rank matrix factorization for hyperspectral image restoration," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 178–188, 2016.
[3] Y. Wang, J. Peng, Q. Zhao, Y. Leung, X. Zhao, and D. Meng, "Hyperspectral image restoration via total variation regularized low-rank tensor decomposition," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 11, no. 4, pp. 1227–1243, 2018.
[4] F. Xiong, J. Zhou, and Y. Qian, "Hyperspectral restoration via l_0 gradient regularized low-rank tensor factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 12, pp. 10410–10425, 2019.
[5] L. Zhuang and M. K. Ng, "Hyperspectral mixed noise removal by ℓ_1-norm-based subspace representation," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 1143–1157, 2020.
[6] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, 2007.
[7] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 2272–2279, 2009.
[8] W. Dong, L. Zhang, G. Shi, and X. Li, "Nonlocally centralized sparse representation for image restoration," *IEEE Trans. Image Process.*, vol. 22, no. 4, pp. 1620–1630, 2012.
[9] Y. Chen, J. Li, and Y. Zhou, "Hyperspectral image denoising by total variation-regularized bilinear factorization," *Signal Process.*, vol. 174, p. 107645, 2020.

[10] G. Chen and S. Qian, "Denoising of hyperspectral imagery using principal component analysis and wavelet shrinkage," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 3, pp. 973–980, 2011.

[11] P. Zhong and R. Wang, "Multiple-spectral-band CRFs for denoising junk bands of hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 4, pp. 2260–2275, 2013.

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

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

[14] Y. Chang, L. Yan, X. L. Zhao, H. Fang, Z. Zhang, and S. Zhong, "Weighted low-rank tensor recovery for hyperspectral image restoration," *IEEE Trans. Cybern.*, vol. 50, no. 11, pp. 4558–4572, 2020.

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

[16] F. Xu, Y. Chen, C. Peng, Y. Wang, X. Liu, and G. He, "Denoising of hyperspectral image using low-rank matrix factorization," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 7, pp. 1141–1145, 2017.

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

[18] L. Zhuang and J. M. Bioucas-Dias, "Hyperspectral image denoising based on global and non-local low-rank factorizations," in *Proc. Int. Conf. Image Process.*, pp. 1900–1904, 2017.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2016.

[20] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," in *Proc. Int. Conf. Learn. Representations*, 2014.

[21] W. Wang, Y. Huang, Y. Wang, and L. Wang, "Generalized autoencoder: A neural network framework for dimensionality reduction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 496–503, 2014.

[22] Z. Wang, Q. She, and T. E. Ward, "Generative adversarial networks in computer vision: A survey and taxonomy," *arXiv preprint arXiv:1906.01529*, 2019.

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

[24] W. Dong, H. Wang, F. Wu, G. Shi, and X. Li, "Deep spatial–spectral representation learning for hyperspectral image denoising," *IEEE Trans. Comput. Imag.*, vol. 5, no. 4, pp. 635–648, 2019.

[25] Q. Yuan, Y. Wei, X. Meng, H. Shen, and L. Zhang, "A multiscale and multidepth convolutional neural network for remote sensing imagery pan-sharpening," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 3, pp. 978–989, 2018.

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

[27] Y. Chang, M. Chen, L. Yan, X. Zhao, Y. Li, and S. Zhong, "Toward universal stripe removal via wavelet-based deep convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2880–2897, 2020.

[28] W. Dong, H. Wang, F. Wu, G. Shi, and X. Li, "Deep spatial–spectral representation learning for hyperspectral image denoising," *IEEE Trans. Comput. Imag.*, vol. 5, no. 4, pp. 635–648, 2019.

[29] V. Lempitsky, A. Vedaldi, and D. Ulyanov, "Deep image prior," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 9446–9454, 2018.

[30] O. Sidorov and J. Y. Hardeberg, "Deep hyperspectral prior: Single-image denoising, inpainting, super-resolution," in *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 3844–3851, 2019.

[31] W.-K. Ma, J. M. Bioucas-Dias, T.-H. Chan, N. Gillis, P. Gader, A. J. Plaza, A. Ambikapathi, and C.-Y. Chi, "A signal processing perspective on hyperspectral unmixing: Insights from remote sensing," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 67–81, 2014.

[32] X. Fu, W.-K. Ma, J. M. Bioucas-Dias, and T.-H. Chan, "Semiblind hyperspectral unmixing in the presence of spectral library mismatches," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 9, pp. 5171–5184, 2016.

[33] C. I. Kanatsoulis, X. Fu, N. D. Sidiropoulos, and W.-K. Ma, "Hyperspectral super-resolution: A coupled tensor factorization approach," *IEEE Trans. Signal Process.*, vol. 66, no. 24, pp. 6503–6517, 2018.

[34] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, "Tensor decomposition for signal processing and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3551–3582, 2017.

[35] J. Liu, Y. Sun, X. Xu, and U. S. Kamilov, "Image restoration using total variation regularized deep image prior," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, pp. 7715–7719, 2019.

[36] M. Ye, Y. Qian, and J. Zhou, "Multitask sparse nonnegative matrix factorization for joint spectral–spatial hyperspectral imagery denoising," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2621–2639, 2014.

[37] M. A. Veganzones, J. E. Cohen, R. C. Farias, J. Chanussot, and P. Comon, "Nonnegative tensor CP decomposition of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 5, pp. 2577–2588, 2015.

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

[39] Y. Wang, J. Peng, Q. Zhao, Y. Leung, X. Zhao, and D. Meng, "Hyperspectral image restoration via total variation regularized low-rank tensor decomposition," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 11, no. 4, pp. 1227–1243, 2018.

[40] F. Xiong, J. Zhou, and Y. Qian, "Hyperspectral restoration via ℓ_0 gradient regularized low-rank tensor factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 12, pp. 10410–10425, 2019.

[41] J. L. Wang, T. Z. Huang, X. L. Zhao, T. X. Jiang, and M. K. Ng, "Multi-dimensional visual data completion via low-rank tensor representation under coupled transform," *IEEE Trans. Image Process.*, vol. 30, pp. 3581–3596, 2021.

[42] Y. Y. Liu, X. L. Zhao, Y. B. Zheng, T. H. Ma, and H. Zhang, "Hyperspectral image restoration by tensor fibered rank constrained optimization and plug-and-play regularization," *IEEE Trans. Geosci. Remote Sens.*, pp. 1–17, 2021.

[43] E. Wycoff, T.-H. Chan, K. Jia, W.-K. Ma, and Y. Ma, "A non-negative sparse promoting algorithm for high resolution hyperspectral imaging," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, pp. 1409–1413, 2013.

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

[45] N. Yokoya, T. Yairi, and A. Iwasaki, "Coupled nonnegative matrix factorization unmixing for hyperspectral and multispectral data fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 2, pp. 528–537, 2012.

[46] H. K. Aggarwal and A. Majumdar, "Hyperspectral unmixing in the presence of mixed noise using joint-sparsity and total variation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 9, pp. 4257–4266, 2016.

[47] Y. Qian, F. Xiong, S. Zeng, J. Zhou, and Y. Y. Tang, "Matrix-vector nonnegative tensor factorization for blind unmixing of hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 3, pp. 1776–1792, 2017.

[48] F. Xiong, Y. Qian, J. Zhou, and Y.-Y. Tang, "Hyperspectral unmixing via total variation regularized nonnegative tensor factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 2341–2357, 2019.

[49] C. Lanaras, E. Baltsavias, and K. Schindler, "Hyperspectral super-resolution by coupled spectral unmixing," in *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 3586–3594, 2015.

[50] L. Loncan, J. Chanussot, S. Fabre, and X. Briottet, "Hyperspectral pan-sharpening based on unmixing techniques," in *Workshop Hyperspectral Image Signal Proces.: Evol. Remote Sens.*, pp. 1–4, 2015.

[51] A. Karami, R. Heylen, and P. Scheunders, "Hyperspectral image compression optimized for spectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 5884–5894, 2016.

[52] Y. Zhao, J. Yang, C. Yi, and Y. Liu, "Joint denoising and unmixing for hyperspectral image," in *Workshop Hyperspectral Image Signal Proces.: Evol. Remote Sens.*, pp. 1–4, 2014.

[53] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, Springer International Publishing, 2015.

[54] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015.

[55] R. Rojas, *The Backpropagation Algorithm*, pp. 149–182. Berlin, Heidelberg: Springer Berlin Heidelberg, 1996.

[56] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 613–627, 1995.

[57] Q. Shi, H. Sun, S. Lu, M. Hong, and M. Razaviyayn, "Inexact block coordinate descent methods for symmetric nonnegative matrix factorization," *IEEE Trans. Signal Process.*, vol. 65, no. 22, pp. 5995–6008, 2017.

[58] H. Othman and Shen-En Qian, "Noise reduction of hyperspectral imagery using hybrid spatial-spectral derivative-domain wavelet shrinkage," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 2, pp. 397–408, 2006.
[59] X. Fu, W.-K. Ma, T.-H. Chan, and J. M. Bioucas-Dias, "Self-dictionary sparse regression for hyperspectral unmixing: Greedy pursuit and pure pixel search are related," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 6, pp. 1128–1141, 2015.
[60] S. Ge, Z. Luo, S. Zhao, X. Jin, and X. Zhang, "Compressing deep neural networks for efficient visual inference," in *Proc. IEEE Int. Conf. Multimedia Expo*, pp. 667–672, 2017.
[61] B. Rasti, P. Ghamisi, and J. A. Benediktsson, "Hyperspectral mixed gaussian and sparse noise reduction," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 3, pp. 474–478, 2020.
[62] S. Li, Q. Hao, G. Gao, and X. Kang, "The effect of ground truth on performance evaluation of hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 12, pp. 7195–7206, 2018.

Yu-Chun Miao is currently a undergraduate majoring in Mathematics and Applied Mathematics from the University of Electronic Science and Technology of China (UESTC), Chengdu, China. His research interests include tensor learning and high-dimensional image processing.

Xi-Le Zhao received the M.S. and Ph.D. degrees from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2009 and 2012. He is currently a Professor with the School of Mathematical Sciences, UESTC. His research interest mainly focuses on model-driven and data-driven methods for image processing problems. His homepage is https://zhaoxile.github.io/.

Xiao Fu (Senior Member, IEEE) received the B.Eng. and M.Sc. degrees from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2005 and 2010, respectively. He received the Ph.D. degree in Electronic Engineering from The Chinese University of Hong Kong (CUHK), Shatin, N.T., Hong Kong, in 2014. He was a Postdoctoral Associate with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, USA, from 2014 to 2017. Since 2017, he has been an Assistant Professor with the School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR, USA. His research interests include the broad area of signal processing and machine learning.

Dr. Fu received a Best Student Paper Award at ICASSP 2014, and was a recipient of the Outstanding Postdoctoral Scholar Award at University of Minnesota in 2016. His coauthored papers received Best Student Paper Awards from IEEE CAMSAP 2015 and IEEE MLSP 2019, respectively. He serves as a member of the Sensor Array and Multichannel Technical Committee (SAM-TC) of the IEEE Signal Processing Society (SPS). He is also a member of the Signal Processing for Multisensor Systems Technical Area Committee (SPMuS-TAC) of EURASIP. He is the Treasurer of the IEEE SPS Oregon Chapter. He serves as an Editor of SIGNAL PROCESSING. He was a tutorial speaker at ICASSP 2017 and SIAM Conference on Applied Linear Algebra 2021.

Jian-Li Wang received the B.S. degree in mathematics and applied mathematics from Neijiang Normal University, Neijiang, China, in 2017. She is currently working toward the Ph.D. degree in the School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu, China. She is also a visiting student with the School of Electrical and Electronic Engineering (EEE), Nanyang Technological University (NTU), Singapore. Her current research interests include high-dimensional data processing, tensor modeling and computing, computer vision, and deep learning. More information can be found in her homepage https://wangjianli123.github.io/homepage/.

Yu-Bang Zheng received the B.S. degree in information and computing science from Anhui University of Finance and Economics, Bengbu, China, in 2017. He is currently working toward the Ph.D. degree in the School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu, China. He is also a visiting student with the Tensor Learning Team, RIKEN Center for Advanced Intelligence Project, Tokyo, Japan. His current research interests include tensor modeling and computing, tensor learning, and high-dimensional data processing. More information can be found in his homepage https://yubangzheng.github.io/.