Unsupervised Hyperspectral Denoising Based on Deep Image Prior and Least Favorable Distribution

Keivan Faghih Niresi and Chong-Yung Chi, *Life Fellow, IEEE*

***Abstract*—This article considers the inverse problem under hyperspectral images (HSIs) denoising framework. Recently, it has been shown that deep learning is a promising approach to image denoising. However, deep learning to be effective usually needs a massive amount of training data. Moreover, in a practical scenario, HSIs may get contaminated by different kinds of noises such as Gaussian and/or sparse noise. Lately, it has been reported that the convolutional neural network (CNN), the core element used by deep image prior (DIP), is able to capture image statistical characteristics without the need of training, i.e., restore the clean image blindly. Nonetheless, there exists some performance gap between DIP and state-of-the-art methods in HSIs (e.g., low-rank models). By applying the Huber loss function (HLF), which is derived through a least favorable distribution in robust statistics, to DIP, we propose a novel unsupervised denoising algorithm, referred as to the HLF-DIP, free from pretraining and without involving any regularizer. Extensive experimental results are provided to demonstrate that the proposed HLF-DIP algorithm significantly outperforms seven state-of-the-art algorithms in both complexity (thanks to no regularization) and robustness against complex noise (e.g., mixed types of noises).**

***Index Terms*—Adam optimizer, convolutional neural network, deep image prior, Huber loss function, hyperspectral denoising, least favorable distribution, robust statistics.**

I. Introduction

HIGH-QUALITY hyperspectral images (HSIs), providing spectral and spatial information of a hyperspectral scene, have been widely used in remote sensing, computer vision, disease diagnosis, etc. [1]–[3] However, HSI, a 3-D data cube composed of the 2-D spatial domain and the spectral domain, is susceptible to various kinds of noises such as Gaussian and sparse noises caused by atmospheric interference and sensor internal issues. Mixed noises severely degrade the performance of subsequent applications, such as classification, superresolution, and unmixing [4]–[8]. For this reason, denoising HSIs is an essential preprocessing for any further applications.

Manuscript received 28 March 2022; revised 5 May 2022 and 8 June 2022; accepted 26 June 2022. Date of publication 1 July 2022; date of current version 1 August 2022. This work was supported by the Ministry of Science and Technology, ROC, under Grants MOST 111-2221-E-007-047-MY2, MOST 110-2221-E-007-048, and MOST 109-2221-E-007-083. *(Corresponding author: Keivan Faghih Niresi.)*

Keivan Faghih Niresi is with the Institute of Communications Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan (e-mail: keivan@m109.nthu.edu.tw; keyvan.faghih@gmail.com).

Chong-Yung Chi is with the Institute of Communications Engineering and Department of Electrical Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan (e-mail: cychi@ee.nthu.edu.tw).

Digital Object Identifier 10.1109/JSTARS.2022.3187722

Some high-performance HSI denoising approaches have been proposed in recent years. Bandwise processing methods are one of the most straightforward tools for applying 2-D denoiser to each band of observed HSIs individually [9], [10]. Nonetheless, HSI data are known to be highly correlated in both the spectral and spatial domains, and ignoring the spectral data correlation may downgrade the denoisers' performance. For this reason, a wavelet shrinkage approach was suggested in the derivative-domain [11]. Moreover, for volumetric data with similar qualities to HSI, block matching and 4-D filtering (BM4D) was proposed [12], in which only Gaussian and Rician noises are considered, and hence the performance of the algorithm degrades very drastically in the scenario of complex noise (e.g., outliers, stripes, and sparse noise). Other HSI denoising algorithms are mostly based on the optimization method while the data-fitting and regularization terms make up the majority of the optimization problems used for the design of HSI restoration algorithms. The prior information about the underlying properties for the recovered HSI corresponds to the sum of the regularization terms. Total variation (TV) [10] is one of the most often used regularization approaches for image restoration in order to reduce noise levels and retain image edges for natural image denoising. However, this method just considers the spatial correlation but ignores the spectral correlation of HSI. To capture the spectral correlation of HSI, in [13], spatio-spectral total variation is proposed using 2D-TV for capturing the smoothness of the spatial dimension and 1D-TV for modeling the spectral correlation. Although spatio-spectral total variation regularization can improve the denoising performance, some artifacts still remain in the restored image. Another effective regularization of HSI is low rankness due to high spectral-spatial correlation. As a result, a significant amount of Gaussian noise can be efficiently eliminated [14]–[16]. Subspace-based approaches have been frequently used in recent years, which adequately account for spectral correlation due to the low-rank property of the HSI, thus yielding improved outcomes for HSI denoising [17], [18]. For HSI mixed noise reduction, the robust principal component analysis (RPCA) was first introduced in [19]. The core concept of RPCA is based on the consideration of the low-rankness of the HSI in the presence of sparse noise. However, the RPCA may not be able to preserve the spatial correlation in HSI, consequently resulting in inferior denoising outcomes for large-magnitude noisy situations. Most existing works focused on replacing the challenging nonconvex rank minimization problem with a convex surrogate. For instance, the nuclear norm employed as the surrogate of the nonconvex data matrix rank has been one

of the most popular options [20] used in HSI inverse problems. Despite the fact that the above-mentioned rank approximations have been demonstrated effective, they all need to perform the computation-expensive singular value decomposition (SVD) in each iteration. Low-rank tensor representation of an HSI can finely preserve the underlying structure of a 3-D hyperspectral cube. Moreover, it also captures data correlation in 3-D simultaneously instead of 2-D, thus achieving better performance than low-rank matrix approximations [21]–[23]. On the other hand, some HSI denoising methods use source separation techniques, such as the unmixing-based denoising (UBD) method proposed in [24]. Prior to denoising, UBD estimates endmember signatures using the computation-efficient vertex component analysis (VCA), under the assumption of the existence of pure pixels, which however may not be true in practical applications. In any event, both the structures of HSIs and noise have been considered in the meantime, thereby drawing a number of HSI denoising methods that are suitable for handling Gaussian noise but not very effective to handle complicated mixed noises.

Recently, deep learning approaches [25]–[30] have shown effective in solving inverse problems of HSIs. In general, supervised algorithms need a large amount of training data to perform excellently in HSI denoising. However, massive amounts of hyperspectral training data may not be available in practice and HSI noise in practical scenarios may be quite complicated. To attack this problem, Ulyanov *et al.* [31] recently introduced deep image prior (DIP), an unsupervised deep learning-based image restoration algorithm. The DIP employs a convolutional neural network (CNN) to restore images without the need of training. The CNN, with a randomly generated network input and the restored image as the network output, can effectively lower Gaussian noise level of the given noisy image through proper iteration steps. Following that, Sidorov *et al.* [32] applied DIP to HSIs restoration. The optimization approach of DIP for HSI restoration is formulated in a fashion similar to supervised methods. Although DIP offered a novel paradigm for solving inverse problems in imaging without any handcrafted priors (regularization), the quality of DIP approaches is not very competitive in performance with state-of-the-art frameworks. Recently for HSI, Nguyen *et al.* [33] proposed a DIP-based model by employing Stein's unbiased risk estimator (SURE) in the cost function. Despite the fact that SURE can improve the performance of DIP for HSI in terms of Gaussian and Poisson noise, it is still not capable to handle the bunch of outliers. In [34], total variation regularization is added to the objective function in order to upgrade the performance of DIP. Although this approach can improve the performance of DIP paradigm, there is still some performance gap that unsupervised low-rank model based methods outperform unsupervised deep learning-based approaches [8]. In [35] and [36], the authors used a plug-and-play framework to boost DIP approaches. Nevertheless, all these methods need regularization besides plugging a different kind of denoiser to boost the performance. Since HSIs may be contaminated by mixed noises, most existing denoisers cannot clean the degraded image very well.

In this work, we propose a novel framework for noise reduction of noisy HSIs motivated by the benefits of DIP approaches and robust statistics as shown in Fig. 1. Different from the previous methods, we reformulate the optimization problem by using the least favorable distribution (LFD). Huber proposed LFD family distribution using normal distribution in the small to mid-magnitude and exponential tails in large magnitude [37], [38], e.g., suitable for modeling outliers in HSIs. Extensive experiments show much superior overall performance of the proposed HLF-DIP denoising algorithm over the state-of-the-art methods.

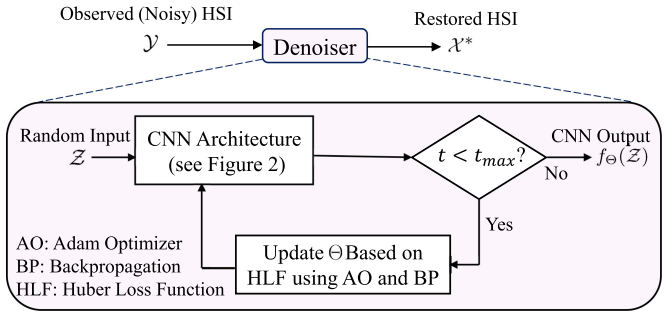


Fig. 1. Proposed HSI denoising framework.

The main contributions of this article are summarized as follows:

1) *To the best of our knowledge*, LFD is considered to formulate HSI denoising inverse problems for the first time, and by robust statistics, complex noises in HSI can be effectively suppressed.
2) With the powerful DIP, a user-friendly unsupervised HSI denoising algorithm, called HLF-DIP, is proposed with no need of both regularization and pretraining. Extensive experimental results demonstrate that its overall performance is significantly superior to state-of-the-art methods, beyond the nontrivial gap of the DIP performance below state-of-the-art methods.

The rest of the article is organized as follows. Section II reviews related background on DIP and robust statistics for ease of the ensuing presentation. In Section III, the proposed HLF-DIP denoising algorithm is presented, followed by the CNN architecture used. In Section IV, extensive experiments on both simulated and real datasets are conducted to evaluate the performance of the proposed algorithm and comparison with some state-of-the-art methods. Finally, some conclusions are drawn in Section V.

II. Related Backgrounds

For ease of the ensuing presentation, we denote third-order tensors by calligraphic letters, e.g., $\mathcal{A} \in \mathbb{R}^{H\times W\times B}$ (real tensor with dimension $H \times W \times B$) and denote its (i, j, k)th entry as a_{ijk}. The Frobenius norm and ℓ_1-norm are defined as $\|\mathcal{A}\|_F = \sqrt{\sum_{ijk} a_{ijk}^2}$ and $\|\mathcal{A}\|_1 = \sum_{ijk} |a_{ijk}|$, respectively.

A. Deep Image Prior

DIP was proposed by Ulyanov *et al.* [31] and the main message of the DIP framework is the capability that the CNN

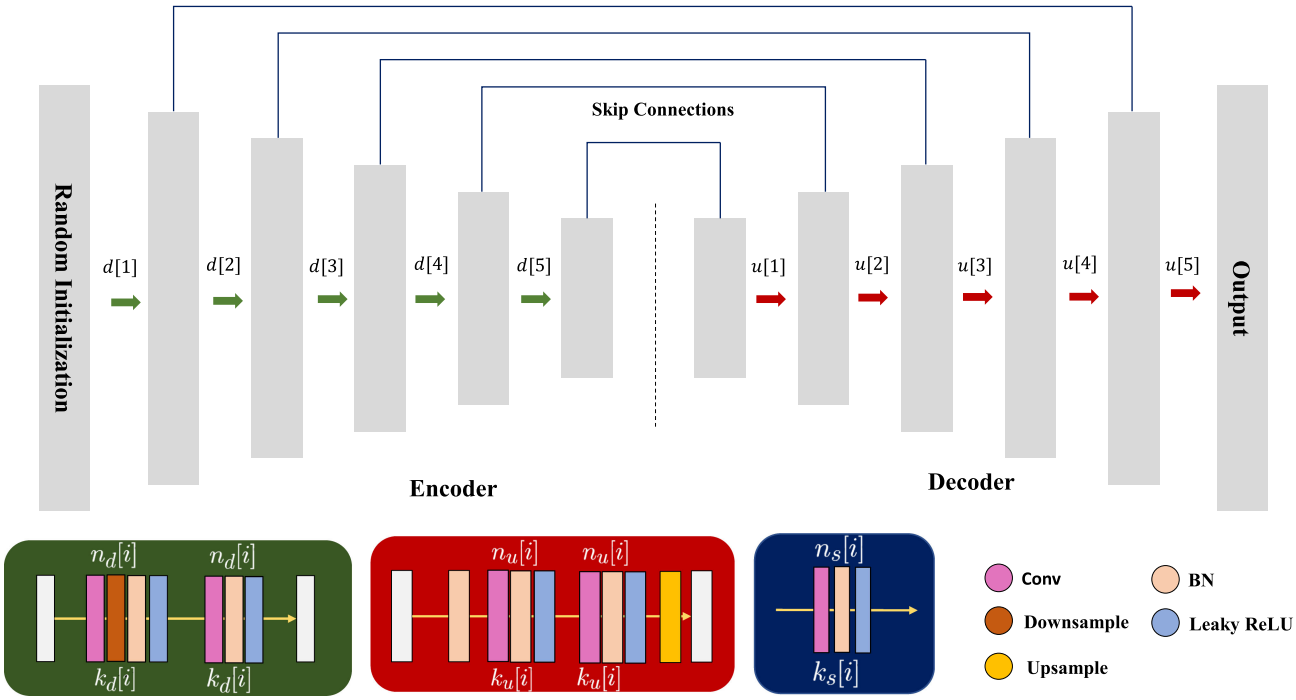


Fig. 2. DIP architecture is a U-Net like network with five downsampling blocks $d[i]$, five upsampling blocks $u[i]$, and five skip-connection blocks $s[i]$. The number of filters at depth i for the upsampling, downsampling and skip-connections are denoted by $n_u[i], n_d[i], n_s[i]$, respectively, and $k_u[i], k_d[i], k_s[i]$ represent the respective kernel sizes.

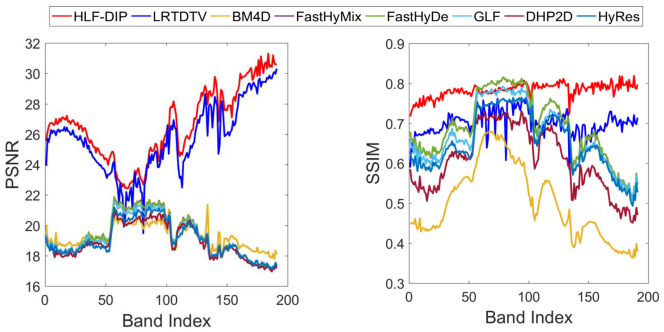


Fig. 3. Quantitative evaluation of various denoising methods (left plot for PSNR in dB and right plot for SSIM between 0 and 1) for every spectral band of Washington DC Mall dataset (Case 5).

can restore noise-degraded images without the need of training data. In addition, the structure of CNN can capture some inherent image attributes. So, it is possible to discharge the regularization from the denoising process. Hence, we can formulate an optimization problem for DIP as

$$\Theta^* = \underset{\Theta}{\operatorname{argmin}} \|\mathcal{Y} - f_\Theta(\mathcal{Z})\|_F^2 \text{ and } \mathcal{X} = f_{\Theta^*}(\mathcal{Z}) \qquad (1)$$

where f_Θ is a convolutional neural network (CNN) with parameter vector Θ, $\mathcal{Z} \in \mathbb{R}^{H\times W\times B}$ is the input of CNN (a random initialization), and $\mathcal{X}$, $\mathcal{Y} \in \mathbb{R}^{H\times W\times B}$ are the restored and observed images, respectively. Here, H and W depict the spatial size of the image and B is the number of channels (bands). With the differentiable objective function, the optimization problem (1) can be solved by a gradient-based algorithm. Due to the fact that (1) is a nonconvex optimization problem (because of the complex structure of CNN), surely there exist many local minimizers to the problem (1). A remedy for this issue is to properly set the maximum number of iterations for handling the optimization problem because it has been found that CNN first fits the signal part, and then after a large number of iteration steps, it starts to fit the noise part (overfitting). For this reason, one can come up with a suitable local minimizer through the trajectory of the optimization process. Finally, the image can be reconstructed by $\mathcal{X}^* = f_{\Theta^*}(\mathcal{Z})$.

B. Robust Statistics

The field of robust statistics arose due to the fact that classical statistics' parametric models are frequently not very good approximations of some critical events or errors as described in [37]–[40], which occur in the gathered data with a wide range of probability distributions. The study of robust statistics, particularly for non-normal distributions, focuses on how to deal with outliers or large-magnitude errors that defy the probability assumptions. The breakdown value (the greatest percentage of outlier data points without producing a nonnegligible variation in the solution) is a useful measure of robustness [41], because a single outlier in one data sample may lead the least-squares approach to come up with a breakdown value of 0% for the resulting estimated denoised data, while this value is much higher for robust estimators. Next, let us briefly review a class of estimators and probability distribution on which the proposed denoising algorithm is based.

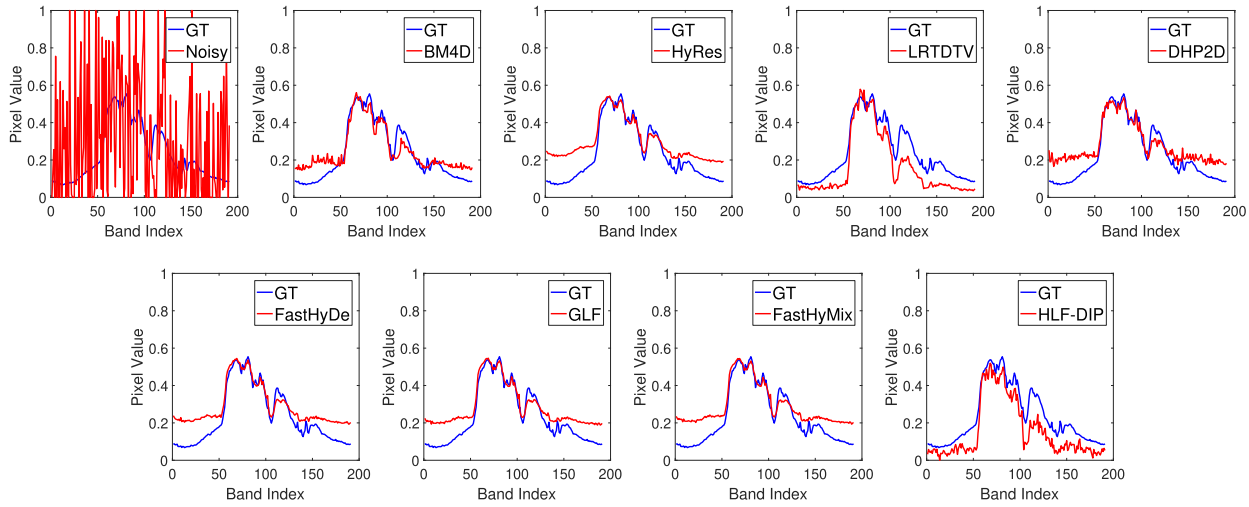


Fig. 4. Restored spectral data by various denoising methods for the pixel (120,130) of Washington DC Mall dataset (**Case 5**).

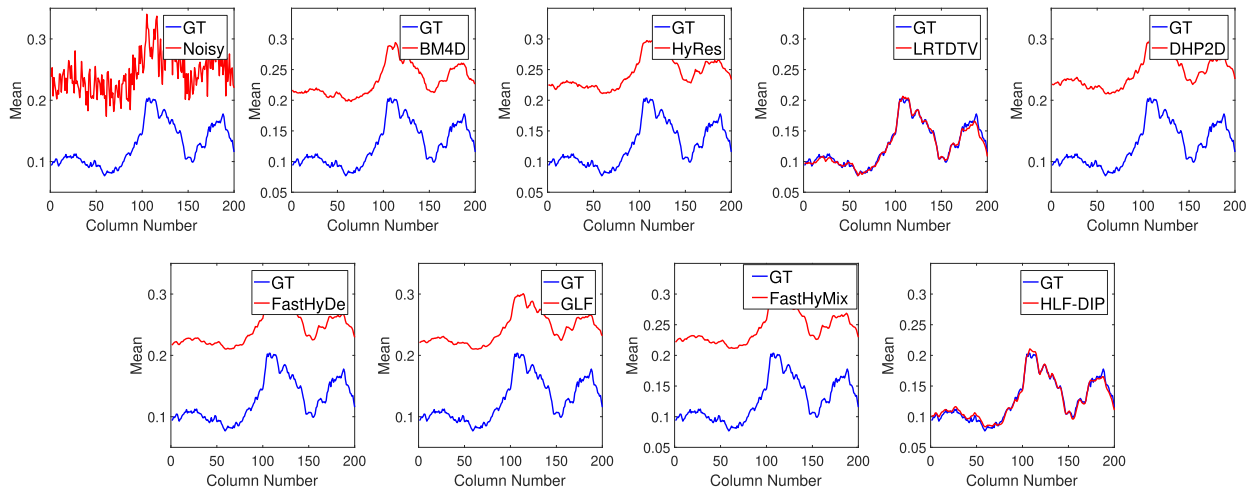


Fig. 5. Vertical mean profiles of the restored data by various denoising methods for band 8 of Pavia University dataset (**Case 5**).

1) M-Estimators: Thanks to high breakdown value and efficiency, M-estimators [37], [39] presently are dominant approaches in robust estimation and they are a generalization of maximum likelihood estimators (MLEs), for which $\sum_i \rho(r_i)$ is minimized where r_i is a residual for data-fitting and $\rho(\cdot)$ is a function with the following desirable properties [37]–[40]:

1) Nonnegative ($\rho(r) \geq 0$) and $\rho(0) = 0$;
2) Symmetric ($\rho(-r) = \rho(r)$);
3) Monotonic (if $|r| \geq |s|$ then $\rho(r) \geq \rho(s)$).

One can easily observe that when $\rho(r_i) = r_i^2$, M-estimators will be identical to the least-squares (LS) approximation.

2) Least Favorable Distribution: Huber created a scaled version of the family $\mathcal{P}_\epsilon$ of ϵ-contaminated normal distributions [38]

$$\mathcal{P}_\epsilon = \{(1-\epsilon)\Phi + \epsilon\Psi : \Psi \in S\}$$

in which, $0 \leq \epsilon < 1$, $\Phi(t) = (2\pi)^{-\frac{1}{2}} \int_{-\infty}^{t} \exp\left(-\frac{1}{2}w^2\right) dw$ is the standard normal cumulative distribution, and S is the set of all symmetric probability distributions such that $\Psi(-t) = 1 - \Psi(t)$. This model is based on the assumption that the degradation process contains a fraction $(1-\epsilon)$ of Gaussian noise with unit variance and a fraction (ϵ) of gross error. Huber [37], [38] created the LFD by assuming that the distribution of the degraded data is an unknown member of the family $\mathcal{P}_\epsilon$ which has a probability density function (PDF) given by

$$p_\delta(t) = (1-\epsilon)\frac{1}{\sqrt{2\pi}} e^{-H_\delta(t)} \tag{2}$$

with δ depending on ϵ through

$$\frac{\epsilon}{1-\epsilon} = \frac{2}{\delta}\Phi'(\delta) - 2\Phi(-\delta) \tag{3}$$

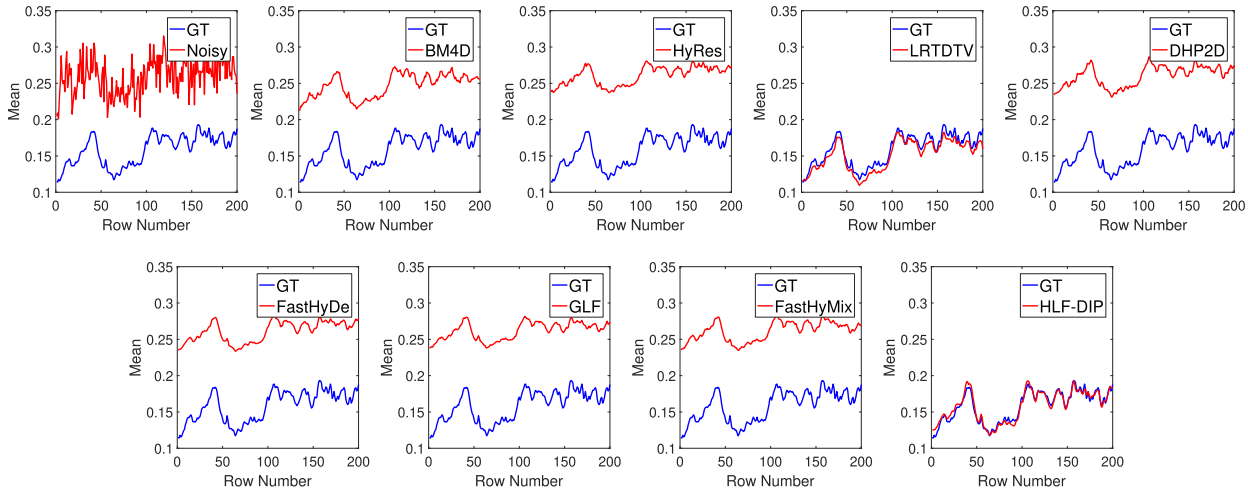


Fig. 6. Horizonal mean profiles of the restored data by various denoising methods for band 43 of Pavia University dataset (**Case 5**).

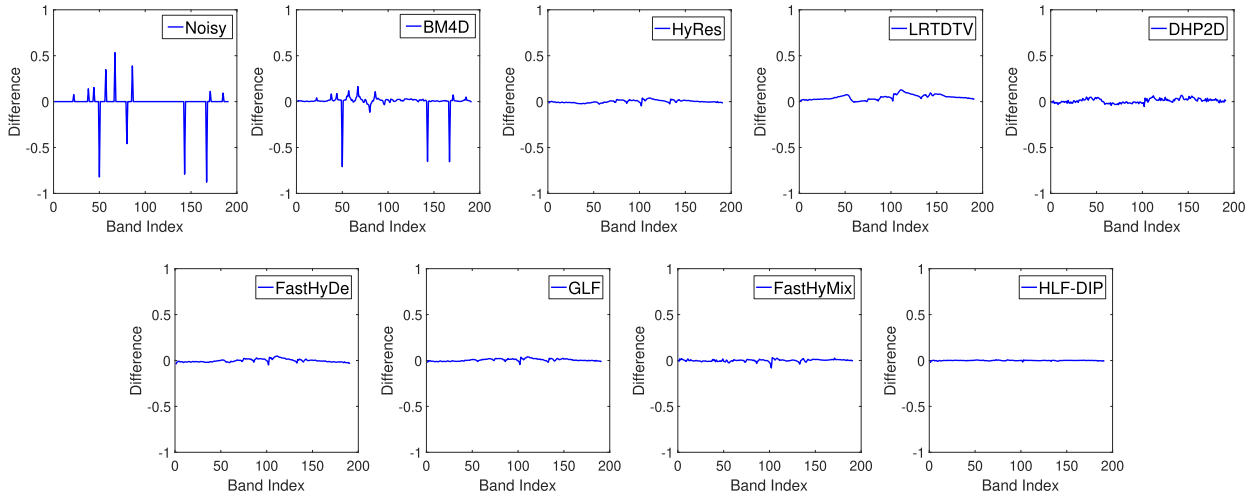


Fig. 7. Difference between the restored data and the GT versus spectral bands for pixel (120,130) of Washington DC Mall dataset (**Case 4**).

(yielded from $\int_{-\infty}^{\infty} p_\delta(t)dt = 1$) and $H_\delta(t)$ is Huber loss function (HLF) defined by

$$H_\delta(t) = \begin{cases} \frac{1}{2}t^2 & |t| \le \delta \\ \delta|t| \quad -\frac{1}{2}\delta^2 & |t| > \delta \end{cases} \tag{4}$$

where δ is a user-defined threshold. The HLF makes use of the advantages of least-squares error (LSE) and the least absolute deviation (LAD) in such a way that $p_\delta(t)$ behaves similar to normal density for small t, and to exponential density for large t. It is worth noting that for $\delta \to \infty$, (4) reduces to LSE. Outliers can also be handled by long tail distribution compared to normal distribution simultaneously. HLF is differentiable, and its derivation is the following clip function:

$$H_\delta{}'(t) = \begin{cases} t & |t| \le \delta \\ \delta \mathrm{sgn}(t) & |t| > \delta \end{cases} \tag{5}$$

where $\mathrm{sgn}(t)$ is the sign function. Since (5) is monotonically a nondecreasing function, H_δ is a convex function [42]. In particular, we would like to mention that edge-preserving and outlier removal are two key capabilities of the HLF [43]–[45].

III. Proposed HLF-DIP Denoising Method

In this section, first of all, we introduce the image degradation model, and then, we formulate the HSI denoising problem using the statistical approach. Then, the proposed denoising algorithm is presented to solve the formulated optimization problem, followed by a discussion for the network structure of the proposed framework.

A. Image Degradation Model

The observed HSI is assumed to be contaminated by Gaussian and sparse noise [19] as follows:

$$\mathcal{Y} = \mathcal{X} + \mathcal{G} + \mathcal{S} = \mathcal{X} + \mathcal{W} \tag{6}$$

in which, $\mathcal{Y}$, $\mathcal{X}$, $\mathcal{G}$, and $\mathcal{S} \in \mathbb{R}^{H\times W\times B}$ are the observed image, clean image, Gaussian noise, and sparse noise, respectively, and $\mathcal{W} = \mathcal{G} + \mathcal{S}$.

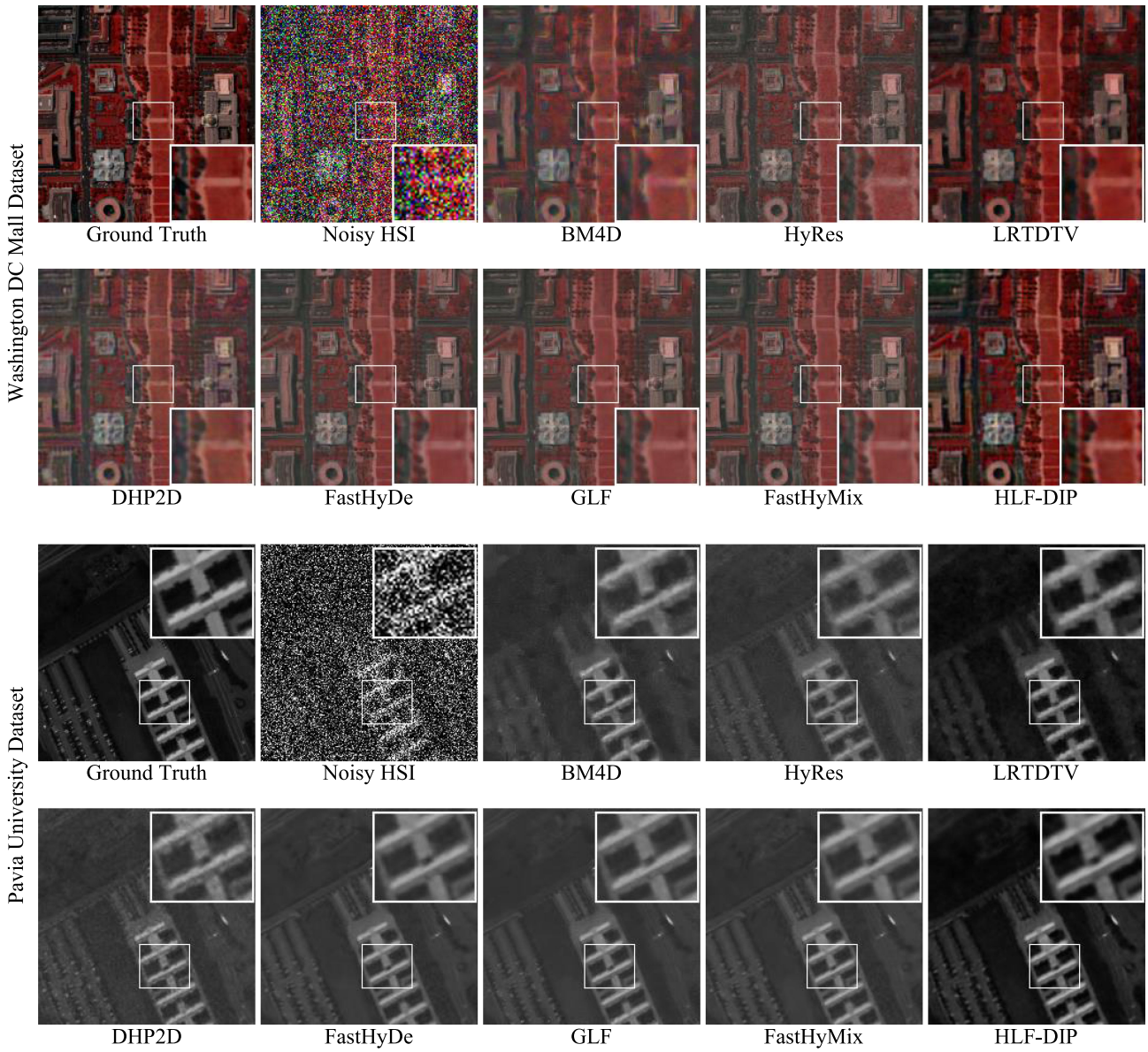


Fig. 8. Visual quality assessment of various methods for **Case 5**. The top two rows are the restored HSI data (bands 57, 27, 17) by pseudocolor for Washington DC Mall dataset, and the bottom two rows are the corresponding results (band 9) for Pavia University dataset.

B. Statistical Approach

The maximum *a posteriori* probability (MAP) approach is a widely used approach to restore the clean image $\mathcal{X}$ from the corrupted image $\mathcal{Y}$. The MAP estimate is to maximize the conditional PDF, that is

$$\mathcal{X}^* = \underset{\mathcal{X}}{\operatorname{argmax}} \log p(\mathcal{X}|\mathcal{Y})$$

where $p(\mathcal{X}|\mathcal{Y})$ is the conditional PDF of $\mathcal{X}$ given $\mathcal{Y}$. By Bayes' theorem, the estimated image can be expressed as

$$\mathcal{X}^* = \underset{\mathcal{X}}{\operatorname{argmin}} \{-\log p(\mathcal{Y}|\mathcal{X}) - \log p(\mathcal{X})\} \tag{7}$$

where $p(\mathcal{X})$ is the prior PDF of $\mathcal{X}$. With $p(\mathcal{Y}|\mathcal{X})$ modeled by (2) (i.e., PDF of $\mathcal{W} = \mathcal{Y} - \mathcal{X}$), the restored HSI image $\mathcal{X}^*$ can be equivalently written as

$$\mathcal{X}^* = \underset{\mathcal{X}}{\operatorname{argmin}} \{H_\delta(\mathcal{Y}, \mathcal{X}) + R(\mathcal{X})\} \tag{8}$$

where $H_\delta(\mathcal{Y}, \mathcal{X}) = \sum_{ijk} H_\delta(y_{ijk} - x_{ijk})$ (i.e., sum of HLFs defined by (4) for all $y_{ijk} \in \mathcal{Y}$ and $x_{ijk} \in \mathcal{X}$), and $R(\mathcal{X}) = -\log p(\mathcal{X})$.

Note that in (8), the first (second) term inside the braces corresponds to the data-fitting (regularization) term. Determining a suitable $R(\mathcal{X})$ is usually difficult due to the lack of reliable rules and schemes, so we use the DIP instead. By replacing $\mathcal{X}$ with a CNN output in (8) with the regularizer $R(\mathcal{X})$ dropped, we come up with

$$\Theta^* = \underset{\Theta}{\operatorname{argmin}}\, H_\delta(f_\Theta(\mathcal{Z}), \mathcal{Y}) \text{ and } \mathcal{X}^* = f_{\Theta^*}(\mathcal{Z}) \tag{9}$$

Algorithm 1: HLF-DIP.

Input: Noisy hyperspectral cube ($\mathcal{Y}$), t_{max} (max. no. of iterations), δ
Initialization: $t = 0$, Θ_0 and $\mathcal{Z}$ randomly given
while $t < t_{max}$ **do:**
Update Θ_{t+1} using Adam optimzer and the CNN backpropagation w.r.t. the Huber loss function $H_\delta(f_\Theta(\mathcal{Z}), \mathcal{Y})$.
$t := t + 1$
end while
Output: Denoised hyperspectral cube $\mathcal{X}^*$

where f_Θ is a CNN with parameter vector Θ, and $\mathcal{Z} \in \mathbb{R}^{H\times W\times B}$ denotes the input of CNN, which is generated randomly. It is notable that problem (8) and problem (10) are equivalent, where the latter is the commonly known optimization-based method with a convex surrogate $\|\mathcal{S}\|_1$ (the convex envelope of the nonconvex $\|\mathcal{S}\|_0$ for the suppression of sparse noise) used as the regularizer, as stated in the following proposition.

Proposition 1: The problem in (8) (with the regularization term $R(\mathcal{X}) = 0$) is equivalent to the following optimization problem:

$$\min_{\mathcal{X},\mathcal{S}} \frac{1}{2}\|\mathcal{Y} - \mathcal{X} - \mathcal{S}\|_F^2 + \delta\|\mathcal{S}\|_1 \tag{10}$$

Proof: The proof of Proposition 1 is given in Appendix. ■

It is worth mentioning that by Proposition 1, a trivial solution to (10) is $\mathcal{X}^* = \mathcal{Y}$ and $\mathcal{S}^* = \mathbf{0}$, so is $\mathcal{X}^* = \mathcal{Y}$ to (8) if the regularization term $R(\mathcal{X})$ is dropped. However, the restored HSI image $\mathcal{X}^* = f_{\Theta^*}(\mathcal{Z}) \neq \mathcal{Y}$ given by (9) depends on the CNN used, whose initial input $\mathcal{Z}$ is far different from $\mathcal{Y}$; meanwhile the sparse noise reduction has been implicitly taken into account by Proposition 1. Similarly, besides the additive noise reduction, other unknown characteristics of the ground truth (e.g., spatial smoothness, spectral correlations, and low-rankness) may also be captured blindly in the restored $\mathcal{X}^*$ given by (9) in spite of no regularizer used. This needs to be justified in the experiment.

Since HLF is differentiable, we can use any gradient-based algorithms to get a desired suboptimal solution $\mathcal{X}^*$ to (9). To update the network parameter Θ, we employ an adaptive moment estimation (Adam) optimizer [46]. The parameter vector Θ is iteratively updated for obtaining $\mathcal{X}^*$ and implemented in Algorithm 1, i.e., the proposed HLF-DIP algorithm. We would like to emphasize that it is an unsupervised denoising algorithm (without training) for obtaining the desired $\mathcal{X}^*$.

C. Convolutional Neural Network Architecture

As proposed in [31], the basis of the DIP network is based on a convolutional encoder-decoder (hourglass) with some skip connections. As shown in Fig. 2, we use five blocks for each encoder–decoder pair, and similar to most popular approaches, convolution (Conv), batch normalization (BN), and activation layer are employed. By using different filters, the convolutional layers can extract distinct spatial characteristics of the image $\mathcal{Y}$. A random noise realization (e.g., with a uniform distribution, $\mathcal{U}(0, 0.1)$) is generated as the CNN input, which is processed by the encoder and results in embedded features. The decoder reconstructs the HSI using the embedded features. Optionally, at each iteration $\mathcal{Z}$ is perturbed with an additive zero-mean Gaussian noise to impede the optimization process. We also employ 2-D convolution because of its superior performance compared to 3-D convolution [32]. It should be noted that the 2-D method does not employ bandwise processing, and it rather captures spectral information in the first convolutional layer's filters and their combinations in the ensuing layers [32]. The BN makes the CNN faster and more stable through the hyperparameter selection by recentering and rescaling. Note that BN was originally thought to reduce the learning rate of the network by moderating internal covariance shift [47], but new research revealed that it actually smoothens the objective function [48]. So, BN layers are often applied after or before a nonlinear activation function like Leaky ReLU [49] as shown in Fig. 2. The skip connection is utilized to solve the vanishing gradients issue in deep neural networks. For downsampling (upsampling), the stride within the convolution (bilinear interpolation) is utilized. For maintaining the image's size, reflection padding was employed in the convolution. The values of all hyperparameters for DIP are summarized in Table I. We empirically observed that adding the sigmoid activation function in the last layer of the DIP, always gives better performance and the yielded $\mathcal{X}^*$ is guaranteed for each data sample belonging to the interval [0,1].

TABLE I
DIP HYPERPARAMETERS USED IN THE EXPERIMENT

Hyperparameter	Value
$\mathcal{Z}$	$\mathcal{U}(0, 0.1)$
$n_d = n_u$	$[128, 128, 128, 128, 128]$
n_s	$[4, 4, 4, 4, 4]$
$k_d = k_u$	$[3, 3, 3, 3, 3]$
k_s	$[1, 1, 1, 1, 1]$
Optimizer	Adam
Learning rate	0.01
Activation function	Leaky ReLU
Leaky ReLU slope	0.01
Upsample mode	Bilinear
Downsample mode	Stride

D. Summary

The proposed HLF-DIP algorithm tries to search for a restored HSI image by using the intelligent CNN for unsupervised learning of the unknown prior information (e.g., spatial and spectral correlations, and other unknown characteristics) from the noisy data at each iteration ($t < t_{max}$), so that the restored image $\mathcal{X}^*$ is equipped with such attributes from the learned prior information. Therefore, a random initial input $\mathcal{Z}$ to the CNN is used by the proposed HLF-DIP without the need of any regularizer, which thus is also a user-friendly denoising algorithm. Finally, we would like to emphasize that the default architecture of DIP is employed just for justifying the function of robust statistics.

TABLE II
QUANTITATIVE ASSESSMENT OF VARIOUS METHODS APPLIED TO WASHINGTON DC MALL DATASET

Scenario	Metric	Noisy HSI	BM4D	HyRes	LRTDTV	DHP2D	FastHyDe	GLF	FastHyMix	HLF-DIP
Case 1	MPSNR	22.75	33.25	36.40	33.84	34.98	37.04	**37.15**	37.04	35.32
	MSSIM	0.602	0.933	0.970	0.942	0.960	**0.974**	**0.974**	**0.974**	0.963
	MSAM	0.367	0.113	0.084	0.084	0.091	0.076	**0.075**	0.077	0.083
	MFSIM	0.827	0.964	0.985	0.960	0.979	**0.987**	**0.987**	**0.987**	0.981
Case 2	MPSNR	15.58	26.10	27.46	29.08	26.68	27.96	27.85	27.94	**30.06**
	MSSIM	0.267	0.771	0.873	0.843	0.840	0.890	0.884	0.890	**0.894**
	MSAM	0.620	0.186	0.159	**0.121**	0.165	0.153	0.154	0.153	0.128
	MFSIM	0.647	0.876	0.950	0.905	0.935	**0.959**	0.955	**0.959**	0.948
Case 3	MPSNR	10.67	20.07	20.02	25.77	19.77	20.29	20.21	20.29	**26.89**
	MSSIM	0.104	0.537	0.687	0.710	0.633	0.723	0.704	0.723	**0.790**
	MSAM	0.796	0.268	0.266	0.164	0.270	0.260	0.261	0.260	**0.159**
	MFSIM	0.508	0.780	0.881	0.835	0.855	**0.902**	0.890	**0.902**	0.894
Case 4	MPSNR	17.38	20.98	31.88	35.78	30.27	32.45	32.59	37.78	**49.49**
	MSSIM	0.397	0.647	0.941	0.968	0.913	0.951	0.953	0.988	**0.998**
	MSAM	0.554	0.423	0.109	0.060	0.119	0.100	0.097	0.064	**0.024**
	MFSIM	0.842	0.885	0.974	0.985	0.961	0.988	0.980	0.988	**0.999**
Case 5	MPSNR	10.02	19.25	19.05	25.71	18.90	19.30	19.22	19.31	**26.66**
	MSSIM	0.089	0.501	0.652	0.701	0.606	0.686	0.688	0.688	**0.783**
	MSAM	0.818	0.282	0.284	0.173	0.287	0.279	0.280	0.277	**0.168**
	MFSIM	0.491	0.765	0.867	0.832	0.845	0.887	0.872	0.887	**0.890**
Case 6	MPSNR	19.57	26.94	30.48	32.05	29.70	32.18	32.35	33.34	**33.68**
	MSSIM	0.500	0.800	0.899	0.919	0.896	0.932	0.936	0.946	**0.954**
	MSAM	0.474	0.225	0.151	0.097	0.139	0.121	0.116	0.113	**0.093**
	MFSIM	0.774	0.909	0.950	0.955	0.948	0.967	0.969	0.971	**0.974**
Case 7†	MPSNR	8.13	25.02	28.81	26.35	28.76	30.02	**30.04**	30.03	29.68
	MSSIM	0.093	0.664	0.864	0.734	0.869	**0.909**	**0.909**	**0.909**	0.905
	MSAM	0.966	0.233	0.158	0.154	0.128	0.129	**0.125**	0.130	**0.125**
	MFSIM	0.423	0.825	0.924	0.843	0.935	**0.944**	**0.944**	**0.944**	0.939

Boldface (Underlined) Numbers in the Table Denote the Best (Second Best) Performance for the Corresponding Case and Performance Index.
† The simulated data for Case 7 are unclipped.

Surely, it is possible to further improve the proposed algorithm by using the neural architecture search reported in [50].

IV. EXPERIMENTAL RESULTS

A. Experimental Settings

Dataset: Experiments are conducted on two HSI datasets. The first one is the scene of Washington DC Mall,[1] which was collected by the Hyperspectral Digital Imagery Collection Experiment (HYDICE) sensor. This sensor has a 1-m spatial resolution and a 10-nm band spacing covering the spectral range of 400–2500 nm. Spectral bands between 0.9 and 1.4 μm where the atmosphere is opaque are omitted from the dataset, leaving 191 usable bands. In the experiment, we consider a patch of size $200 \times 200 \times 191$ (i.e., $H = W = 200$ and $B = 191$) from the Washington DC Mall image. The second dataset Pavia University[2] acquired by the ROSIS sensor during a flight campaign over Pavia, northern Italy. Pavia University has 103 spectral bands, but some of the samples in this dataset contain no information and must be deleted in advance. In the original Pavia University, 16 very low signal-to-noise ratio (SNR) bands were removed due to water vapor in the environment. Therefore, we consider a patch from the Pavia University image with a data size of $200 \times 200 \times 87$ (i.e., $H = W = 200$ and $B = 87$).

Performance Evaluation: The proposed HLF-DIP algorithm (i.e., Algorithm 1) is compared with seven state-of-the-art denoisers: BM4D [12], HyRes [51], LRTDTV [21], DHP2D [32], FastHyDe [18], GLF [52], FastHyMix [53] for six different cases. BM4D [12] is benefited by the grouping and collaborative filtering paradigm, which can effectively reduce the Gaussian noise. HyRes [51] is a parameter-free method based on a sparse low-rank model and ℓ_1-norm regularization for the least-squares problem. In LRTDTV [21], the tensor Tucker decomposition is used to capture the global spectral correlations and spatial-spectral total variation regularization (SSTV) to enhance the piecewise smooth structure in both spatial and spectral domains. Moreover, this framework is capable of suppressing sparse noise by adopting ℓ_1-norm regularization. DHP2D [32] is an unsupervised approach based on LSE and 2-D convolution. GLF [52] suggested a low-rank tensor factorization of nonlocal 3-D patches extracted from the given noisy HSI, yielding good-quality HSI denoising results. FastHyDe [18] exploits both the low-rank and spatial self-similarity information for sparse representations of HSI, leading to fast and high-quality

[1][Online]. Available: http://lesun.weebly.com/hyperspectral-data-set.html
[2][Online]. Available: http://ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes

TABLE III
QUANTITATIVE ASSESSMENT OF VARIOUS METHODS APPLIED TO PAVIA UNIVERSITY DATASET

Scenario	Metric	Noisy HSI	BM4D	HyRes	LRTDTV	DHP2D	FastHyDe	GLF	FastHyMix	HLF-DIP
Case 1	MPSNR	22.61	35.49	36.15	34.16	37.05	37.38	**37.63**	37.50	37.10
	MSSIM	0.433	0.933	0.939	0.916	0.951	0.957	**0.960**	0.956	0.951
	MSAM	0.388	0.085	0.077	0.078	0.068	0.068	**0.064**	0.066	**0.064**
	MFSIM	0.753	0.962	0.974	0.957	0.978	0.979	**0.981**	0.979	0.979
Case 2	MPSNR	15.65	27.96	27.88	30.53	28.64	28.66	28.47	28.66	**32.10**
	MSSIM	0.156	0.788	0.815	0.829	0.862	0.866	0.865	0.867	**0.886**
	MSAM	0.669	0.154	0.151	0.115	0.139	0.139	0.139	0.138	**0.091**
	MFSIM	0.542	0.887	0.922	0.905	0.939	**0.940**	0.937	0.939	**0.940**
Case 3	MPSNR	10.73	21.10	20.30	27.60	20.84	20.56	20.32	20.56	**29.80**
	MSSIM	0.056	0.581	0.622	0.716	0.689	0.695	0.675	0.694	**0.825**
	MSAM	0.825	0.231	0.235	0.158	0.224	0.224	0.227	0.224	**0.118**
	MFSIM	0.401	0.800	0.836	0.843	0.875	0.871	0.854	0.871	**0.903**
Case 4	MPSNR	17.39	20.03	31.30	37.02	32.01	32.30	32.53	40.54	**47.18**
	MSSIM	0.397	0.500	0.880	0.964	0.910	0.923	0.928	0.987	**0.992**
	MSAM	0.557	0.444	0.103	0.049	0.091	0.090	0.088	0.042	**0.025**
	MFSIM	0.794	0.839	0.950	0.985	0.963	0.963	0.967	0.993	**0.997**
Case 5	MPSNR	10.06	20.19	19.28	27.36	19.48	19.52	19.28	19.52	**29.39**
	MSSIM	0.049	0.549	0.587	0.705	0.626	0.661	0.638	0.659	**0.817**
	MSAM	0.843	0.242	0.245	0.163	0.240	0.238	0.241	0.238	**0.123**
	MFSIM	0.385	0.788	0.815	0.838	0.858	0.853	0.833	0.853	**0.894**
Case 6	MPSNR	19.20	26.71	30.29	33.42	32.38	31.85	32.36	32.65	**35.37**
	MSSIM	0.359	0.702	0.810	0.902	0.886	0.874	0.884	0.885	**0.943**
	MSAM	0.521	0.245	0.157	0.089	0.120	0.123	0.117	0.118	**0.073**
	MFSIM	0.680	0.873	0.918	0.950	0.958	0.934	0.949	0.941	**0.974**
Case 7†	MPSNR	8.13	27.11	28.34	27.70	29.83	**30.74**	30.67	30.70	30.63
	MSSIM	0.054	0.677	0.735	0.711	0.841	**0.857**	0.851	0.854	0.852
	MSAM	0.983	0.186	0.141	0.156	0.116	0.113	**0.110**	0.114	0.112
	MFSIM	0.322	0.826	0.870	0.840	0.908	**0.913**	0.910	**0.913**	0.911

Boldface (Underlined) Numbers in the Table Denote the Best (Second Best) Performance for the Corresponding Case and Performance Index.
† The simulated data for Case 7 are unclipped.

performance. FastHyMix [53] uses a Gaussian mixture model to characterize the complex distribution of mixed noise and take advantage of two major characteristics of hyperspectral data: low-rank in the spectral domain and high correlation in the spatial domain. The Gaussian mixture model can be used to better estimate the intensity of Gaussian noise and the location of sparse noise, thereby giving rise to promising performance.

In practice, the data dynamic range for each pixel is always constrained though it is not considered in some existing HSI denoising algorithms [18], [51]–[53]. However, without loss of generality, in our experiment, we consider the data on the normalized range [0,1] (i.e., minimum and maximum being 0 and 1, respectively) for the generation of noisy HSI data. The noise is added to the noise-free HSI $\mathcal{X}$ (within the interval [0,1] to generate the noisy HSI $\widehat{\mathcal{Y}} = \mathcal{X} + \mathcal{W}$ (which may not be fully within the interval [0,1]). Then as processed by digital image sensors, the clipped (or censored) observations $\mathcal{Y}$ in our work are obtained as [54], [55]:

$$y_{ijk} = \max(0, \min(\widehat{y}_{ijk} = x_{ijk} + w_{ijk}, 1).$$

It is noteworthy that clipping may cause a large PDF deviation from the perfect Gaussian noise [56], [57] if the standard deviation of the noise is large. Nevertheless, we would like to mention that the clipping effect resulting in a pixel value equal to zero or unity can be viewed as the case of an outlier or sparse noise (salt & pepper noise) in this pixel, thereby having been taken into account in the proposed denoising algorithm.

The observed HSI images are generated with Gaussian noise $\mathcal{N}(0, \sigma^2)$ and/or salt & paper noise, and stripe noise (artifacts) added in all bands of the noise-free images. The parameter $\delta = 0.001$ (cf., Section IV.D) is used for the proposed algorithm (except for **Case 7** where HSI is corrupted by perfect zero-mean Gaussian noise and $\delta = 1.3$ is used), while for the other algorithms under test, parameters are tuned based on suggested or default values according to the research papers and/or available source codes provided by the authors. We employ the PyTorch framework [58] to run the HLF-DIP on a PC with 128 GB RAM, an 11th Gen Intel(R) Core (TM) i9-11900 K @ 3.50 GHz CPU, and an NVIDIA GeForce RTX 3090 GPU in the simulated experiment for the following cases:

Cases 1–3: Gaussian noise with $\sigma = 20, 50, 100,$ respectively.

Case 4: Salt & pepper noise (5%).

Case 5: Gaussian noise with $\sigma = 100$ + salt & pepper noise (5%).

Case 6: Gaussian noise with $\sigma = 20$ + salt & pepper noise (1%) + stripes (the number of stripes being randomly chosen from 20 to 40 for each band).

Case 7: Gaussian noise with $\sigma = 100$ (unclipped data).

A suitable parameter t_{max} is used by the proposed HLF-DIP algorithm (cf. the discussion in Section IV.D-2), while t_{max} is set to 1200 for real noisy data experiments.

B. Quantitative Results

In order to investigate the robustness and effectiveness of all the algorithms under test, four performance indexes are used.

1) The mean peak signal-to-noise ratio (MPSNR), given in dB, used for measuring the spectral average of spatial qualities, is defined as

$$\text{MPSNR} = \frac{1}{B}\sum_{i=1}^{B} 10\log_{10}\left(\frac{\max(\mathbf{x}_i^2)}{\text{MSE}(\mathbf{x}_i, \mathbf{x}_i^*)}\right)$$

where B is the number of spectral bands, $\text{MSE}(\mathbf{x}_i, \mathbf{x}_i^*)$ is the mean squared error between bands of ground truth (GT) image $\mathbf{x}_i$ and reconstructed image $\mathbf{x}_i^*$. The larger the MPSNR, the better the performance.

2) The mean structural similarity index measure (MSSIM) [59] has been used for evaluating the presence of artifacts in the restored image and its value is between 0 and 1, with the unity value achieved when two images are identical. It is defined as

$$\text{MSSIM} = \frac{1}{B}\sum_{i=1}^{B} \frac{(2\mu_{\mathbf{x}_i}\mu_{\mathbf{x}_i^*} + C_1)(2\sigma_{\mathbf{x}_i\mathbf{x}_i^*} + C_2)}{(\mu_{\mathbf{x}_i}^2 + \mu_{\mathbf{x}_i^*}^2 + C_1)(\sigma_{\mathbf{x}_i}^2 + \sigma_{\mathbf{x}_i^*}^2 + C_2)}$$

where $\mu_{\mathbf{x}_i}$ and $\sigma_{\mathbf{x}_i}$ represent the ground truth image's mean value and standard deviation, respectively, $\sigma_{\mathbf{x}_i\mathbf{x}_i^*}$ is the covariance between $\mathbf{x}_i$ and $\mathbf{x}_i^*$, C_1 and C_2 are two constants used to verify that the outcomes are stable when $\sigma_{\mathbf{x}_i}^2 + \sigma_{\mathbf{x}_i^*}^2$ or $\mu_{\mathbf{x}_i}^2 + \mu_{\mathbf{x}_i^*}^2$ is close to zero.

3) Mean Spectral Angle Mapper (MSAM) calculates the mean angle (in radian) between spectrum vectors of the restored image and the reference image across all spatial positions [60]. Unlike the other performance indexes, smaller MSAM indicates better denoising performance.

4) Mean Feature Similarity Index Measure (MFSIM) (between 0 and 1) accounts for the perceptual consistency between the restored image and the GT based on the phase congruency and gradient magnitude [61]. Similar to MPSNR and MSSIM, the larger the MFSIM, the better the denoising performance.

The experimental results of the above four metrics obtained on the Washington DC Mall dataset and Pavia University dataset are summarized in Tables II and III, respectively. The best (second-best) values of MPSNR, MSSIM, MSAM, and MFSIM for each experiment are boldfaced (underlined).

Some observations from the two tables are given as follows. For the results of Case 1 through Case 3, the smaller the σ, the better the performance of each algorithm under test. The proposed algorithm performs best using the metrics MPSNR and MSSIM values for larger σ (Case 2 and Case 3 in Table II) for the Washington DC dataset, and best using all the four metrics (Case 2 and Case 3 in Table III) for the Pavia University dataset, though its performances for the other cases are very competitive with the best performance among the algorithms under test. On the other hand, for the results of Case 4 and Case 5, the proposed algorithm significantly outperforms all the other algorithms under test. The reason is that clipping effects for these two cases cause underexposure or overexposure in many noisy pixel values due to the large value of σ, and consequently, the resulting additive noise is no longer a perfect Gaussian noise assumed by the other algorithms. These results also justify that the CNN used in the proposed algorithm well captures image statistics, together with the edge-preserving and outlier removal capabilities of the HLF used by the proposed algorithm [43]–[45]. Although LRTDTV (that uses the SSTV regularizer) performs second-best in handling censoring effects, its parameter complexity is much higher than the proposed algorithm (without using any regularizer). The same observations from Case 4 and Case 5 also apply to Case 6 when the observed data is corrupted by some stripes in addition to low-magnitude Gaussian and sparse noise. Notably, because the standard deviation value σ of Gaussian noise is not very large for this case, the clipping effect is not too serious.

To further evaluate the effectiveness of HLF-DIP, let us present more details of restored spectral data for Case 5. Fig. 3 shows the results for the Washington DC Mall dataset in terms of PSNR and SSIM for every spectral band. As can be seen from Fig. 3, HLF-DIP performs best almost over all the bands and the performance improvement margin is also significant in most spectral bands. Furthermore, Fig. 4 shows the restored spectral data by various denoising methods for the pixel (120,130) in the spatial domain of the Washington DC Mall dataset, where the recovered spectral data by the LRTDTV and HLF-DIP methods are obviously better approximations of the GT than the other methods. It is worth mentioning that, because of low-rank or/and spatial-spectral regularizers used by LRTDTV, GLF, FastHyMix, FastHyDe, and HyRes, spectral correlations are certainly well-preserved by these methods. Besides, HLF-DIP performs best except for some zigzag patterns in the restored spectral data due to no regularizer used in HLF-DIP.

Next, we show the vertical mean profile and the horizontal mean profile of the restored HSI data in the spatial domain for Pavia University dataset for band 8 and band 43, in Figs. 5 and 6 (Case 5), respectively. In the presence of a mixture of large-magnitude Gaussian noise, outliers, along with clipping effects for this case, it can be seen that the noisy vertical mean profile and noisy horizontal profile are much larger than those of the GT. Nevertheless, HLF-DIP and LRTDTV produce noticeably better approximations to the GT than the other denoising methods, thereby demonstrating their superior performance.

To show the superior robustness of the proposed HLF-DIP algorithm over the other algorithms, the difference between the restored spectral curves and the spectrum of the GT pixel (120,130) in the spatial domain obtained from the Washington DC Mall dataset is displayed in Fig. 7 for Case 4. One can observe from this figure that the salt & pepper noise is actually composed of those outliers occurring only in some sparse bands. Obviously, BM4D is quite sensitive to outliers. All the other methods can cope with outliers, while the proposed HLF-DIP denoising method shows the least difference almost over all the

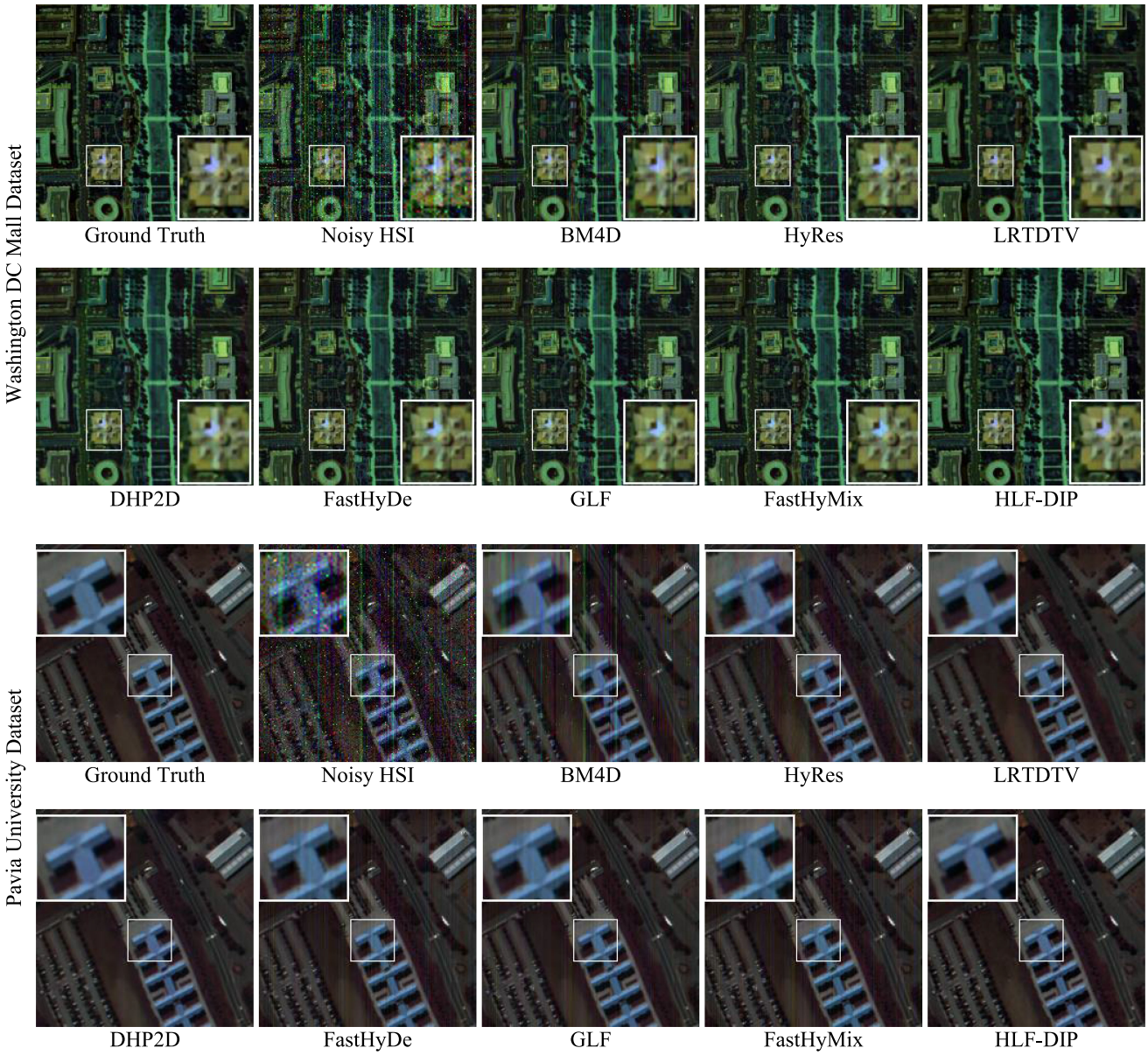


Fig. 9. Visual quality assessment of various methods for **Case 6**. The top two rows are the restored HSI data (bands 20, 50, 150) by pseudocolor for Washington DC Mall dataset, and the bottom two rows are the corresponding results (bands 70, 40 30) for Pavia University dataset.

spectral bands, demonstrating its best performance thanks to robust statistics used.

For visual quality assessment of all the algorithms under test, some results for the two datasets are illustrated in Fig. 8 (Case 5) and Fig. 9 (Case 6) that also show the best performance of the proposed denoising algorithm. These experimental results demonstrate that the overall performance of the proposed denoising algorithm is much superior to the other algorithms under test.

As for the simulation results for Case 7 (which and Case 3 are for strong Gaussian noise except for no clipping for the former in the data generation) in Tables II and III, it can be seen that DHP2D, FastHyDe, GLF, FastHyMix, and HLF-DIP have comparable performances. Overall, GLF (FastHyDe) performs best for Washington DC (Pavia University) dataset. However, each algorithm under test performs worse for Case 3 than for Case 7 to a different degree; nevertheless, the proposed HLF-DIP performs much better than the other algorithms, further justifying its robustness against the clipping effect caused by the sensor (which, as previously discussed in the data generation of Section IV-A, can be treated as the case of outliers or sparse noise).

C. Real Noisy Data Experiments

This section assesses the proposed method's performance on two real HSI data sets:

Indian Pines Dataset and HYDICE Urban Dataset. The former was collected by the AVIRIS sensor over the Indian Pines test site in North-western Indiana, acquiring a 145×145 HSI

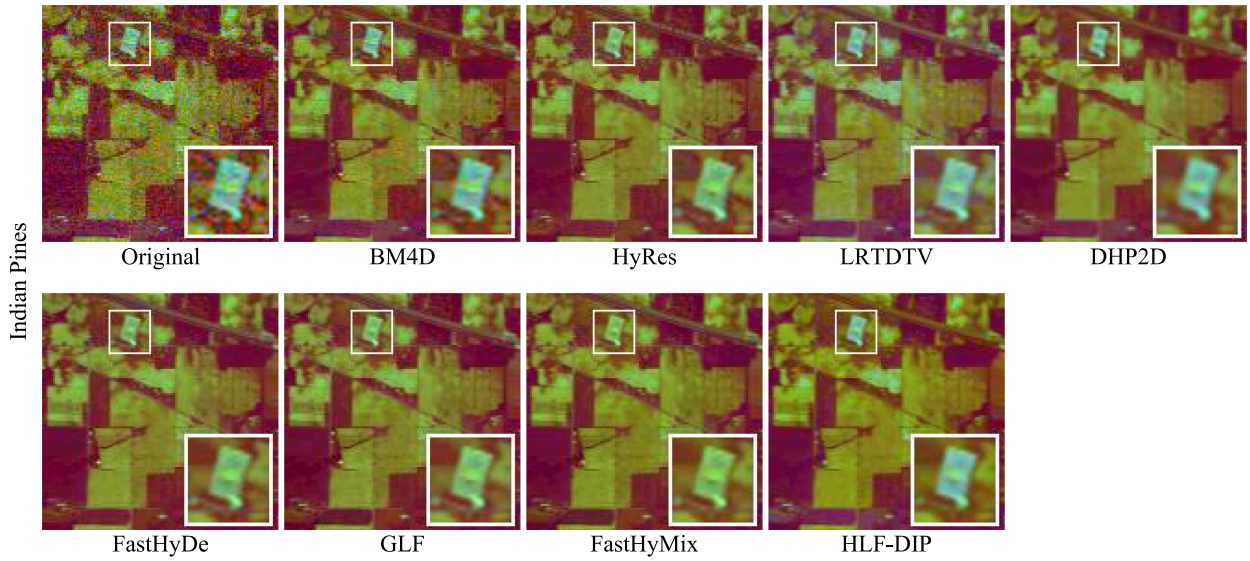


Fig. 10. Restored spectral data (bands 150, 30, 1) by pseudocolor for the Indian Pines dataset for visual quality assessment of various denoising methods.

(i.e., $H = W = 145$) with 224 spectral reflectance bands over the wavelength range 0.4–2.5 μ meters. Besides, there are noisy bands in this dataset (see the original pseudocolor image on the top left in Fig. 10). We consider a patch of size $128 \times 128 \times 224$ (i.e., $H = W = 128$ and $B = 224$), which, itself, is a noisy image. The second dataset for a real noisy data (see the top left of the first two rows and that of the bottom two rows in Fig. 11) experiment is the HYDICE Urban. In this dataset, there are 210 spectral bands ranging from 400 to 2500 nm, resulting in a spectral resolution of 10 nm. Gaussian noise, stripe noise, and mixed noise of these two types are the most common noise types in this dataset.

As seen from Fig. 10, the BM4D approach fails to restore the image completely. Even though HyRes can reduce some noise intensity, there is still appreciable noise remaining. DHP2D performs well in noise suppression but the restored image is oversmoothed and somewhat the edges are deformed. FastHyDe, GLF, and FastHyMix perform well in noise suppression. However, due to the subspace projection used by them, some features are missing in the restored images as shown in Fig. 10. In comparison with the other algorithms under test, the proposed HLF-DIP method and LRTDTV achieve a visually better and competitive restoration outcome in the Indian Pines dataset, and moreover, HLF-DIP has better performance in maintaining the color intensity of the original data.

On the other hand, for the Urban dataset, as shown in the top two rows for band 109 in Fig. 11, all the denoising methods can degrade the noise level in recovered images. However, some stripes remain in the restored images obtained by BM4D, HyRes, DHP2D, and FastHyDe. Among them, BM4D performs worst in both noise reduction and stripes suppression. LRTDTV can reduce stripes and noise noticeably, however, the restored image has some stripes left. GLF, FastHyMix, and HLF-DIP can restore the image well and maintain more details in the meantime. However, the proposed HLF-DIP method performs slightly better in maintaining the details of the image. Moreover, the obtained vertical mean profiles of the original image and recovered images shown in Fig. 11 for the Urban dataset are provided in Fig. 12. One can observe from Fig. 12, that "noisy" oscillation patterns resulting from the image inherent features and the mixed noise become much more smooth after denoising, i.e., the image features are maintained well in addition to noise suppression by GLF, FastHyMix, and HLF-DIP, while the proposed HLF-DIP performs best.

Next, the corresponding results for band 107 obtained by the denoising algorithms under test are shown in the bottom two rows in Fig. 11. One can see that the original image is seriously corrupted by strip artifacts and Gaussian noise, and that GLF and FastHyMix can produce the images with a nice quality of image details and noise reduction, owing to the spectral correlation taken into consideration. It is notable that the proposed HLF-DIP can almost completely remove stripes and mixed noises, while some stripes still remain in the restored HSIs obtained by GLF and FastHyMix in spite of more image details seemingly maintained. However, the other five algorithms under test apparently fail to denoise the original noisy image. The good performance of the proposed HLF-DIP algorithm can also be collectively justified and supported by the corresponding features for bands 109 and 107 in Fig. 11. Therefore, the above simulated and real data experiments have demonstrated the efficacy of the proposed HLF-DIP denoising algorithm.

D. Parameter Analysis

1) Choice of Parameter δ for HLF: Model-based methods for HSI denoising consider data-fitting and some regularization terms in the objective function of the optimization problem. For instance, in LRTDTV [21], five regularization parameters (Frobenius norm regularization, TV regularization, ℓ_1-norm regularization, rank regularization for 3-D hyperspectral cube, and SSTV regularization) need to be tuned and noise level must be identified for obtaining the best result. BM4D [12]

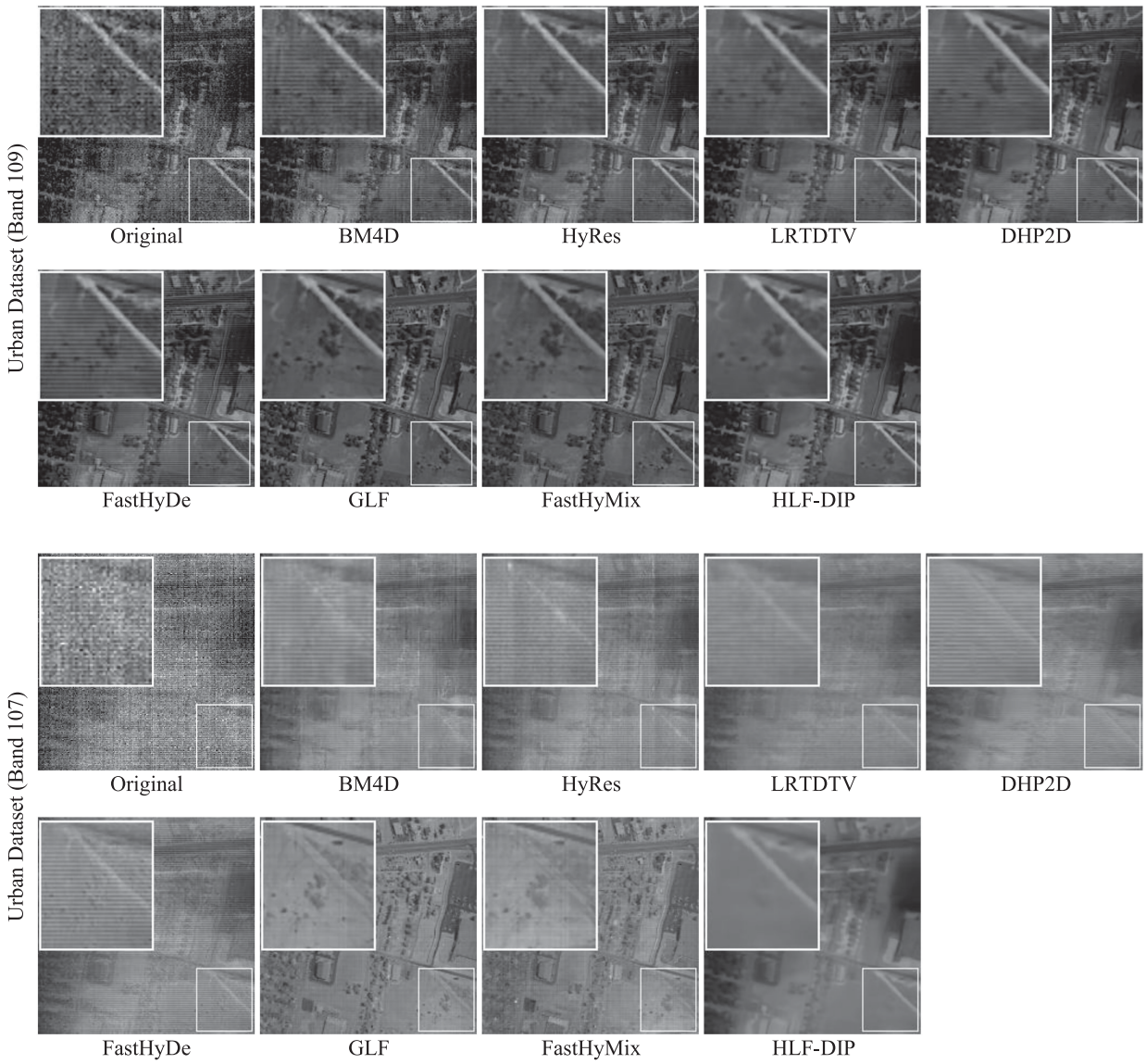


Fig. 11. Restored spectral data (band 109 for the top two rows and 107 for the bottom two rows) of the Urban dataset for visual quality assessment of various denoising methods.

includes a number of settings that need to be properly tuned such as hard-thresholding, block matching, and Wiener filtering, and furthermore, in order to achieve high-quality denoising, it requires prior information of the noise (either Gaussian or Rician) and standard deviation which is frequently unknown in real case scenarios.

In (9), the user-defined threshold δ for HLF needs to be tuned manually. By (3), the optimal value for δ can be defined when the fraction of contamination ϵ is known. Although such information may not be available in general, some intuitive analysis is instrumental as follows. For small (large) δ, HLF is essentially more (less) robust to large-magnitude noises and outliers (small-magnitude noise, e.g., Gaussian noise). In other words, δ corresponds to a tradeoff parameter for the denoiser to predominantly suppress either large-magnitude noise (with small δ) or small-magnitude noise (with large δ). The thresholds for obtaining 95% (85%) asymptotic relative efficiency (ARE) for the Gaussian noise scenario are $\delta = 1.345$ ($\delta = 0.7317$) [38]. Furthermore, by Proposition 1, δ value in the optimization problem (9) is also used as ℓ_1-norm regularization parameter for suppressing the sparse noise in the optimization problem (10). Therefore, we can choose $\delta = \frac{1}{\sqrt{HW}}$ as a suitable starting value suggested in [62]. For validation, we conducted the same experiments for all $\delta \in \{0.001, 0.01, 0.1.0.2, 0.7, 1.3\}$ for the investigation of the effects of δ on the quality of the restored HSIs. We empirically found that in the presence of outliers, choosing a very small δ always yields a better result. So, when the noisy image involves sparse noises, it is proper to choose a small δ. In addition, we observed that for the case of the relatively small-magnitude noise ($\sigma = 20$), the quality

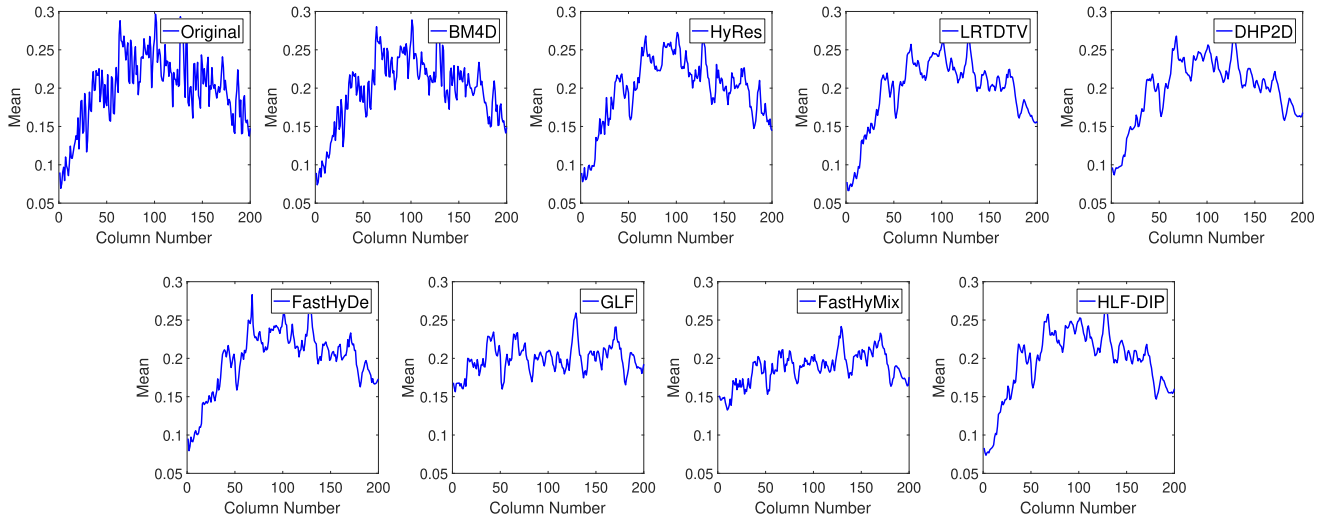


Fig. 12. Vertical mean profiles of the results (band 109 for the Urban dataset) shown in the top two rows in Fig. 11.

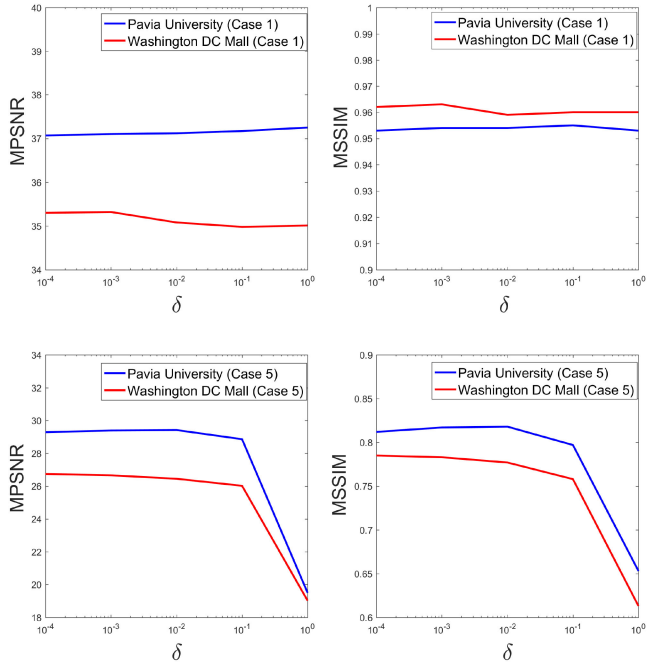


Fig. 13. Performance (MSPNR (dB) and MSSIM [between 0 and 1)] of the proposed HLF-DIP denosing algorithm versus δ for **Case 1** (top row) and **Case 5** (bottom row) of the simulated datasets (Pavia University and Washington DC).

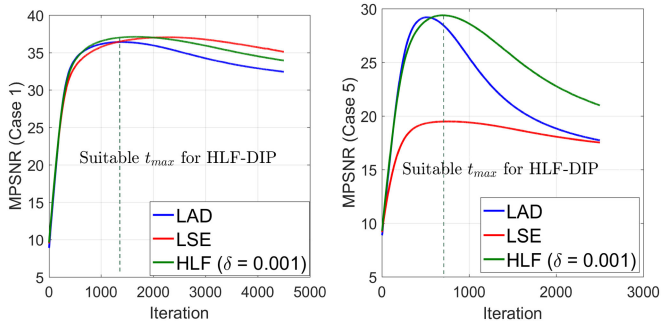


Fig. 14. MPSNR (in dB) performances (versus iteration) for Case 1 (left plot) and Case 5 (right plot) on the Pavia University dataset, using the three loss functions, LAD (ℓ_1-norm), LSE (ℓ_2-norm square), and HLF (fusion of ℓ_2-norm square and ℓ_1-norm), respectively.

difference of restored HSI between $\delta = 0.001$ and $\delta = 0.7$ is not much, though small δ yields better results. The reason for this is that the small-magnitude noise also implies low-power noise in the given noisy HSI. Though a small δ is preferred, $\delta = 0$ is not admissible simply due to the corresponding HLF being identical to zero.

Fig. 13 shows the MPSNR and MSSIM performances of the proposed HLF-DIP denoising algorithm versus δ ($10^{-4} \sim 10^{0} = 1$) for Case 1 (top row) and Case 5 (bottom row) with the same simulated data described in Section IV-A. One can see from this figure, that its performance is approximately piecewise linear with small variation for all δ in Case 1 and for $\delta < 10^{-1}$ in Case 5, but it drops linearly with δ for $\delta > 10^{-1}$ at a perceivable rate in Case 5. Therefore, a small value for δ is always a good choice, that accounts for $\delta = 10^{-3}$ used in our simulated experiment.

2) Choice of the Maximum Number of Iterations t_{max}: One common issue of DIP-based unsupervised denoising methods including the proposed HLF-DIP is that a reliable early-stopping criterion is not yet existent. Instead, one may need to find a good value of t_{max} from the visual quality assessment of the restored HSI data. This is certainly feasible for the simulated experiment because the performance index of the restored HSI data can provide the information of choosing a good value of t_{max} in an iterative fashion as done in the preceding simulated experiment.

To further qualitatively illustrate the role of t_{max}, Fig. 14 shows some MPSNR performances versus iteration t using the three loss functions, LAD (ℓ_1-norm of $\mathcal{E} \triangleq \mathcal{Y} - f_\Theta(\mathcal{Z})$, LSE ($\ell_2$-norm square of $\mathcal{E}$), and HLF (fusion of ℓ_1-norm for a subset $\mathcal{E}_1 \subset \mathcal{E}$ and ℓ_2-norm square for the complementary subset $\mathcal{E}_2 \subset \mathcal{E}$) with $\delta = 0.001$, for the Pavia University dataset. Some observations from this figure are as follows. HLF operates like LAD over some initial iterations due to large $|e_{ijk}| > \delta$ (the (i,j,k)th entry of the tensor $\mathcal{E} \subset \mathbb{R}^{H\times W\times B}$ occurring in most entries of $\mathcal{E}$), i.e., their performance curves are quite close,

and then switch its role as LSE later when $|e_{ijk}| < \delta$ is true in most entries of $\mathcal{E}$. The three peak values of the three respective performance curves are quite close for Case 1 (small-magnitude noise) though LSE and HLF work better than LAD for Case 1. While for Case 5 (large-magnitude noise), the two peak values are close for LAD and HLF and they are significantly larger than for LSE, thus yielding the best results for the Pavia University as shown in Table III. We would like to emphasize that the larger MPSNR indicates the smaller error $\mathcal{E}' \triangleq \mathcal{X} - f_\Theta(\mathcal{Z})$ between the GT image and the restored image. In other words, $\mathcal{E}$ decreases with t (iteration number), while $\mathcal{E}'$ decreases with t for $t < t_{max}$ but increases with t for $t > t_{max}$, indicating that the CNN starts to mistake some noise for the restored image (i.e., overfitting) for $t > t_{max}$, hence the optimization process should be stopped at iteration t_{max}, e.g., via visual quality assessment. Nevertheless, we would like to emphasize that when the level of noise is known, one can use the simple stopping criterion method proposed in [63] as mentioned by [31]. However, finding a reliable automatic stopping mechanism of DIP (i.e., the choice of t_{max} of the proposed HLF-DIP) is still an open issue [64]–[66], which is left for a future study.

V. Conclusion

We have presented an unsupervised HSI denoising algorithm (HLF-DIP algorithm) under the DIP framework (see Fig. 1) by minimizing the HLF (parameterized by a single parameter δ) without any regularizer and free from pretraining. The proposed algorithm shows the edge-preserving and outlier removal capabilities under the scenario of mixed types of noises (e.g., Gaussian noise, sparse noise), and is more user-friendly than most existing methods thanks to no regularizers. Extensive experimental results have demonstrated that the overall performance of the proposed algorithm is much superior to seven state-of-the-art methods.

Some further studies left in the future include

1) a reliable scheme or rule for the choice of the parameter t_{max};
2) other M-estimators for HSI denoising;
3) more powerful CNN architecture for HSI denoising;
4) application to other types of image data, such as biomedicine and synthetic-aperture radar; and
5) generalization of the proposed algorithm to other HSI inverse problems such as inpainting, deblurring, and unmixing.

Appendix

Proof of Proposition 1

Proof: Because both problem (8) without the term $R(\mathcal{X})$ and problem (10) are actually pixelwise decoupled, it suffices to show that they are equivalent for any pixel and spectral band. Assume that y, x, and s are the associated elements in $\mathcal{Y}$, $\mathcal{X}$ and $\mathcal{S}$ for a pixel in the data cube. Hence, we can rewrite problem (10) as

$$\min_{x,s} J(x,s) = \min_x \{\min_s J(x,s)\} \tag{11}$$

where

$$J(x,s) = \frac{1}{2}(y-x-s)^2 + \delta|s|. \tag{12}$$

The inner minimization problem in (11) is an unconstrained convex problem while the objective function $J(x,s)$ is differentiable except for $s=0$. Provided that $s^*=0$, it can be found by solving

$$\frac{dJ(x,s)}{ds} = s - (y-x) + \delta \mathrm{sgn}(s) = 0. \tag{13}$$

By (13) we have

$$s^\star = (y-x) - \delta \mathrm{sgn}(s). \tag{14}$$

Next, we solve $s^\star$ for the three cases $|y-x| > \delta$, $|y-x| < \delta$, and $|y-x| = \delta$, respectively.

Case 1 ($|y-x| > \delta$): It can be inferred from (14), that

$$s^\star \triangleq \begin{cases} (y-x) - \delta > 0 & \text{if } y - x > \delta \\ (y-x) + \delta < 0 & \text{if } y - x < -\delta. \end{cases} \tag{15}$$

Substituting (15) into (12) results in

$$J(x, s^\star) = \delta|y-x| - \frac{1}{2}\delta^2. \tag{16}$$

Case 2 ($|y-x| < \delta$): It can be inferred from (14) that if $s^\star > 0$, then $s^\star = y - x - \delta < 0$ (which is impossible); if $s^\star < 0$, then $s^\star = y - x + \delta > 0$ (which is impossible); hence we have the unique solution $s^\star = 0$ since $J(x,s)$ is convex and coercive, and thus

$$J(x, s^\star = 0) = \frac{1}{2}(y-x)^2. \tag{17}$$

On the other hand, it can be shown that the directional derivative at $s^\star = 0$ is given by

$$\begin{aligned} J'(s^\star = 0; v) &= \liminf_{\lambda \downarrow 0} \frac{J(\lambda v) - J(0)}{\lambda} \\ &= -(y-x)v + \delta|v| \geq 0 \quad \forall v \in \mathbb{R} \end{aligned} \tag{18}$$

thereby justifying that $s^* = 0$ is indeed the unique minimizer (surely the unique stationary point) for this case.

Case 3 ($|y-x| = \delta$): For this case, it can be readily inferred that

$$\begin{aligned} J(x,s) &= \frac{s^2}{2} - ((y-x)s - |y-x| \cdot |s|) + \frac{(y-x)^2}{2} \\ &\Rightarrow \min_{s \in \mathbb{R}} \{J(x,s)\} = J(x, s^* = 0) = \frac{1}{2}(y-x)^2 \end{aligned} \tag{19}$$

where the implication is due to $(y-x)s - |y-x| \cdot |s| \leq 0$ for any $s \in \mathbb{R}$.

According to the resulting $J(x, s^\star)$ [cf. (16)] in Case 1, (cf. (17)) in Case 2, and [cf. (19)] in Case 3, one can conclude

$$J(x, s^\star) = \begin{cases} \frac{1}{2}(y-x)^2 & |y-x| \leq \delta \\ \delta|y-x| - \frac{1}{2}\delta^2 & |y-x| > \delta \end{cases}$$

which is exactly the HLF $H_\delta(y-x)$, implying that problem (10) is equivalent to problem (8) (with the regularization term dropped). ■

REFERENCES

[1] J. M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. Nasrabadi, and J. Chanussot, "Hyperspectral remote sensing data analysis and future challenges," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 2, pp. 6–36, Jun. 2013.

[2] G. Lu and B. Fei, "Medical hyperspectral imaging: A review," *J. Biomed. Opt.*, vol. 19, no. 1, pp. 1–24, 2014.

[3] D.-W. Sun, *Hyperspectral Imaging for Food Quality Analysis and Control*. New York, NY, USA: Elsevier, 2010.

[4] C.-I. Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*. vol. 1, Berlin, Germany: Springer Science & Business Media, 2003.

[5] B. Rasti, B. Koirala, P. Scheunders, and P. Ghamisi, "How hyperspectral image unmixing and denoising can boost each other," *Remote Sens.*, vol. 12, no. 11, 2020, Art. no. 1728.

[6] T. Akgun, Y. Altunbasak, and R. M. Mersereau, "Super-resolution reconstruction of hyperspectral images," *IEEE Trans. Image Process.*, vol. 14, no. 11, pp. 1860–1875, Nov. 2005.

[7] W.-K. Ma *et al.*, "A signal processing perspective on hyperspectral unmixing: Insights from remote sensing," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 67–81, Jan. 2014.

[8] B. Rasti, P. Scheunders, P. Ghamisi, G. Licciardi, and J. Chanussot, "Noise reduction in hyperspectral imagery: Overview and application," *Remote Sens.*, vol. 10, no. 3, 2018, Art. no. 482.

[9] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 60–65.

[10] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D, Nonlinear Phenomena*, vol. 60, no. 1–4, pp. 259–268, 1992.

[11] H. Othman and S.-E. 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, Feb. 2006.

[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, Jan. 2013.

[13] H. K. Aggarwal and A. Majumdar, "Hyperspectral image denoising using spatio-spectral total variation," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 3, pp. 442–446, Mar. 2016.

[14] J. Xue, Y. Zhao, W. Liao, and S. G. Kong, "Joint spatial and spectral low-rank regularization for hyperspectral image denoising," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 1940–1958, Apr. 2018.

[15] J. Xue, Y. Zhao, W. Liao, and J. C.-W. Chan, "Nonlocal low-rank regularized tensor decomposition for hyperspectral image denoising," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 5174–5189, Jul. 2019.

[16] W. Dong, G. Shi, and X. Li, "Nonlocal image restoration with bilateral variance estimation: A low-rank approach," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 700–711, Feb. 2013.

[17] W. He, N. Yokoya, and X. Yuan, "Fast hyperspectral image recovery of dual-camera compressive hyperspectral imaging via non-iterative subspace-based fusion," *IEEE Trans. Image Process.*, vol. 30, pp. 7170–7183, 2021.

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

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

[20] M. Golbabaee and P. Vandergheynst, "Hyperspectral image compressed sensing via low-rank and joint-sparse matrix recovery," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2012, pp. 2741–2744.

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

[22] M. Wang, Q. Wang, J. Chanussot, and D. Li, "Hyperspectral image mixed noise removal based on multidirectional low-rank modeling and spatial–spectral total variation," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 488–507, Jan. 2021.

[23] Y.-B. Zheng, T.-Z. Huang, X.-L. Zhao, Y. Chen, and W. He, "Double-factor-regularized low-rank tensor factorization for mixed noise removal in hyperspectral image," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 12, pp. 8450–8464, Dec. 2020.

[24] D. Cerra, R. Müller, and P. Reinartz, "Unmixing-based denoising for destriping and inpainting of hyperspectral images," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2014, pp. 4620–4623.

[25] W. Xie and Y. Li, "Hyperspectral imagery denoising by deep learning with trainable nonlinearity function," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 11, pp. 1963–1967, Nov. 2017.

[26] 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, Feb. 2019.

[27] 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, Dec. 2019.

[28] Q. Shi, X. Tang, T. Yang, R. Liu, and L. Zhang, "Hyperspectral image denoising using a 3-D attention denoising network," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 12, pp. 10348–10363, Dec. 2021.

[29] K. Wei, Y. Fu, and H. Huang, "3-D quasi-recurrent neural network for hyperspectral image denoising," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 363–375, Jan. 2021.

[30] F. Xiong, J. Zhou, Q. Zhao, J. Lu, and Y. Qian, "MAC-Net: Model aided nonlocal neural network for hyperspectral image denoising," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, 2022.

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

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

[33] H. V. Nguyen, M. O. Ulfarsson, and J. R. Sveinsson, "Hyperspectral image denoising using SURE-based unsupervised convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 4, pp. 3369–3382, Apr. 2021.

[34] 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.*, 2019, pp. 7715–7719.

[35] Z. Sun, F. Latorre, T. Sanchez, and V. Cevher, "A plug-and-play deep image prior," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 8103–8107.

[36] G. Mataev, P. Milanfar, and M. Elad, "DeepRED: Deep image prior powered by RED," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, 2019, pp. 1–10.

[37] P. J. Huber, *Robust Statistics*. vol. 523, Hoboken, NJ, USA: Wiley, Inc., 2004.

[38] P. J. Huber, "Robust estimation of a location parameter," in *Breakthroughs in Statistics*. Berlin, Germany: Springer, 1992, pp. 492–518.

[39] A. M. Zoubir, V. Koivunen, E. Ollila, and M. Muma, *Robust Statistics for Signal Processing*. Cambridge, U.K.: Cambridge Univ. Press, 2018.

[40] R. A. Maronna, R. D. Martin, V. J. Yohai, and M. Salibián-Barrera, *Robust Statistics: Theory and Methods (With R)*. Hoboken, NJ, USA: Wiley, 2019.

[41] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*. Hoboken, NJ, USA: Wiley, 2005.

[42] C.-Y. Chi, W.-C. Li, and C.-H. Lin, *Convex Optimization for Signal Processing and Communications: From Fundamentals to Applications*. Boca Raton, FL, USA: CRC Press, 2017.

[43] T. Lukić, J. Lindblad, and N. Sladoje, "Regularized image denoising based on spectral gradient optimization," *Inverse Problems*, vol. 27, no. 8, 2011, Art. no. 085010.

[44] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud, "Deterministic edge-preserving regularization in computed imaging," *IEEE Trans. Image Process.*, vol. 6, no. 2, pp. 298–311, Feb. 1997.

[45] D. Geman and G. Reynolds, "Constrained restoration and the recovery of discontinuities," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 3, pp. 367–383, Mar. 1992.

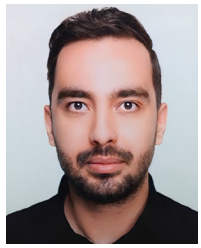
[46] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, San Diego, CA, USA, May 2015, pp. 1–15.

[47] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[48] S. Santurkar, D. Tsipras, A. Ilyas, and A. Mkadry, "How does batch normalization help optimization?," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 2488–2498.

[49] A. L. Maas *et al.*, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1–6.

[50] Y.-C. Chen, C. Gao, E. Robb, and J.-B. Huang, "NAS-DIP: Learning deep image prior with neural architecture search," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 442–459.
[51] B. Rasti, M. O. Ulfarsson, and P. Ghamisi, "Automatic hyperspectral image restoration using sparse and low-rank modeling," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 12, pp. 2335–2339, Dec. 2017.
[52] L. Zhuang, X. Fu, M. K. Ng, and J. M. Bioucas-Dias, "Hyperspectral image denoising based on global and nonlocal low-rank factorizations," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 12, pp. 10438–10454, Dec. 2021.
[53] L. Zhuang and M. K. Ng, "FastHyMix: Fast and parameter-free hyperspectral image mixed noise removal," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2021.3112577.
[54] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian, "Practical Poissonian–Gaussian noise modeling and fitting for single-image raw-data," *IEEE Trans. Image Process.*, vol. 17, no. 10, pp. 1737–1754, Oct. 2008.
[55] K. Zhang, W. Zuo, and L. Zhang, "FFDNet: Toward a fast and flexible solution for CNN-based image denoising," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4608–4622, Sep. 2018.
[56] R. Oten and R. J. de Figueiredo, "Adaptive alpha-trimmed mean filters under deviations from assumed noise model," *IEEE Trans. Image Process.*, vol. 13, no. 5, pp. 627–639, May 2004.
[57] T. Plotz and S. Roth, "Benchmarking denoising algorithms with real photographs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1586–1595.
[58] A. L. Maas *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037.
[59] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[60] F. A. Kruse *et al.*, "The spectral image processing system (SIPS)–interactive visualization and analysis of imaging spectrometer data," *Remote Sens. Environ.*, vol. 44, pp. 145–163, 1993.
[61] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, Aug. 2011.
[62] E. J. Candés, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?," *J. ACM*, vol. 58, no. 3, pp. 1–37, 2011.
[63] M. Burger, S. Osher, J. Xu, and G. Gilboa, "Nonlinear inverse scale space methods for image restoration," in *Proc. Int. Workshop Variational, Geometric, Level Set Methods Comput. Vis.*, 2005, pp. 25–36.
[64] Z. Cheng, M. Gadelha, S. Maji, and D. Sheldon, "A Bayesian perspective on the deep image prior," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5443–5451.
[65] H. Wang, T. Li, Z. Zhuang, T. Chen, H. Liang, and J. Sun, "Early stopping for deep image prior," 2021, *arXiv:2112.06074*.
[66] Y. Jo, S. Y. Chun, and J. Choi, "Rethinking deep image prior for denoising," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 5087–5096.



Keivan Faghih Niresi received the B.Sc. degree in electrical engineering from the University of Guilan, Rasht, Iran, in 2019. He is currently working toward the M.Sc. degree with the Institute of Communications Engineering, College of Electrical Engineering and Computer Science, National Tsing Hua University, Hsinchu, Taiwan.

His research interests lie primarily in convex optimization, machine learning, signal processing, and inverse problems, with applications in hyperspectral and medical imaging.

Chong-Yung Chi (Life Fellow, IEEE) received the B.S. degree from Tatung Institute of Technology, Taipei, Taiwan in 1975, the master's degree from National Taiwan University, Taipei, Taiwan in 1977, and the Ph.D. degree from the University of Southern California, Los Angeles, CA, USA, in 1983, all in electrical engineering.

He is currently a Professor of National Tsing Hua University, Hsinchu, Taiwan. He has published more than 240 technical papers (with citations more than 6500 times by Google-Scholar), including more than 90 journal papers (mostly in IEEE TRANSACTIONS ON SIGNAL PROCESSING), more than 140 peer-reviewed conference papers, 3 book chapters, and 2 books, including a textbook, *Convex Optimization for Signal Processing and Communications from Fundamentals to Applications*, CRC Press, 2017 (which has been popularly used in a series of invited intensive short courses at 10 top-ranking universities in Mainland China since 2010 before its publication). His current research interests include signal processing for wireless communications, convex analysis and optimization for blind source separation, biomedical and hyperspectral image analysis, and graph based learning and signal processing.

Dr. Chi received 2018 IEEE Signal Processing Society Best Paper Award, entitled "Outage Constrained Robust Transmit Optimization for Multiuser MISO Downlinks: Tractable Approximations by Conic Optimization," *IEEE Transactions on Signal Processing*, vol. 62, no. 21, Nov. 2014. He has been a Technical Program Committee member for many IEEE sponsored and co-sponsored workshops, symposiums and conferences on signal processing and wireless communications, including Co-Organizer and General Co-Chairman of 2001 IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC). He was an Associate Editor (AE) for four IEEE Journals, including IEEE TRANSACTIONS ON SIGNAL PROCESSING for 9 years (5/2001-4/2006, 1/2012-12/2015), and he was a member of Signal Processing Theory and Methods Technical Committee (SPTM-TC) (2005-2010), a member of Signal Processing for Communications and Networking Technical Committee (SPCOM-TC) (2011-2016), and a member of Sensor Array and Multichannel Technical Committee (SAM-TC) (2013-2018), IEEE Signal Processing Society.