Learning An Explicit Weighting Scheme for Adapting Complex HSI Noise

Xiangyu Rui[1], Xiangyong Cao[1], Qi Xie[1], Zongsheng Yue[1], Qian Zhao[1], Deyu Meng[1,2*]
[1]Xi'an Jiaotong University; [2]Pazhou Lab, Guangzhou
xyrui@outlook.com, zsyzam@gmail.com
{caoxiangyong, xie.qi, timmy.zhaoqian, dymeng}@mail.xjtu.edu.cn

Abstract

An efficient approach for handling hyperspectral image (HSI) denoising issue is to impose weights on different HSI pixels to suppress negative influence brought by noisy elements. Such weighting scheme, however, largely depends on the prior understanding or subjective distribution assumption on HSI noises, making them easily biased to complicated real noises, and hardly generalizable to diverse practical scenarios. Against this issue, this paper proposes a new scheme aiming to capture general weighting principle in a data-driven manner. Specifically, such weighting principle is delivered by an explicit function, called hyper-weight-net (HWnet), mapping from an input noisy image to its properly imposed weights. A Bayesian framework as well as a variational inference algorithm for inferring HWnet parameters is elaborately designed, expecting to extract the latent weighting rule for general diverse and complicated noisy HSIs. Comprehensive experiments substantiate that the learned HWnet can be not only finely generalized to different noise types from those used in training, but also effectively transferred to other weighted models. Besides, as a sounder guidance, HWnet can help to more faithfully and robustly achieve deep hyperspectral prior(DHP). The extracted weights by HWnet are verified to be able to effectively capture complex noise knowledge underlying input HSI, revealing its working insight in experiments.

1. Introduction

Hyperspectral images (HSIs) record various adjacent electromagnetic spectrums of the same scene and provide more plentiful information than gray-scale or RGB images. Accordingly, over the past decades, HSIs have contributed to many practical applications such as food security [29], disease diagnosis [35], remote sensing [50] and so on.

The practically collected HSIs, however, are often corrupted with complex noises due to sensor erros, atmosphere, photon, *etc*., like stripe and deadline ones [3]. Especially,

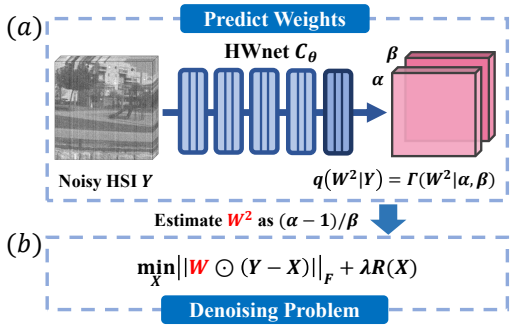


Figure 1: (a) Our method learns an explicit HWnet function, capable of directly obtaining a proper weighting scheme for an input noisy HSI image. (b) The learned HWnet can be used in plug & play for weight setting of a general weighted denoising model.

the real HSI noises are usually distributed non-i.i.d. across both spatial and spectral modes, *i.e*., different in different spatial locations [4], and varied in different bands [9]. This tends to largely hamper the performance of subsequent HSI processing tasks, and a pre-processing step for HSI denoising is thus necessary. But the complexity of real HSI noises makes it extremely hard to design a unique model for efficient and general HSI denoising.

An efficient utilized approach is to impose weights on different HSI pixels, expressed as:

$$\min_{X} ||W \odot (Y - X)||_F + \lambda R(X), \tag{1}$$

where $Y, X \in R^{hw \times b}$ represent the observed noisy and the recovered HSIs, respectively, h, w, b denote the spatial height, width, and spectral band number of the investigated HSI. Note that we have reshape the spatial width and height modes as one unique dimension. $||\cdot||_F$ denotes the Frobenius norm, and $\odot$ means the element-wise Hadamard production. $R(\cdot)$ denotes a general regularization term. The rationality of this scheme can be easily interpreted: larger weights should be assigned to less polluted HSI pixels/areas and vice verse, so as to suppress the negative influence brought by the noisy elements.

There are mainly two manners used for presetting the weighting scheme in (1). One is to pre-specify a certain weighting function [10] or directly pre-fix weights on all elements [41] based on prior understanding to the problem or

*Corresponding author

the data. A main clue is to assign weights inversely proportional to the estimated standard deviations on pixel noises before or during algorithm iterations [16]. Albeit easy to implement, such weighting scheme is generally hard to be properly pre-specified for real diversely complicated noises.

Another manner is to make all weights included in the model capable of being automatically and adaptively evaluated. The main idea is to make certain noise distribution assumptions, like mixture models [25][3][43], and then using EM or variational inference algorithms to optimize all involved variables (including the desired images and noise distribution parameters). During the learning process, the step of updating the recovered image naturally corresponds to a weighting scheme similar as (1), in which all weights can be automatically ameliorated in iterations based on the noise information delivered in the current step. The performance of this strategy, however, too much depends on the properness of the subjectively pre-assumed noise distribution. When the pre-assumption is largely deviated from real ones, the performance of the algorithm will be unavoidably degraded. It is thus critical to achieve a weighting regime which is able to adapt diverse and complex noise distributions for handling real-world HSI denoising issues.

Against this issue, this study raises a new weighting scheme on the model (1) to make it well self-adaptable to complex HSI noises. The basic understanding is that: since the proper weight scheme for W depends on the underlying noise extents embedded in the input noisy HSI Y, there rationally exists an implicit relationship mapping from Y to W. Accordingly, the proposed method aims to make this mapping explicit by designing it as a parametric function, called hyper-weight-net(HWnet), and learn all its parameters through a data-driven way. By learning such an explicit mapping function, which is expected to be capable of capturing the common and general weighting principle for HSIs, it can be readily used to insert weights into general weighted HSI recovery models (1) for any newly tested HSIs (as shown in Fig. 1), with no need of any noise distribution pre-assumptions.

Our contribution can be mainly summarized as follows:

1) A novel method is proposed to capture the general weighting principle for HSI denoising models in a data-driven manner. The extracted explicit weighting scheme is expected to adapt a wide range of complex HSI noises.

2) The learned weighting scheme is verified to have an excellent generalization capability, not only able to be finely used for test HSIs with noises evidently different from those contained in the training data, but also capable of being readily used in general HSI recovery weighted methods for direct weight assigning with no need of manual noise pre-assumptions.

3) It is verified that the weights extracted by our method can effectively reveal complex noise insights underlying the input noisy HSI. Remarkably, it can be easily used as an ameliorated loss for the known DHP network[32]. Instead of orienting to the input noisy images, the network can be better trained under a sounder guidance by using the rectified weighted loss deduced with our method.

Sec. 2 briefly introduces related work. Sec. 3 provides the proposed model as well as its solving strategy. Sec. 4 reports experimental results for performance evaluation of the proposed method. The paper is finally concluded. Throughout the paper, we denote scalar, vector, matrix as non-bold lower case, bold lower case, upper case letters, respectively.

2. Related Works

Existing HSI denoising methods have deeply explored HSI internal prior structures. As known in [2], HSI is with evident low-rank property along its spectral mode. By using this prior, [8][5] used PCA to extract clean image from noisy HSI. [46] adopted nuclear norm to describe this prior, and [40] extended this idea to weighted Schatten p-norm. [51] combined low rank models with sparse coding to jointly extract global and local features. [53] proposed a fast denoising algorithm based on low rank and sparse representation. [15] applied non-local similarigy in the denoising. [22] considered to plug an off-the-shell denoising network to a non-negative matrix factorization model. [17] combined spectral low rank constrain with spatial TV constraint to better preserve spatial structures. [19][36][27] further extended 2d TV regularization to spatial and spectral space. Seeing HSI as a 3D cube, [24][28][38][7] proposed tensor based recovery using tensor factorization methods.

Some other works focused on encoding complex noise contained in HSIs. [52] found that less bands of HSIs are corrupted by noise and proposed to select quality superior bands to help denoising targeted noisy bands. [5] combined PCA with noise adjust strategy. [16] took spectral noise distinction into consideration and adjusts output image in each iteration using estimated noise variance. [17] used sparse constraint on image residuals, which assume that noises follows i.i.d. Laplace distribution. [40] applied $l_{2,1}$ constraint on noise to make the denoising model more robust.

To make the model better adapt the real complex noises, [4] proposed to globally model noise using more powerful mix EP distributions. [9] further noticed the noise distinctions among bands, and proposed to model noise as a non-i.i.d. manner along spectrum. Later, [45] extend this idea to model HSI noise structure by a two-level hierarchical Dirichlet process, which better adapts such non-i.i.d. noise structures in real HSIs.

Recently, based on powerful learning ability of deep neural network, several deep learning (DL) methods have been proposed for the task. [6] designed a 2D CNN with dilation and residual learning. By concatenating adjacent bands, [44] proposed a jointly spatial-spectral network. Besides,

since HSIs are 3D data, several 3D networks are designed to better extract spatial-spectral features [11][23][37].

Albeit achieving good performance, DL approach is known to rely on the pre-collected training data. If there are evident bias between testing and training samples, the overfitting issue tends to easily occur. Comparatively, traditional model-driven methodology is with better interpretability and less relies on pre-collected training samples. But their performance largely rely on the proper pre-assumptions on data, especially on noises in our task. This work thus mainly aims to alleviate such subjective prior-assumption issue for traditional models, and make them better adapt real complex noises in real scenarios, so as to appeal that besides popular DL regimes, such conventional model-driven manner is also hopeful to be further explored.

3. The Proposed Method

Our aim is to learn an explicit function mapping from a noisy image Y to the weight matrix W, which can be readily imposed on the model (1) for adapting complex image noises. We denote the desired HWnet as $C_\theta(\cdot)$ with θ as its parameters. Given the training data $\{Y^m, X_{gt}^m\}_{m=1}^M$, where Y^m and X_{gt}^m represent a pair of noisy and expected clean images, respectively, and M denotes the training sample number, we expect to learn the form of $C_\theta(\cdot)$ in an end-to-end data-driven manner. To realize this aim, our main strategy is to construct a variational parametric approximation to the posterior of the latent variables, including the noise knowledge W and the latent clean image X, conditioned on the noisy image Y. Such an explicit parametric variational posterior can then easily guide to form $C_\theta(Y)$, and directly infer the weight matrix from any test noisy image. To this aim, we first require to formulate a rational full Bayesian model of the problem based on training image pairs.

3.1. Full Bayesian Model Based on Training Image Pairs

Denote A_{ij} as the element of a matrix A in its i-th row and j-th column. For an image pair $\{Y, X_{gt}\}$ in training set, Eq. (1) implies the following model to express the generation process of Y conditioned on latent variables W, X:

$$Y_{ij} \sim \mathcal{N}(Y_{ij}|X_{ij}, 1/W_{ij}^2), i \in [hw], j \in [b]. \quad (2)$$

Note that we directly take the latent variable W_{ij}^2 as precision of the Gaussian distribution for generating Y_{ij}. This can be naturally deduced by understanding the deterministic model (1) under the maximum a posteriori framework. This is also why W implicitly delivers the noise information underlying the input image Y [21]. Besides, different W_{ij}s across the entire image represent the non-i.i.d. noise property for the problem, make the model capable of sufficiently flexible to adapt complex noises in real noisy HSIs.

To achieve the full Bayesian model for the latent variables W and X conditioned on Y, we need to construct possibly faithful priors for W and X, respectively. Fortunately, the supervised noisy-clean image pairs are provided in the training data, making such task easy to be attained.

For the latent clean image variable X, it should be close to the given groundtruth image X_{gt}. We thus impose the following conjugate Gaussian distribution on it:

$$X \sim \mathcal{N}(X|X_{gt}, \varepsilon^2), \quad (3)$$

where ε^2 is set as a small value (1e-5 throughout all our experiments) to make X largely distributed close to X_{gt}[1].

For latent variable W_{ij}^2 representing the noise precision, we can easily set it as a conjugate prior distribution:

$$W_{ij}^2 \sim \Gamma(\rho + 1, \rho\sigma_{ij}^2), \quad (4)$$

which can guarantee W_{ij}^2 with mode $1/\sigma_{ij}^2$(σ_{ij}^2 is the noise variance in Y_{ij}). Such noise knowledge can be directly obtained for synthetic noisy images, or easily estimated by calculating the average of image noises (*i.e.*, $Y - X_{gt}$) around a local area of a pixel. We need to set a large ρ the make the distribution concentrated around the expected precision, and we just easily set it as 25, which is large enough to guarantee a good performance of our method consistently.

Combining (2), (4) and (3), a full Bayesian model for the problem can then be obtained:

$$p(X, Y, W^2) = p(Y|X, W^2)p(X)p(W^2). \quad (5)$$

Our aim is then to construct a variational parametric posterior to approximate $p(X, W^2|Y)$, through which to obtain the expected explicit weight function $C_\theta(\cdot)$.

3.2. Variational Parametric Posterior

We then need to construct a variational distribution $q(X, W^2|Y)$ to approximate the real posterior $p(X, W^2|Y)$ as following:

$$\begin{aligned} q(X, W^2|Y) &= q(W^2|Y)q(X|Y, W^2), \\ &= \prod_{ij} q(W_{ij}^2|Y) \prod_{ij} q(X_{ij}|Y, W^2). \end{aligned} \quad (6)$$

Based on the conjugate priors in (3) and (4), it is rational to formulate the variational posterior forms of W^2 and X as:

$$q(W_{ij}^2|Y) = \Gamma(W_{ij}^2|\alpha(Y;\theta)_{ij}, \beta(Y;\theta)_{ij}), \quad (7)$$

$$q(X_{ij}|Y, W^2) = \mathcal{N}(X_{ij}|(G(Y;W))_{ij}, \eta^2). \quad (8)$$

where $(\alpha(Y;\theta)_{ij}, \beta(Y;\theta)_{ij})$ represent the prediction functions for getting posterior parameters of latent variable W^2. The mapping $G(Y;W)$ denotes the algorithm for solving (1) to input a noisy image Y and output a recovery image under imposed weight W. Our aim is then to optimize the parameter θ to get an explicit scheme $C_\theta(Y) = (\alpha, \beta)$ for predicting weights directly from a test noisy image[2].

[1]The given X_{gt} is actually not the exact latent clean image X since any "clean" image definitely contain certain noises. Even when it is sufficiently clean, we can set ε extremely small to make X close to an impulse response on X_{gt}.

[2]When we obtain $C_\theta(Y)$, the weight for each Y_{ij} of a test image Y can be easily valued as the mode $W_{ij}^2 = (\alpha_{ij} - 1)/\beta_{ij}$ of the corresponding predicted Gamma distribution.

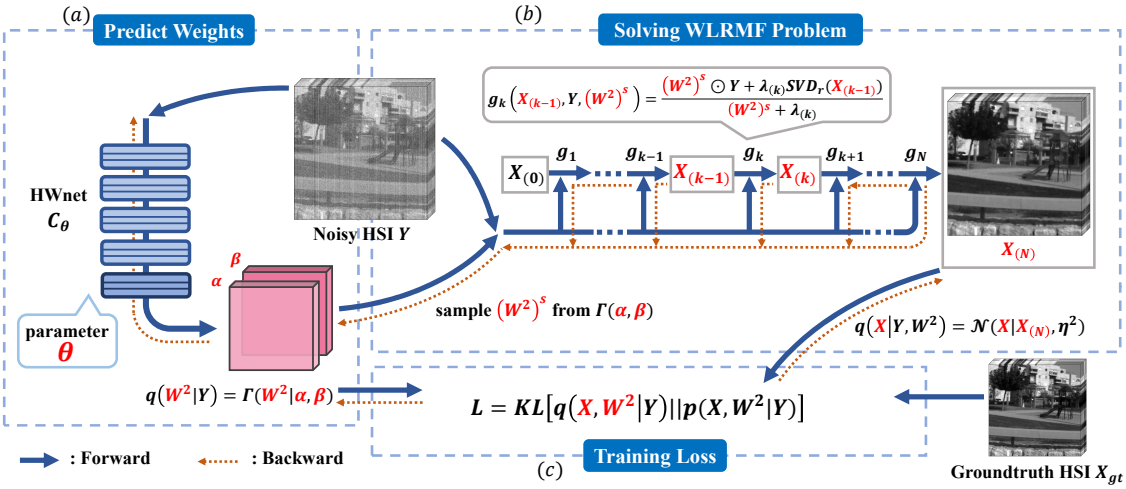


Figure 2: Training phase: We highlight all variables that are related to network parameter θ in red to show how θ flows to loss function in feedforward process. (a) C_θ takes noisy HSI Y as input and outputs parameters α, β of variational distribution of W^2, *i.e.* $q(W^2|Y) = \Gamma(W^2|\alpha,\beta)$. (b) Sample $(W^2)^s$ from $\Gamma(\alpha,\beta)$. Then algorithm G that solves WLRMF problem takes the noisy HSI Y and the sampling $(W^2)^s$ as input and outputs evaluated image $X_{(N)}$. The variational distribution of latent image X takes the form $\mathcal{N}(X|X_{(N)},\eta^2)$. (c)The loss function is to inference variational posterior distributions of W^2 and X.

Two remarks are necessary to be clarified. One is that the variance parameter η^2 can also be optimized. But since it is not very relevant to our concerned $C_\theta(\cdot)$, we just easily set it as a small constant value 1e-2. Besides, we want to emphasize that the parameter θ is shared by posteriors calculated on all training pairs, and thus our method is expected to extract the general statistical inference insight from the noisy image to its proper weight values.

Then we have three questions required to be answered. How to construct the form of $C_\theta(\cdot)$, how to express the algorithm as an explicit function G, and how to design the loss objective to calculate the parameter θ.

3.3. Design HWnet C_θ

To deal with HSIs with different spatial-spectral sizes and make use of abundant spectral information, we design the HWnet $C_\theta(\cdot)$ using pseudo-3d block(P3D) in [30] as basic convolution block. P3D separates a 3d conv-layer with kernel size $3\times3\times3$ to two 3d conv-layers connected by 'ReLU': '$3\times3\times1$ conv + ReLU + $1\times1\times3$ conv', which halves the needed parameters. Similar to a 5-layer DnCNN architecture [48], the overall C_θ has the structure of '(P3D+ReLU)+(P3D+BN+ReLU)*3+full 3d conv', and the feature channels in each layer is 64.

3.4. Taking WLRMF Algorithm as An Explicit Function

As widely known for general HSIs, a clean HSI X possesses evident low-rank structure along its spectral mode. Similar as many current works, we can formulate the following optimization model for (1):

$$\min_{X,U,V} ||W \odot (Y - X)||_F^2 + \lambda||X - UV^T||_F^2 \quad (9)$$

where $U \in R^{hw\times r}, V \in R^{b\times r}$, and $r \ll hw$, implying the low-rank penalty imposed on the recovery image X. This model exactly complies with the one known as WLRMF [33]. We employ it due to its simplicity for our weighting function construction task.

(9) can be easily solved using alternating minimization strategy. In each iteration, the updating step can be exactly written as the closed-form expression. Specifically, in the k-th iteration, the low-rank approximation UV^T of the input matrix can be updated by calculating

$$\begin{aligned}\{U_{(k)}V_{(k)}^T\} &= \arg\min_{UV^T} ||X_{(k-1)} - UV^T||_F^2 \\ &= \text{SVD}_r(X_{(k-1)}),\end{aligned} \quad (10)$$

where $\text{SVD}_r(\cdot)$ corresponds to the closed form solution by applying SVD to $X_{(k-1)}$ to get its rank-r approximation[3].

Then, X can be updated by computing:

$$\begin{aligned}X_{(k)} &= \arg\min_X ||W \odot (Y - X)||_F^2 + \lambda_{(k)}||X - U_{(k)}V_{(k)}^T||_F^2 \\ &= \frac{W^2 \odot Y + \lambda_{(k)}U_{(k)}V_{(k)}^T}{W^2 + \lambda_{(k)}}\end{aligned} \quad (11)$$

where the trade-off parameter $\lambda_{(k)}$ is set by $\lambda_{(k)} = \gamma\lambda_{(k-1)}$, where $\gamma > 1$, along iterations. Division in Equ. (11) is in element-wise meaning.

For each iteration of the algorithm, we can thus write the an explicit updating equation as:

$$\begin{aligned}X_{(k)} &= g_k(X_{(k-1)}, Y; W^2) \\ &= \frac{W^2 \odot Y + \lambda_{(k)}\text{SVD}_r(X_{(k-1)})}{W^2 + \lambda_{(k)}}.\end{aligned} \quad (12)$$

Furthermore, assume we set the iteration number as N, and take $X_{(0)}$ as initialization, we can express the algorithm exactly as the following function expression:

[3]The solution of U and V actually are not unique due to they have a freedom of scalar transformation. But the low-rank approximation UV^T is unique for the problem. We thus directly use it as the optimization variable.

$$X_{(N)} = g_N(...g_1(X_{(0)}, Y; W^2), Y; W^2) = G(Y; W^2), \quad (13)$$

which can be depicted as a "pseudo deep network", containing only shared W as its parameter in each of its layers, as can be seen in Fig. 2 (b).

3.5. Loss Designing for Parameter Training

Our goal is to then minimize the KL-divergence between the variational posterior $q(X, W^2|Y)$ and true posterior $p(X, W^2|Y)$. The objective is expressed as follows:

$$\min_{\theta} KL[q(X, W^2|Y)||p(X, W^2|Y)]. \quad (14)$$

Based on the variational inference principle [13], (14) is equivalent to the following optimization problem:

$$\begin{aligned} &\min_{\theta} -\mathbb{E}_{q(X,W^2|Y)}[\ln p(Y|X, W^2)] + KL[q(W^2|Y)||p(W^2)] \\ &+ \mathbb{E}_{q(W^2|Y)}\{KL[q(X|Y, W^2)||p(X)]\}. \end{aligned} \quad (15)$$

The objective function (15) contains three parts:

$$\begin{aligned} L_1 &= -\mathbb{E}_{q(X,W^2|Y)}[\ln p(Y|X, W^2)] \\ &\approx \sum_{m=1}^{M}\Big\{\frac{1}{S}\sum_{s=1}^{S}\frac{1}{2}||(W^m)^s \odot (Y^m - (X_{(N)}^m)^s)||_F^2 + \\ &\sum_{ij}\Big\{\frac{\eta^2\alpha_{ij}^m}{2\beta_{ij}^m} + \frac{1}{2}\ln 2\pi - \frac{1}{2}[\psi(\alpha_{ij}^m) - \ln\beta_{ij}^m]\Big\}\Big\}, \end{aligned} \quad (16)$$

$$\begin{aligned} L_2 &= KL[q(W^2|Y)||p(W^2)] \\ &= \sum_{m=1}^{M}\Big\{\sum_{ij}\Big\{(\alpha_{ij}^m - \rho - 1)\psi(\alpha_{ij}^m) - \ln\Gamma(\alpha_{ij}^m) \\ &+ \ln\Gamma(\rho+1) + (\rho+1)(\ln\beta_{ij}^m - \ln\rho(\sigma_{ij}^m)^2) \\ &+ \alpha_{ij}^m(\frac{\rho(\sigma_{ij}^m)^2}{\beta_{ij}^m} - 1)\Big\}\Big\}, \end{aligned} \quad (17)$$

$$\begin{aligned} L_3 &= \mathbb{E}_{q(W^2|Y)}\big[KL[q(X|Y, W^2)||p(X)]\big] \\ &\approx \sum_{m=1}^{M}\Big\{\frac{1}{S}\sum_{s=1}^{S}\Big\{\frac{1}{2\varepsilon^2}||(X_{(N)}^m)^s - X_{gt}^m||_F^2\Big\} \\ &+ \frac{\eta^2}{2\varepsilon^2} - \frac{1}{2}\ln\frac{\eta^2}{\varepsilon^2} - \frac{1}{2}\Big\}. \end{aligned} \quad (18)$$

Parts of integrations over W_{ij}^2 in L_1 and L_3 is intractable. We thus approximate them by randomly sampling S samples $\{[(W_{ij}^2)^m]^s\}_{s=1}^S$ from $q(W_{ij}^2|Y^m)$. $(X_{(N)}^m)^s$ means algorithm output image with input $[(W^2)^m]^s$ as in Equ. (13). It is convenient to use automatic differentiation tools in Pytorch[26] framework to calculate stochastic gradients. Another complex operator is SVD in (12). We refer to [18] to directly calculate derivatives wrt SVD.

3.6. Compared with Deep Unfolding Methods

Our proposed training strategy, especially the algorithm function G as defined in (13), seems similar to the deep unfolding approach by constructing deep network architecture through simulating the unrolling form of an algorithm [39][47][12]. However, our "unfolded" network G exactly accords with the iterative calculation process of the algorithm. Comparatively, conventional deep unfolding networks cannot exactly comply with the algorithm due to some inevitable approximations. In this sense, this new "deep unfolding" attempt is more interpretable and with more essential relationship with model-driven approaches.

4. Experimental Results

20 images were randomly selected from CAVE dataset [42] as training samples, and the left 12 ones are taken as test ones. The original image size is $512\times512\times31$. We crop 1250 overlapping patches of size $96\times96\times20$ from them for implementation convenience. Data augmentation schemes including rotation and flipping are used, resulting in total 10000 patches. In the WLRMF, the rank r is set as 3 due to its strong correlation among all spectral bands. The iteration number N is set as 20. In Equ. (12), the intermediate output $X_{(k)}$ stays invariant if W^2 and $\lambda_{(k)}$ scale with equal proportion, and we easily set initial trade-off parameter $\lambda_{(0)}$ as $0.5 * \text{mean}(W^2)$. We use ADAM optimizer [20] to train C_θ for 50 epoches. The initial learning rate is set as 1e-3 and decays by factor 0.8 every 4 epoches.

To create paired training patches, we firstly generate spatial-spectral variant sigma map Σ of size $96 \times 96 \times 20$, and then generate Gaussian noise to each patch as:

$$\begin{aligned} Y^m &= X_{gt}^m + n^m, \\ n^m &\sim \mathcal{N}(0, 1) \odot \Sigma^m. \end{aligned} \quad (19)$$

For testing, ICVL dataset[1], CAVE dataset and remote sensed image PaviaU[4] are employed. HYDICE Urban[5] image is used for real noisy HSI experiment. We finetune the pretrained $C_\theta(\cdot)$ on part of PaviaU for 8 epoches. The finetuned $C_\theta(\cdot)$ is used for testing the rest PaviaU image and HYDICE Urban image. Each band of HSI is normalized to [0,1] as in [46][9]. The iteration number N in WLRMF can be adjusted during testing phase and we set it as 150.

For synthetic experiments, we generate 7 different noise types. Albeit trained on noise type as (19), we expect the learned C_θ to be generalized to more diverse noise cases.

Case 1(i.i.d Gaussian): All bands of HSI are corrupted by Gaussian noise with zero mean and noise level $\sigma = 30$.

Case 2(non i.i.d Gaussian): The entire HSI is corrupted by zero-mean Gaussian noise with different intensities from band to band. The intensity ranges from 10 to 70.

Case 3(spatial-spectral variant Gaussian): HSI is corrupted by spatial-spectral variant Gaussian noise as in (19).

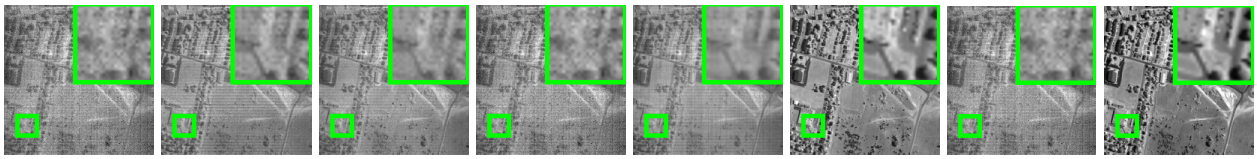
Case 4(Gaussian + Stripe): HSI is corrupted by non-i.i.d Gaussian noise in Case 2. 10 bands for ICVL and CAVE and 40 bands for PaviaU are randomly chosen to add stripe noise with 0.05-0.2 percentages.

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

[5] https://hdl.handle.net/11681/2925

(a) Original (b) Noisy (c) SVD (d) LRMR (e) LRTA (f) PARAFAC (g) NMoG (h) TDL (i) HW-LRMF

Figure 3: Visual comparison results at 15^{th} band of ICVL HSI. The noisy HSI is corrupted with Gaussian and stripe noise. The figures can be better observed by zooming in on the screen.



(a) Original (b) SVD (c) LRMR (d) LRTA (e) PARAFAC (f) NMoG (g) TDL (h) HW-LRMF

Figure 4: Visual denoising results at 104^{th} band on real noisy HYDICE Urban image.

Case 5(Gaussian + Impulse): Each band of HSI is contaminated by Gaussian noise in Case 2. 10 bands for ICVL and CAVE and 40 bands for PaviaU are added with impulse noise. The intensity ranges from 0.1 to 0.5.

Case 6(Gaussian + Deadline): All bands are corrupted by Gaussian noise in Case 2. 10 bands for ICVL and CAVE and 40 bands for PaviaU are randomly added with deadline. The number of deadline account for 0.05-0.2 percentages.

Case 7(Mixture Noise): Each band is randomly corrupted by at least one kind of noise in Cases 2-6.

The peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) are used for quantitatively evaluation.

4.1. Comparison Experiments

As mentioned above, once HWnet $C_\theta(\cdot)$ is obtained in the training stage, it can be readily used to WLRMF model (9) for new coming noisy HSIs by estimating W^2 as mode of $\Gamma(\alpha, \beta)$, where (α, β) are outputs of $C_\theta(\cdot)$. We denote the pre-trained $C_\theta(\cdot)$ together with WLRMF as HW-LRMF. In this section, we directly compare the HW-LRMF with SVD(baseline), LRMR[46], LRTA[31], PARAFAC[24], TDL[28] and NMoG[9]. All parameters are set as in their papers or in their provided codes.

From Tab. 1[6], one can see the superiority of the predicted weights W. Specifically, the proposed method outperforms SOTA low rank methods, LRMR and NMoG, under all noise settings. Although using very simple low rank matrix factorization framework, HW-LRMF surpasses SOTA tensor based methods LRTA, PARAFAC and TDL in almost all cases. Besides, although C_θ is only trained with noise Case 2, it still finely generalized to all rest different noise types. More results can be found in supplementary materials (SM).

Fig. 3 shows visual comparison of denoising effects under *Case 4*. LRTA and PARAFAC fail to remove stripes and vestige of stripes can be seen in TDL. SVD, LRMR and NMoG remove most stripes while still exist obvious random noise. Our HW-LRMF achieves best visual performance with less apparent stripes and random noise.

Fig. 4 shows results of real HSI noise removal of all competing methods. Except for NMoG and HW-LRMF, other methods can not efficiently remove their heavy noises. Comparatively, it is clear that our method preserves more details and achieves better visual results.

4.2. Transferring Weights to Other Models

One of our major interests is whether the pre-trained $C_\theta(\cdot)$ can be finely transferred on other weighted models (1) for weight imposing. To this aim, we select NAILRMA[16], LLRT[7] and NGmeet[15] for verification. NAILRMA employs HSI's low rank structure and thus shares some common property with LRMF model. It adjusts intermediate output using estimated noise variance in each band by a prefixed noise estimation function. NGmeet and LLRT are SOTA Gaussian denoising methods. Both global low rank property and local features on reduced images are considered in NGmeet. LLRT is a tensor based method, whose image prior is far more different from low rank property.

We reformulate their original models as weighted vision (1) and imposing weights predicted by HWnet, namely HW-NAILRMA, HW-NGmeet and HW-LLRT. Their original optimization methods are used to solve the weighted models. HW-NAILRMA and HW-NGmeet do not introduce extra hyper-parameters and all other parameters are set as the original ones. HW-LLRT needs one extra hyper-parameter and other parameters also inherit from original models. More details are introduced in SM.

In Tab. 2, it is seen that overall pretrained C_θ is beneficial for other denoising models to cope with complex noises. HW-NAILRMA surpasses original method in all noise settings. NGmeet achieves better restoration results for i.i.d Gaussian noise removal. But for complicated noise cases, the imposed weights W by HWnet greatly help boost performance. Although image prior used in LLRT is different from that in LRMF, W still helps improve the denoising results. Note that for NGmeet and LLRT, improvements brought by weighting scheme gradually decreases

[6]The results on CAVE and PaviaU are presented in SM due to page limitation.

Table 1: Quantitative comparison of HW-LRMF and other denoising methods on ICVL dataset. The best results are in **bold**.

Case	Index	Methods							
		Noisy	SVD	LRMR	LRTA	PARAFAC	NMoG	TDL	HW-LRMF
1	PSNR	18.59	29.21	30.24	34.95	30.93	30.99	**37.09**	34.93
	SSIM	0.5063	0.8886	0.9122	0.9353	0.8715	0.9371	**0.9697**	0.9431
2	PSNR	16.30	26.89	26.27	28.42	32.43	30.13	28.23	**32.52**
	SSIM	0.3273	0.7776	0.6975	0.7368	0.8748	0.8691	0.7822	**0.8946**
3	PSNR	17.53	27.84	24.92	19.57	31.49	30.71	22.11	**33.67**
	SSIM	0.4611	0.8173	0.7635	0.5052	0.8539	0.9078	0.6687	**0.9295**
4	PSNR	16.16	25.41	27.00	27.26	27.38	29.13	28.14	**31.80**
	SSIM	0.4471	0.8111	0.8439	0.7873	0.8068	0.9159	0.8607	**0.9258**
5	PSNR	14.58	21.98	27.30	24.51	27.64	28.91	23.74	**31.58**
	SSIM	0.2854	0.6204	0.7844	0.6248	0.7867	0.8475	0.6254	**0.8960**
6	PSNR	15.59	24.15	25.40	23.21	23.46	28.08	22.84	**31.26**
	SSIM	0.4596	0.8114	0.8305	0.6893	0.6808	0.9120	0.7379	**0.9324**
7	PSNR	14.08	21.28	23.97	20.46	22.05	25.38	20.11	**28.38**
	SSIM	0.3665	0.6973	0.7549	0.5288	0.6021	0.8461	0.5686	**0.8838**

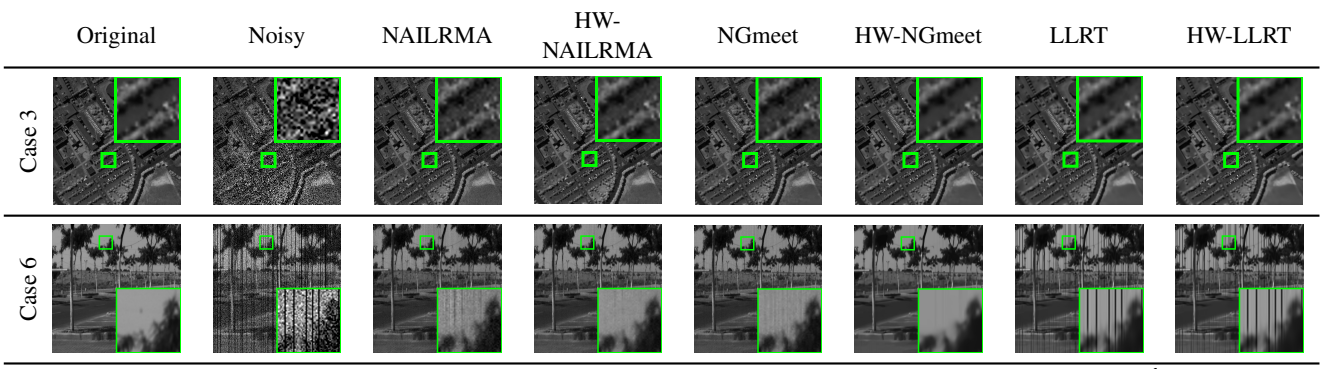


Figure 5: Denoising results of transfer experiments. The first row is for 'spatial-spectral variant Gaussian' noise at 71^{th} band of PaviaU image, and the second row is for 'Gaussian + deadline' noise at 13^{th} band of image in ICVL dataset.

when noise becomes more complex. This is in accordance with expectation since C_θ is trained with WLRMF model and is thus still certainly biased towards WLRMF model.

Fig. 5 shows the denoising results obtained by original models and its corresponding weighted vision under Case 3 and 6. It can be observed that the weighted models achieves better visual results for all competing methods. Results on real noise removal can be seen on SM.

4.3. Deep Hyperspectral Prior Experiments

In this part, we apply pre-trained C_θ to deep hyperspectral prior (DHP)[32]. DHP extends deep image prior [34] to HSI denoising, where the latter finds that the CNN C_μ itself contains descriptions for image prior. One can retrieve relatively clean image from only noisy image by feeding random input Z to a CNN and directly minimizing the MSE loss between the output and noisy image

$$\min_{\mu} ||C_\mu(Z) - Y||_F^2. \tag{20}$$

Attributed to the noise extraction capability of HWnet, we further extends DHP to its weighted vision HW-DHP:

$$\min_{\mu} ||W \odot (C_\mu(Z) - Y)||_F^2. \tag{21}$$

where W is estimated from $C_\theta(Y)$ as square root of mode of $q(W^2|Y)$ as aforementioned. The clean image is expected to be captured under a sounder guidance considering the complex noise embedded in Y.

We test the two training strategies under Case 2 and Case 5. Fig. 6 compares their denoising results. For 'non i.i.d Gaussian' noise, both DHP and HW-DHP firstly learn image prior and the PSNR value continues to rise. Then DHP quickly overfits to noise after around 5000 iteration, while HW-DHP slowly starts to overfit at around 10000 iteration. For 'Gaussian and impulse' noise, DHP can not efficiently extract image features from noisy observation due to the noise complexity. However, our HW-DHP attains much robust and better denoising results.

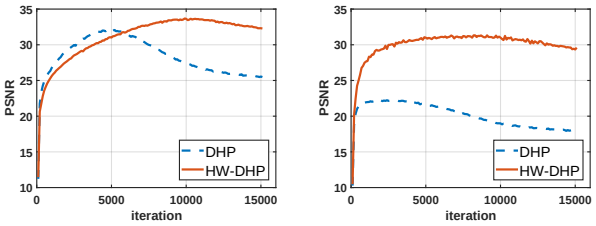
Fig. 6 seeds two useful messages: 1) The weight W trained with the help of handcrafted image prior could in turn assists unsupervised network learning, *i.e.*, explore image feature in neural networks. 2) Weighted loss (21) holds more resistance to overfitting to noise patterns and interference from violent noise.

4.4. Physical Meanings of Weights by HWnet

In this part, we take a deeper look at the shape of the extracted weight W by pre-trained C_θ. The noise is generated as (19). We visualize Σ on one band in Fig. 7b. As presented in (2), W^2 is interpreted as precision of Gaussian distribution. Since we inference the

Table 2: Quantitative comparison of transfer experiments on ICVL dataset. The best results are in **bold**.

Case	Index	NAILRMA	HW-NAILRMA	NGmeet	HW-NGmeet	LLRT	HW-LLRT	ablation
1	PSNR	32.47	**36.52**	**40.23**	39.98	**39.55**	39.48	39.55
	SSIM	0.9451	**0.9680**	**0.9813**	0.9798	**0.9771**	0.9769	0.9771
2	PSNR	31.11	**34.96**	34.93	**37.18**	38.66	**38.82**	38.66
	SSIM	0.8576	**0.9287**	0.9369	**0.9424**	0.9621	**0.9626**	0.9621
3	PSNR	30.64	**34.47**	34.68	**38.35**	38.42	**39.24**	38.42
	SSIM	0.8468	**0.9508**	0.9487	**0.9708**	0.9700	**0.9734**	0.9700
4	PSNR	29.79	**33.83**	32.79	**34.95**	35.79	**35.92**	35.79
	SSIM	0.9123	**0.9555**	0.9473	**0.9571**	0.9601	**0.9612**	0.9601
5	PSNR	26.25	**29.07**	30.10	**30.81**	30.82	**30.84**	30.82
	SSIM	0.7823	**0.8578**	0.8467	**0.9000**	0.8810	**0.8814**	0.8810
6	PSNR	27.89	**31.62**	26.52	**31.73**	29.34	**29.56**	29.34
	SSIM	0.9093	**0.9513**	0.8546	**0.9284**	0.8878	**0.8899**	0.8878
7	PSNR	24.71	**28.13**	26.43	**27.57**	26.62	**26.89**	26.62
	SSIM	0.8030	**0.8913**	0.8061	**0.8630**	0.8108	**0.8206**	0.8108

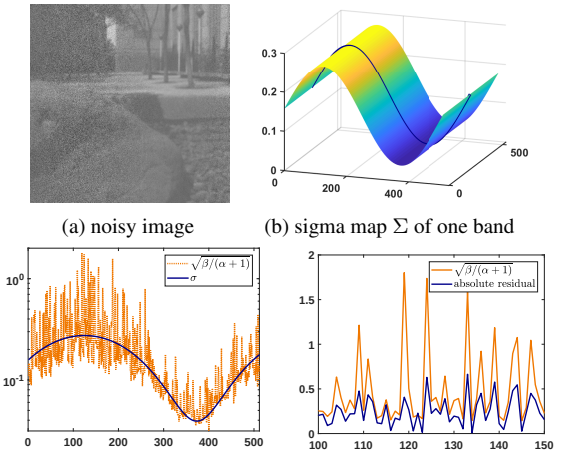


(a) Case2: non i.i.d Gaussian noise (b) Case5: Gaussian + impulse noise

Figure 6: Comparison results of DHP and HW-DHP

posterior distribution of W^2 as $\Gamma(\alpha, \beta)$, then $1/W^2 \sim \text{IG}(\alpha, \beta)$. The mode of $1/W^2$ is $\beta/(\alpha+1)$. Fig. 7c plots two lines from $\sqrt{\beta/(\alpha+1)}$ and Σ from the same position. The trend of $\sqrt{\beta/(\alpha+1)}$ is consistent with that of Σ overall, because in loss function (15), the term $KL[q(W^2|Y)||p(W^2)]$ controls the variational posterior distribution of W^2 not too far from prior distribution (4) whose mode is $1/\sigma^2$. $\mathbb{E}_{q(W^2|Y)}\{KL[q(X|Y,W^2)||p(X)]\}$ in (15) causes $\sqrt{\beta/(\alpha+1)}$ to vibrate around Σ. Specifically, we plot local part of $\sqrt{\beta/(\alpha+1)}$ and the absolute residual in Fig. 7d. The tendencies of this two lines are also consistent, yet on pixel level. This explains that loss term $\mathbb{E}_{q(W|Y)}\{KL[q(X|Y,W)||p(X)]\}$ intrinsically rectifies the proportion of residuals in (1) elementwisely.

To further verify the noise knowledge extraction capability of HWnet, like FFDnet[49] and CBDnet[14], we concatenate the estimated $1/W$ with noisy image as input to blind denoising networks, and take HSI-DeNet[6] and DSSnet[11] as baseline networks. The main modification to network structure is doubling the channels numbers in the first convolution layer, while for HSI-DeNet(2D CNN), a single layer of 3D convolution is additionally added before original first layers. We find that the modified networks with additional input $1/W$ evidently outperforms baseline networks. The improvement reveals that such predicted weights do provide insightful noise information and help network for better denoising. Details on experiment settings



(a) noisy image (b) sigma map Σ of one band

(c) Two lines from Σ and $\sqrt{\beta/(\alpha+1)}$ in the same position as highlighted in (b). (d) local part of $\sqrt{\beta/(\alpha+1)}$ and absolute residual between noisy image and clean one.

Figure 7: Physical meanings of weights extracted by C_θ

and results are included in SM.

5. Conclusion

In this paper, we have proposed a novel strategy to learn an explicit weighting function, called HWnet, for guiding to impose weights for any testing noisy HSIs. By taking the HWnet as a PnP weighting imposer, it has been verified that it can not only help the method finely adapt diverse and complex noises, but also can be readily extended to be used in general weighted models for improve their denoising capability. The weighted loss by HWnet can also be taken as a ameliorated loss beyond traditional MSE, to help get a sounder guidance for network training. Such new weighting scheme thus possesses a potential usefulness for performance enhancement for both traditional model-driven and more popular data-driven DL approaches.

Acknowledgement This research was supported by National Key R&D Program of China (2020YFA0713900) and the China NSFC projects (11690011, 61721002, U1811461, 61906151, 62076196).

References

[1] Boaz Arad and Ohad Ben-Shahar. Sparse recovery of hyperspectral signal from natural rgb images. In *European Conference on Computer Vision*, pages 19–34. Springer, 2016. 5

[2] José M Bioucas-Dias and José MP Nascimento. Hyperspectral subspace identification. *IEEE Transactions on Geoscience and Remote Sensing*, 46(8):2435–2445, 2008. 2

[3] Xiangyong Cao, Yang Chen, Qian Zhao, Deyu Meng, Yao Wang, Dong Wang, and Zongben Xu. Low-rank matrix factorization under general mixture noise distributions. In *Proceedings of the IEEE international conference on computer vision*, pages 1493–1501, 2015. 1, 2

[4] Xiangyong Cao, Qian Zhao, Deyu Meng, Yang Chen, and Zongben Xu. Robust low-rank matrix factorization under general mixture noise distributions. *IEEE Transactions on Image Processing*, 25(10):4677–4690, 2016. 1, 2

[5] Chien-I Chang and Qian Du. Interference and noise-adjusted principal components analysis. *IEEE transactions on geoscience and remote sensing*, 37(5):2387–2396, 1999. 2

[6] Yi Chang, Luxin Yan, Houzhang Fang, Sheng Zhong, and Wenshan Liao. Hsi-denet: Hyperspectral image restoration via convolutional neural network. *IEEE Transactions on Geoscience and Remote Sensing*, 57(2):667–682, 2018. 2, 8

[7] Yi Chang, Luxin Yan, and Sheng Zhong. Hyper-laplacian regularized unidirectional low-rank tensor recovery for multispectral image denoising. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4260–4268, 2017. 2, 6

[8] Guangyi Chen and Shen-En Qian. Denoising of hyperspectral imagery using principal component analysis and wavelet shrinkage. *IEEE Transactions on Geoscience and remote sensing*, 49(3):973–980, 2010. 2

[9] Yang Chen, Xiangyong Cao, Qian Zhao, Deyu Meng, and Zongben Xu. Denoising hyperspectral image with non-iid noise structure. *IEEE transactions on cybernetics*, 48(3):1054–1066, 2017. 1, 2, 5, 6

[10] Li Dong, Jiantao Zhou, and Guangtao Zhai. Efficient image sensor noise estimation via iterative re-weighted least squares. In *2017 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1326–1331. IEEE, 2017. 1

[11] Weisheng Dong, Huan Wang, Fangfang Wu, Guangming Shi, and Xin Li. Deep spatial–spectral representation learning for hyperspectral image denoising. *IEEE Transactions on Computational Imaging*, 5(4):635–648, 2019. 3, 8

[12] Weisheng Dong, Peiyao Wang, Wotao Yin, Guangming Shi, Fangfang Wu, and Xiaotong Lu. Denoising prior driven deep neural network for image restoration. *IEEE transactions on pattern analysis and machine intelligence*, 41(10):2305–2318, 2018. 5

[13] Charles W Fox and Stephen J Roberts. A tutorial on variational bayesian inference. *Artificial intelligence review*, 38(2):85–95, 2012. 5

[14] Shi Guo, Zifei Yan, Kai Zhang, Wangmeng Zuo, and Lei Zhang. Toward convolutional blind denoising of real photographs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1712–1722, 2019. 8

[15] Wei He, Quanming Yao, Chao Li, Naoto Yokoya, and Qibin Zhao. Non-local meets global: An integrated paradigm for hyperspectral denoising. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6861–6870. IEEE, 2019. 2, 6

[16] Wei He, Hongyan Zhang, Liangpei Zhang, and Huanfeng Shen. Hyperspectral image denoising via noise-adjusted iterative low-rank matrix approximation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6):3050–3061, 2015. 2, 6

[17] Wei He, Hongyan Zhang, Liangpei Zhang, and Huanfeng Shen. Total-variation-regularized low-rank matrix factorization for hyperspectral image restoration. *IEEE transactions on geoscience and remote sensing*, 54(1):178–188, 2015. 2

[18] Catalin Ionescu, Orestis Vantzos, and Cristian Sminchisescu. Training deep networks with structured layers by matrix backpropagation. *arXiv preprint arXiv:1509.07838*, 2015. 5

[19] Cheng Jiang, Hongyan Zhang, Liangpei Zhang, Huanfeng Shen, and Qiangqiang Yuan. Hyperspectral image denoising with a combined spatial and spectral weighted hyperspectral total variation model. *Canadian Journal of Remote Sensing*, 42(1):53–72, 2016. 2

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[21] William Leeb. Matrix denoising for weighted loss functions and heterogeneous signals. *arXiv preprint arXiv:1902.09474*, 2019. 3

[22] Baihong Lin, Xiaoming Tao, and Jianhua Lu. Hyperspectral image denoising via matrix factorization and deep prior regularization. *IEEE Transactions on Image Processing*, 29:565–578, 2019. 2

[23] Wei Liu and Joonwhoan Lee. A 3-d atrous convolution neural network for hyperspectral image denoising. *IEEE Transactions on Geoscience and Remote Sensing*, 57(8):5701–5715, 2019. 3

[24] Xuefeng Liu, Salah Bourennane, and Caroline Fossati. Denoising of hyperspectral images using the parafac model and statistical performance analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 50(10):3717–3724, 2012. 2, 6

[25] Deyu Meng and Fernando De La Torre. Robust matrix factorization with unknown noise. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1337–1344, 2013. 2

[26] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E.

Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 5

[27] Jiangjun Peng, Qi Xie, Qian Zhao, Yao Wang, Leung Yee, and Deyu Meng. Enhanced 3dtv regularization and its applications on hsi denoising and compressed sensing. *IEEE Transactions on Image Processing*, 29:7889–7903, 2020. 2

[28] Yi Peng, Deyu Meng, Zongben Xu, Chenqiang Gao, Yi Yang, and Biao Zhang. Decomposable nonlocal tensor dictionary learning for multispectral image denoising. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2949–2956, 2014. 2, 6

[29] Jianwei Qin, Kuanglin Chao, Moon S Kim, Renfu Lu, and Thomas F Burks. Hyperspectral and multispectral imaging for evaluating food safety and quality. *Journal of Food Engineering*, 118(2):157–171, 2013. 1

[30] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *proceedings of the IEEE International Conference on Computer Vision*, pages 5533–5541, 2017. 4

[31] Nadine Renard, Salah Bourennane, and Jacques Blanc-Talon. Denoising and dimensionality reduction using multilinear tools for hyperspectral images. *IEEE Geoscience and Remote Sensing Letters*, 5(2):138–142, 2008. 6

[32] Oleksii Sidorov and Jon Yngve Hardeberg. Deep hyperspectral prior: Single-image denoising, inpainting, super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 2, 7

[33] Nathan Srebro and Tommi Jaakkola. Weighted low-rank approximations. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 720–727, 2003. 4

[34] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9446–9454, 2018. 7

[35] Tuan Vo-Dinh. A hyperspectral imaging system for in vivo optical diagnostics. *IEEE Engineering in Medicine and Biology Magazine*, 23(5):40–49, 2004. 1

[36] Yao Wang, Jiangjun Peng, Qian Zhao, Yee Leung, Xi-Le Zhao, and Deyu Meng. Hyperspectral image restoration via total variation regularized low-rank tensor decomposition. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(4):1227–1243, 2017. 2

[37] Kaixuan Wei, Ying Fu, and Hua Huang. 3-d quasi-recurrent neural network for hyperspectral image denoising. *IEEE Transactions on Neural Networks and Learning Systems*, 2020. 3

[38] Qi Xie, Qian Zhao, Deyu Meng, and Zongben Xu. Kronecker-basis-representation based tensor sparsity and its applications to tensor recovery. *IEEE transactions on pattern analysis and machine intelligence*, 40(8):1888–1902, 2017. 2

[39] Qi Xie, Minghao Zhou, Qian Zhao, Zongben Xu, and Deyu Meng. Mhf-net: an interpretable deep network for multispectral and hyperspectral image fusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 5

[40] Yuan Xie, Yanyun Qu, Dacheng Tao, Weiwei Wu, Qiangqiang Yuan, and Wensheng Zhang. Hyperspectral image restoration via iteratively regularized weighted schatten p-norm minimization. *IEEE Transactions on Geoscience and Remote Sensing*, 54(8):4642–4659, 2016. 2

[41] Jun Xu, Lei Zhang, and David Zhang. A trilateral weighted sparse coding scheme for real-world image denoising. In *Proceedings of the European conference on computer vision (ECCV)*, pages 20–36, 2018. 1

[42] F. Yasuma, T. Mitsunaga, D. Iso, and S.K. Nayar. Generalized Assorted Pixel Camera: Post-Capture Control of Resolution, Dynamic Range and Spectrum. Technical report, Nov 2008. 5

[43] Hongwei Yong, Deyu Meng, Wangmeng Zuo, and Lei Zhang. Robust online matrix factorization for dynamic background subtraction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(7):1726–1740, 2017. 2

[44] Qiangqiang Yuan, Qiang Zhang, Jie Li, Huanfeng Shen, and Liangpei Zhang. Hyperspectral image denoising employing a spatial–spectral deep residual convolutional neural network. *IEEE Transactions on Geoscience and Remote Sensing*, 57(2):1205–1218, 2018. 2

[45] Zongsheng Yue, Deyu Meng, Yongqing Sun, and Qian Zhao. Hyperspectral image restoration under complex multi-band noises. *Remote Sensing*, 10(10):1631, 2018. 2

[46] Hongyan Zhang, Wei He, Liangpei Zhang, Huanfeng Shen, and Qiangqiang Yuan. Hyperspectral image restoration using low-rank matrix recovery. *IEEE transactions on geoscience and remote sensing*, 52(8):4729–4743, 2013. 2, 5, 6

[47] Kai Zhang, Luc Van Gool, and Radu Timofte. Deep unfolding network for image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3217–3226, 2020. 5

[48] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017. 4

[49] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing*, 27(9):4608–4622, 2018. 8

[50] Lefei Zhang, Liangpei Zhang, Dacheng Tao, and Xin Huang. On combining multiple features for hyperspectral remote sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 50(3):879–893, 2011. 1

[51] Yong-Qiang Zhao and Jingxiang Yang. Hyperspectral image denoising via sparse representation and low-rank constraint. *IEEE Transactions on Geoscience and Remote Sensing*, 53(1):296–308, 2014. 2

[52] Xiangtao Zheng, Yuan Yuan, and Xiaoqiang Lu. Hyperspectral image denoising by fusing the selected related bands. *IEEE Transactions on Geoscience and Remote Sensing*, 57(5):2596–2609, 2018. 2

[53] Lina Zhuang and José M Bioucas-Dias. Fast hyperspectral image denoising and inpainting based on low-rank and sparse representations. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(3):730–742, 2018. 2