HSI-DeNet: Hyperspectral Image Restoration via Convolutional Neural Network

Yi Chang, Student Member, IEEE, Luxin Yan, Member, IEEE, Houzhang Fang, Sheng Zhong, and Wenshan Liao

Abstract—The spectral and spatial information in hyperspectral images (HSIs) are the two sides of the same coin. How to jointly model them is the key issue for HSIs noise removal, including random noise, structural stripe noise, and dead pixels/lines. In this work, we introduce the deep convolutional neural network (CNN) to achieve this goal. The learned filters can well extract the spatial information within their local receptive field. Meanwhile, the spectral correlation can be depicted by the multiple channels of the learned 2D filters, namely the number of filters in each layer. The consequent advantages of our CNN-based HSIs denoising method (HSI-DeNet) over previous methods are three-folds. Firstly, the proposed HSI-DeNet can be regarded as a tensor-based method by directly learning the filters in each layer without damaging the spectral-spatial structures. Secondly, the HSI-DeNet can simultaneously accommodate various kinds of noise in HSIs. Moreover, our method is flexible for both single image and multiple images with slightly modifying the channels of the filters in first and last layers. Last but not least, our method is extremely fast in the testing phase, which makes it more practical for real application. The proposed HSI-DeNet is extensively evaluated on several HSIs, and outperforms state-of-the-art HSIs denoising methods in term of both the speed and performance.

Index Terms—Hyperspectral image restoration, denoising, destriping, convolutional neural network.

I. INTRODUCTION

HYPERSPECTRAL image restoration has been a hot topic in the past few years, which includes a variety of classical low-level vision tasks, such as HSIs denoising, destriping, super-resolution, and so on. In this work, we focus on various noise removal in HSIs. Numerous works have been proposed for advancing these fields and tremendous progress has been achieved in recent years [1]–[8]. However, there still exist several dark clouds over the HSI restoration filed, which are urgently needed to be solved.

Firstly, different from the natural 2D image, the HSIs could deliver additional spectral information of the scenes, resulting in three dimensional images. Due to the inherent 3D tensor format of HSIs, the previous vector/matrix-based methods can not fully exploit the spectral-spatial structural correlation. The way of modeling the spectral-spatial structural correlation in HSIs contributes much to the final restoration performance. Thus, we raise the first question: how to jointly utilize both the spatial and spectral information, and at the same time preserve the spectral-spatial structural correlation intactly for HSI 3D modeling?

Secondly, the noise characteristic in HSIs is complicated, which mainly includes both the random noise and structural stripe noise, due to the multi-detector imaging systems. Such a mixed noise situation makes the classical Gaussian assumption invalid. Although the elaborated mixture of Gaussian (MoG) methods have been proposed [9], [10], the restoration results are still somewhat unsatisfactory for the stripe noise. Hence, we ask the second question: should we choose the statistical modeling method directly to the end for HSI noise modeling?

Thirdly, the size of the HSIs is quite large due to the additionally spectral dimension. Most of previous methods suffer from very long running time, due to the large data size and the complicated operations on HSIs, such as non-local patch/cubic searching [11], [12]. This makes the state-of-the-art HSI restoration methods unpractical for real application. Consequently, we present the third question: could we just use simple and effective models for fast speed?

In recent years, for the three mentioned problems (data modeling, noise characteristic modeling, speed), several works have made great progress.

For HSIs data modeling, existing HSIs restoration methods can be roughly classified into three categories: one-dimensional vector based sparse representation methods, two-dimensional matrix based low-rank matrix recovery methods, and three-dimensional tensor based approximation methods. The previous two kinds of methods are easy to break the spectral-spatial structural correlation, due to the vectorization. The tensor-based methods are naturally proposed to preserve the intrinsic structure correlation with better restoration results, especially the low-rank tensor-based methods in recent two years [11]–[15]. However, the rank of the tensor in Tucker decomposition based method is usually defined as the sum of the rank along each of its modes, and still need to resort to the matrix. The problem has been alleviated, but still exists.

For noise characteristic modeling, there are two kinds of works: image denoising and image decomposition. The first kind of methods [9], [10] introduced the mixture of Gaussian model to accommodate the distribution of the mixed noise in HSIs. However, for complex mixed noise in HSIs, especially the stripe noise with similar structure as the image content, these methods can not well differ the noise from the HSIs. As for the latter method, the authors [16] introduced the RPCA [17] model by regarding the stripe noise as the structural...
error component to be estimated with Gaussian assumption. Although they are suitable for Gaussian and stripe noise, they are less effective for other kinds of mixed noise.

For the running time, the researchers in HSIs have paid less attention to this problem. Most of the optimization-based methods are time-consuming. The most representative method with fast processing time is BM4D [18], since it employs the 3D transform → filtering → inverse 3D transform pipeline with very simple operations. However, it is not data adaptive, and only works well for the random noise but less effective for stripe noise. In [11], we put forward the idea ‘making it shorter’ by utilizing the key low-rank property in HSI and discarding the weaker correlation so as to relieve the computation burden. Nevertheless, the running time is still intolerant.

In this work, we propose a convolutional neural network-based HSI restoration method to practically resolve these three problems.

**HSI Data Modeling.** Instead of optimizing a cost function with various unsupervised priors for the HSIs, we treat HSI restoration task as a discriminative mapping (learning the mapping parameters) problem and present a CNN-based HSIs restoration model. Naturally, the learned multi-channel filters in each layer can simultaneously model the spectral-spatial information explicitly. The spatially structured pattern in HSIs can be well represented by the local receptive filed of the learned filters via spatial convolution operation, and the spectral correlation in HSIs can be depicted by the learned multi-channel filters along the third axis, which in HSIs restoration we call it spectral filter. The spectral filter can be regarded as a spectral average operator. Moreover, the degraded HSIs can be directly imported to the CNN model without any further vectorization or matrixization.

**Noise Modeling.** As for the mixed noises, we do not apply any explicit expression to fit their distributions, since the noise in HSIs is too much complicated. Instead, our philosophy is to bypass the difficulty of constructing sophisticated distribution, and resort to a large amount of data. We make use of noisy HSIs to implicitly fit the distribution of the noises via the CNN model. Thus, the highly nonlinear fitting capability of the CNN enables our HSI-DeNet to handle arbitrary mixed noise for both the single image or multiple images easily.

**Running Time.** As for the training phase, we employ the residual learning strategy for fast convergence. As for the test phase, the CNN which mainly contains several convolution and activate layers, is quite suitable for parallel computation on GPU. Given a 512*512*10 noisy image, it only takes about twenty milliseconds to be processed in test phase.

As for the architecture of our HSI-DeNet, we mainly apply the residual learning strategy, dilated convolution, and multi-channel filtering. We do not directly export the desired clean image, but the residual noisy image. The residual learning has been demonstrated to be very effective in speeding up the training and boosting the final performance both in terms of low-level vision [19]–[21] and mid-level vision [22], [23]. The dilated convolution [24] is used to enlarge the receptive field of the filters in the spatial domain, meanwhile the multichannel filtering is designed for comprehensively capturing the spectral information. Note that, our concentration is not about the CNN design, but to demonstrate that the CNN is quite suitable for the HSIs restoration task.

We further extend the HSI-DeNet into an adversarial framework. We introduce the adversarial sub-network as a learnable prior. The $L_2$ loss based generator sub-network focuses on the pixel-level modeling benefiting for quantitative assessment (PSNR and SSIM) of the restored image; while the adversarial loss based discriminator sub-network aims at the feature-level modeling benefiting for the qualitative assessment (visual appearance) of the restored image. The two terms compete with each other, which is very similar to the fidelity + prior/regularization framework in optimization based methods. The contributions of the proposed work are summarized as follows:

- To our knowledge, this is the first work for HSIs restoration with fully convolutional neural network. The proposed HSI-DeNet can be regarded as the learning multiple channels 2D filters, and well preserve the spectral-spatial correlation.
- We incorporate the residual learning, dilated convolution, and multichannel filtering into the network for better modeling the HSIs. Moreover, we explore the adversarial network for HSIs restoration with better balance in qualitative and quantitative terms.
- The HSI-DeNet is robust and effective for mixed noise in HSIs, and very flexible for arbitrary input. Even for single image overwhelmed with mixed noise, we could also obtain satisfactory result.
- The proposed method has been tested on extensive HSI datasets with impressive results. Compared with previous methods, our method has achieved faster testing speed and better restoration performance.

The remainder of this paper is organized as follows. The related HSIs restoration methods are introduced in Section II. Section III presents the concrete architecture of HSI-DeNet and its adversarial extension. Extensive experimental results are reported in Section IV. Section V concludes this paper.

**II. RELATED WORKS**

The HSI community is indeed highly associated with the development of computer vision community. To date, a variety of HSIs denoising methods have been proposed in accordance with the most popular tools at that time. In this work, we classify them into three main categories, and compare them with the proposed method, respectively.

**Filter-based methods:** At the beginning of the 21-st century, the most powerful representation tool was the wavelet and its variations [25]–[27]. Unsurprisingly, followed by this direction, Othman et al. [28] proposed a hybrid spatial-spectral derivative domain wavelet shrinkage model with a fixed wavelet dictionary to reduce the noise in HSIs. A generalized multidimensional Wiener filter for denoising is adapted to HSIs [29]. The interested readers could refer to the related works [30]–[32]. However, the main drawback of these methods is that they used the hand-crafted and fixed wavelet basis for all HSIs. It has been verified that the learning
representation is more powerful than that of the pre-defined hand-craft representation \([33]\). Our idea to learn the filters is in line with these filtering-based methods. However the HSI-DeNet is more adaptive to the HSIs, which facilitates us to learn a more over-complete representation.

**Optimization-based methods:** Most of the HSIs restoration methods are optimization-based, including one-dimensional vector based sparse representation methods, two-dimensional matrix based low-rank matrix recovery methods, and three-dimensional tensor based approximation methods. The most representative methods for the one-dimensional method are the total-variational \([34]\) and dictionary learning \([35]\) method. In 2012, Yuan et al. \([36]\) proposed a HSIs denoising algorithm by employing a spectral-spatial adaptive total variation model. Zhao et al. \([37]\) introduced a HSIs denoising method by jointly utilizing sparsity and low-rank property of HSIs in spatial and spectral domains.

With the development of the robust principal component analysis (RPCA) \([17]\) and the fast optimization algorithm \([38]\), the two-dimensional low-rank matrix recovery methods have shown its effectiveness for HSIs restoration \([9]\), \([10]\), \([37]\), \([39]\)–\([42]\). By lexicographically ordering the 3-D cube into a 2-D matrix representation along the spectral dimension, the authors proposed a low-rank matrix restoration model for mixed noise removal in HSIs \([40]\), \([41]\). In \([42]\), we proposed a globally low-rank decomposition model for HSIs destriping, since only parts of data vectors are corrupted by the stripes but the others are not. However, these vector/matrix-based methods inevitably cause damage to the spectral-spatial structural correlation for the 3D tensor HSIs.

To alleviate this issue, the tensor-based HSIs denoising methods have emerged \([43]\)–\([45]\). In recent two years, when tensor decomposition meets the sparsity property, this direction has yielded state-of-the-arts HSIs restoration works \([11]\)–\([15]\). These tensor-based methods substantially improved the HSIs denoising performance, at the cost of higher computational burden. Our start point to better preserve the spectral-spatial correlation from the tensor perspective is the same as these methods. While our HSI-DeNet mainly relies on the fully convolutional operation, the previous tensors-based methods still need to resort to the matrix. Compared with the optimization-based methods, our HSI-DeNet shows better performance in running time. Moreover, the optimization-based methods are with the strong assumption to the Gaussian noise. Our work bypasses this assumption by using large training dataset to implicitly fit the distribution of arbitrary noisy input.

**Learning-based methods:** The deep learning has been widely used in HSIs mid-level tasks, such as classification \([46]\), pansharpening \([47]\), \([48]\), object detection \([49]\), to name a few. Also, the CNN has proven its effectiveness in natural image low-level vision tasks, such as denoising \([19]\), super-resolution \([50]\) and so on. Thus, it is natural for us to introduce the CNN to HSIs restoration task. Compared with classical network, our HSI-DeNet has two distinct characteristics. On one hand, we consider the spatial and spectral property of the HSIs, and apply the dilated convolution and multi-channel filters to model them, respectively. On the other hand, we output the residual noise, not the clean image. Such a residual learning strategy shows faster training convergence speed.
III. THE PROPOSED HSI-DeNet

A. Preliminary for CNN

Assuming there are $D$ layers in the designed network, for a given sample $Y \in \mathbb{R}^{R \times C \times B}$, the output of the first layer is $X^{(1)} = S(W^{(1)} \otimes Y + P^{(1)}) \in \mathbb{R}^{R \times C \times B_1}$, where $W^{(1)}$ is the projection matrix to be learned from the first layer, $P^{(1)}$ is the bias vector, $\otimes$ is the convolutional operator, $B_1$ is the channel number of the first layer, and $S : \mathbb{R} \rightarrow \mathbb{R}$ is the nonlinear activation function which handles each pixel individually, such as the sigmoid or rectified linear unit (RELU). Next, the output of the first layer $X^{(1)}$ is treated as the input of the second layer. Consequently, the output of the $d$-th layer can be expressed as:

$$X^{(d)} = S(W^{(d)} \otimes X^{(d-1)} + P^{(d)}) \in \mathbb{R}^{R \times C \times B_d}.$$  \hspace{1cm} (1)

The Eq. 1 known as the forward procedure is to extract the features from the input data in a hierarchy manner. The visual appearance of the features can be seen Fig. 1. The goal is to learn the mapping parameters by transforming the degraded data domain to the desired data domain. In conventional model-based methods, the restoration procedure can be formulated in the maximum a posteriori (MAP) framework from the statistical perspective as follow:

$$\hat{X} = \arg \max_X p(Y|X)p(X) \propto \arg \max_X \{\log p(Y|X) + \log p(X)\}.$$ \hspace{1cm} (2)

Each component in Eq. 2 such as the noise (posterior term) and the clean data (prior term), are modeled with one specific and explicit distribution. The intuition to transform the degraded data to another domain behind the conventional methods and CNN is similar. For model-based methods, they make an assumption for the data distribution in the transformed domain, such as the gradient domain based TV prior [34] and the wavelet domain based Framelet prior [51], [52]. Usually we can apply the fast optimization method, such as ADMM [38] by introducing auxiliary variable $A$ to solve Eq. 2:

$$A^{(k+1)} = \text{shrink}_\alpha (DX^{(k)} + \alpha J^{(k)}),$$ \hspace{1cm} (3)

where $D$ is the sparse transformation operator, $A$ is the auxiliary variable which can be approximately equivalent to $X$, $J$ can be regarded as the compensating variation, $\alpha$ is the regularization parameter, $\text{shrink}_\alpha$ is the soft shrinkage operator, and $k$ is the iteration number.

We can observe that the Eq.1 and 3 are very similar to each other. Both of them obtain the desired solution gradually via a linear transformation and then non-linear activation function. The number of the recursion depends on the depth of deep model and the iteration of optimization method, respectively. This intrinsic similarity can partially explain why the deep model is also suitable for image restoration task. However, the transformation parameters in CNN model are adaptively learned to implicitly fit the distribution of the training dataset, which makes them more professional for specific task.

B. Problem Formulation

The noise degradation model in this work is formulated as

$$Y = X + N,$$ \hspace{1cm} (4)

where $Y \in \mathbb{R}^{R \times C \times B}$ is the measured HSI, $R$, $C$ and $B$ stand for the numbers of the row, column and band respectively, $X$ is the desired clear HSI, and $N$ is the noise in HSI, which includes various noise components. Note that, the goal of our work is to estimate residual noise component $N$, not the clear image, from the degraded image $Y$. The main reason is that we adopt the residual learning idea from [19], [22] to train a residual mapping $F(Y) = N$. The restoration problem is formulated as a regression task as follows:

$$J_{Recon} = \frac{1}{2} \| F(Y) - N \|^2,$$ \hspace{1cm} (5)

where $F$ is the composite network mapping function of $S$.

C. Architecture of HSI-DeNet

In the proposed HSI-DeNet, we use a very deep convolutional network followed by the [19], [50], [53]. Each convolutional layer consists of $M_d$ filters with the size of $3 \times 3 \times N$, except the first and last output layer. The channel of the first and last output layer has to match the spectral dimension of the input HSI. We use $3 \times 3$ filters throughout the network with stride 1, which has been demonstrated that the decomposition of larger size filters into small size filters with deeper layers would make the model more discriminative [19], [20], [22], [50], [53]. To avoid the boundary effect and preserve the spatial size, we pad each layer with the same size as the original image.

The architecture of HSI-DeNet is shown in Table. 1. Each block contains three components: convolutional, batch normalization (BN), and rectified linear unit (RELU), as shown in Fig. 2. We denote the Convolutional(C) + Batch normalization(BN) + RELU(RELU) block as CBR. The depth $D$ of HSI-DeNet is 19 (including the $L_2$ loss layer). The main reasons for us to choose the depth as 19 are three-fold. On one hand, the depth in the CNN model is similar to the iteration number in optimization based methods. Many works [20], [54] have discussed their relationship and design their deep architecture based on the optimization solvers. Since the iteration number of the non-convex problem is usually determined empirically, we also observe from the experimental results that the depth 19 is very robust for our HSI restoration task. On the other hand, we follow the widely used VGG-nets to set the depth of our HSI-DeNet as 19. This depth can achieve well balance between training difficulty and representation ability. Last but not least, our training image size is 40 * 40 which is slightly smaller than the receptive field of our model. This means our model could utilize the whole contextual information of the given HSIs.

The BN layer is incorporated for avoiding the gradient vanishing or divergence issue. And the RELU layer is utilized for pursuing sparsity and also for its highly nonlinear ability. Note that, we do not contain any pooling layer in our network. The main reason is that the image processing task is a
regression task while the high level vision tasks are always classification based tasks. In image processing, we need to estimate the pixel level information. However, the pooling layer would inevitably cause information loss. Therefore, in HSI-DeNet, we do not apply any pooling layer.

1) Residual Learning: Previous deep learning based image processing methods directly mapped the degraded image into the clean one [55]–[57]. However, the gradient vanishing issue restricts these methods to train a very deep model with powerful representation. In this work, we added a skip connection between the input and output, which means that the network actually learns the difference between the input and output. This residual learning scheme proposed by He et al. [22] figured out creative way to learn the sparse residual image, not the image itself, since the sparser gradient of the residual image was easier to be propagated. The loss function of our network is $1/2 \times ||N - f(Y)||$, where $f$ is the network mapping function. Several pioneer works have demonstrated its effectiveness in various fields [19]–[21], [23], [50]. Therefore, it is natural for us to apply the residual learning in HSIs restoration.

2) Dilated Convolution and Multichannel Filtering: It is known that the more contextual information CNN models, the better restoration results they obtain [50]. Modern networks integrate multi-scale contextual information, namely enlarge the receptive field of the network via designing deeper layers [19], [53]. However, this may increase the difficulty of the training procedure due to the gradient vanishing issue. In this work, we introduce the dilated convolution [24] into the middle layers (Layer CBR7, 8, 9 with dilation 2), which aggregates multi-scale contextual information without losing resolution or increasing the depth of the network. We can observe from Table I the receptive field of our model is larger than the size of the image, which means we could completely make use of both the local and non-local contextual information.

The classical networks, such as Alexnet [33], VGGnet [53], and U-net [58] all employed the multichannel filtering strategy, in which each layer contains multiple feature maps. These models increased the number of the filters at first few layers, and then reduced the number of the filters gradually. Such a flexible manner greatly increases the representation ability of the network. For HSIs with multiple spectra (corresponding to multichannel in the network), the multichannel filtering in each layer becomes much more important, which undoubtedly increases the representation ability for the spectral information. As shown in Table I we gradually increase the channels from 10, 64, 128 to 256, and then decrease it symmetrically from 256, 128, 64, to 10.

3) Training Details: We initialize the convolutional filters with Xavier method [59]. The learning rate is initially set as 0.001 and decreased to a small value 0.00001. The momentum and decay are fixed as 0.9 and 0, respectively. The ADAM solver [60] is introduced to optimized the model. We trained the model with 300 epoches with the batchsize as 128. We obtained the training data from the ICVL [61] which includes 201 scenes. We cropped 500 sub-images from them as the training dataset and 50 sub-images as the test dataset. The training data was normalized to [0, 1]. Since the earlier image bands (mainly from the 400nm to 450nm) in ICVL contain random noises, we just extracted the band 550nm to 640nm with interval 10 (namely 10 bands) as the input. Note that, the training samples in our model are 500 (180 × 180) images. However, since the CNN does not require fixed inputs, we extracted 16 (40 × 40) sub-samples, via the sliding window with stride 40 from each sample. Then, we augmented each sub-sample 8 times with flip and rotation. Therefore, the total training samples (40 × 40 × 10) in our experiment are 500 × 16 × 8 = 64000. If we further consider each band as an image (Compared with single image based CNN models), the final training samples would be regarded as 64000 × 10 = 640000. That is the main reason why our training model works well with only 500 samples. In our experiment, we did not observe obvious difference for the training data ranging from 300 to 700. The MatConvnet toolbox [62] is employed to train the HSI-DeNet.

4) Relationship with Previous Methods: Comparing with the filtering-based methods, such as the wavelet, the learned filters in HSI-DeNet could be regarded as its data adaptive version. The previous hand-craft wavelets could only capture the specific image structures, such as horizontal, vertical and diagonal information. While hundreds of various filters in HSI-DeNet are more representative for HSI structure, and

<table>
<thead>
<tr>
<th>Layer</th>
<th>CR</th>
<th>CBR1</th>
<th>CBR2</th>
<th>CBR3</th>
<th>CBR4</th>
<th>CBR5</th>
<th>CBR6</th>
<th>CBR7</th>
<th>CBR8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kernel Size</td>
<td>3×3</td>
<td>3×3</td>
<td>3×3</td>
<td>3×3</td>
<td>3×3</td>
<td>3×3</td>
<td>3×3</td>
<td>3×3</td>
<td>3×3</td>
</tr>
<tr>
<td>Filter Number</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>128</td>
<td>128</td>
<td>128</td>
<td>256</td>
<td>256</td>
</tr>
<tr>
<td>Dilation</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Receptive Field</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>9</td>
<td>11</td>
<td>13</td>
<td>15</td>
<td>19</td>
<td>23</td>
</tr>
<tr>
<td>Image Size</td>
<td>40×40×64</td>
<td>40×40×64</td>
<td>40×40×64</td>
<td>40×40×64</td>
<td>40×40×128</td>
<td>40×40×128</td>
<td>40×40×128</td>
<td>40×40×256</td>
<td>40×40×256</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Layer</th>
<th>CBR9</th>
<th>CBR10</th>
<th>CBR11</th>
<th>CBR12</th>
<th>CBR13</th>
<th>CBR14</th>
<th>CBR15</th>
<th>CBR16</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kernel Size</td>
<td>3×3</td>
<td>3×3</td>
<td>3×3</td>
<td>3×3</td>
<td>3×3</td>
<td>3×3</td>
<td>3×3</td>
<td>3×3</td>
<td>3×3</td>
</tr>
<tr>
<td>Filter Number</td>
<td>256</td>
<td>128</td>
<td>128</td>
<td>128</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>10</td>
</tr>
<tr>
<td>Dilation</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Receptive Field</td>
<td>27</td>
<td>29</td>
<td>31</td>
<td>33</td>
<td>35</td>
<td>37</td>
<td>39</td>
<td>41</td>
<td>43</td>
</tr>
<tr>
<td>Image Size</td>
<td>40×40×256</td>
<td>40×40×128</td>
<td>40×40×128</td>
<td>40×40×128</td>
<td>40×40×64</td>
<td>40×40×64</td>
<td>40×40×64</td>
<td>40×40×64</td>
<td>40×40×10</td>
</tr>
</tbody>
</table>
Figure 3. Comparison between the features extracted by HSI-DeNet and TV methods. It is evident that in each iteration/layer, the sparsity-based shrinkage methods (here we just give an example with TV) utilize the same features, while the HSI-DeNet could extract the hierarchical features.

Figure 4. The flowchart of the proposed HSI-DeGAN.

Table II

<table>
<thead>
<tr>
<th>Layer</th>
<th>CR</th>
<th>CBR1</th>
<th>CBR2</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kernel Size</td>
<td>4*4</td>
<td>4*4</td>
<td>4*4</td>
<td>5*5</td>
</tr>
<tr>
<td>Filter Number</td>
<td>64</td>
<td>128</td>
<td>256</td>
<td>1</td>
</tr>
<tr>
<td>Stride</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Image Size</td>
<td>20<em>20</em>64</td>
<td>10<em>10</em>128</td>
<td>5<em>5</em>256</td>
<td>1<em>1</em>1</td>
</tr>
</tbody>
</table>

the additional non-linear response function further increase its generalization ability.

Comparing with the sparsity-based optimization methods, such as dictionary learning [35], total variational [63], and low-rank [40], their relationships with the CNN have been discussed in [20], [54]. For example, the iteration of the shrinkage can be unfolded and regarded as the non-linear response in the network. As shown in Fig. 3 in each phase (iteration in optimization methods and layers in CNN, respectively), the optimization based methods employ the same transformation for the image, while the CNN extracts the different scale and directional information gradually. Thus, we can roughly regard the conventional methods as the shallow model while the CNN is the deep model with stronger representation.

D. Extension to Adversarial Network

The generator networks optimized based on the pixel-wise loss function, such as mean squared error or $L_1$, tend to produce the over-smooth results [64], [65], since they mainly focus on the pixel-wise level image differences. To exploit priors from the image level, we introduce the adversarial discriminators for the generator. The generative network (described in Section III-C) can be further incorporated in an adversarial framework, by additionally incorporating with a discriminator. Adversarial network [66] is a recent approach that has shown remarkable performances to generate synthetic photo-realistic images. Generative Adversarial Networks (GAN) are composed of two models that are alternatively trained to compete with each other. The generator is trained to produce the true data distribution $p_{data}$ so that the generated images are difficult for the discriminator to differentiate from real images. Meanwhile, the discriminator is served as a classifier to distinguish fake images generated by $G$ from real images. The objective function of the adversarial networks is expressed as follow:

$$J_{adv} = E_{X \sim p_{data}}[\log D(X)] + E_{Y \sim p_{Y}}[\log(1 - D(G(Y)))]$$

(6)

where $p_{Y}$ and $p_{X}$ denote the distributions of noise input $Y$ and real data $X$, $G$ denotes the generator and $D$ represents the discriminator.

We show the general flowchart of the proposed method in Fig. 4. The proposed HSI-DeGAN mainly contains two parts: the generator and discriminator sub-network. The red arrow denotes the forward procedure, and the black arrow indicates the backward propagation. The left green rectangular box is the $L_2$ reconstruction based generator (Section III-C), which can be regarded as the fidelity term in the optimization method, since both of them are used to constrain restored result should be consistent with the observation. The right orange rectangular box is the adversarial perceptual loss based discriminator, which can be equivalent to the prior/regularization term for further solution refine. Here, the discriminator makes the distribution between the restored image and ground true.
indistinguishable, which can be regarded as the adaptively learned prior for the data.

The input to the discriminator is a pair of images: a ground truth image and a restored image by the generator. The output of the discriminator is a binary value for the ground truth image to be one and the restored image to be zero. The detailed architecture of the discriminator is shown in Table II. Here, the dilation of each layer is set as 1. We gradually decrease the image size by introducing the stride. The whole loss function of our network is defined as:

\[ J_f = \lambda_1 J_{Recon} + \lambda_2 J_{Adver}, \]

where \( \lambda_1 \) and \( \lambda_2 \) are the weights to balance the effects of different losses. The generator and discriminator can be trained adversarially/alternatively. The discriminator aims to maximize the above objective while the generator tries to minimize the objective. Note that, in the testing phase, only the generator network is required to obtain the clear image which is as natural as the real image.

IV. EXPERIMENTAL RESULTS

A. Experimental Setting

We use the ICVL [61] as our training dataset for all tasks. The test datasets include the CAVE and ICVL. The ICVL database images were acquired using a Specim PS Kappa DX4 hyperspectral camera and a rotary stage for spatial scanning. At this time it contains 201 images and will continue to grow progressively. Images were collected at 1392 × 1300 spatial resolution over 519 spectral bands (400-1000nm at roughly 1.25nm increments). The CAVE database images were acquired using a Cooled CCD camera (Apogee Alta U260). The database consists of 32 scenes at 512 × 512 spatial resolution from 400nm to 700nm at 10nm steps (31 bands total).

For the HSI denoising methods, we compare with block-matching and 3D filtering (BM3D) [67], parallel factor analysis (PARAFAC) [44], low-rank matrix recovery (LRMR) [40], block-matching and 4D filtering (BM4D) [68], tensor dictionary learning (TDL) [14], intrinsic tensor sparsity regularization (ITSReg) [15], Laplacian regularized low-rank tensor recovery (LLRT) [11]. We use the codes provided by the authors downloaded from their homepages, and fine tune the parameters by default or following the rules in their papers to achieve the best performance. And once our manuscript has been accepted, the training and testing code of our methods can be downloaded from the homepage of the author [1].

The peak signal-to-noise ratio (PSNR), structure similarity (SSIM [69]), erreur relative globale adimensionnelle de synthese (ERGAS [70]) and spectral angle map (SAM [71]) are employed for the quantitative assessment. The PSNR and SSIM evaluate the spatial quality, and the ERGAS and SAM assess the spectral quality. The larger PSNR and SSIM values are, and the smaller ERGAS and SAM values are, the better the restored images are.

B. Simulated Noise Removal

We test four noisy cases: the random noise, the stripe noise, the mixed random and stripe noise, and the single image.

1) Random Noise Removal: Here, we give two examples degraded with additive Gaussian random noise with zero mean and different variances. The BM3D and BM4D [Fig. 5 and 6 (c), (f)] tended to introduce the unexpected ringing artifacts. The LRMR [Fig. 5 and 6 (e)] suffered from the residual noise,
since it has strong assumption on the low-rank constraints of the multispectral inputs. The low-rank tensor-based TDL, ITSReg and LLRT worked well in light noise case [Fig. 5(g)-(i)], while for heavy noise case [Fig. 6(g)-(i)] they either left residual noise or over-smoothed the HSI. In both Fig. 5 and Fig. 6, the HSI-DeNet could remove the noise satisfactorily and preserve clear details, such as the ellipse region in Fig. 5 and rectangle regions in Fig. 6.

Moreover, with the increase of the noise level, the HSI-DeNet obtained much more advantageous over other methods in terms of the quantitative assessments. The main reason is that, when the noise is overwhelmed in HSIs, the local, nonlocal self-similarity or spectral correlation have been severely damaged. Thus the performance of previous methods inevitably decreased rapidly. On the contrary, the learning based HSI-DeNet could infer the missing information from the external dataset, also benefitting from its intrinsic tensor-based structural preserving ability. The HSI-DeNet consistently obtained the best results for both the spectral and spatial assessments. This demonstrated that our HSI-DeNet could better preserve spectral integrity due to the tensor level operation, and also better preserve spatial structural details due...
to the learned information from the external dataset.

2) Stripe Noise Removal: As far as we know, most of the afore-mentioned HSI denoising can not handle the stripe noise. We compared the HSI-DeNet with the state-of-the-art destriping methods: total variational (TV) [34], unidirectional variational model (UTV) [63], wavelet-Fourier adaptive filtering (WFAF) [72], statistical linear destriping (SLD) [73], Low-rank multiple image decomposition (LRMID) [42], anisotropic spectral-spatial total variational (ASSTV) [3]. From Fig. 7(c), (d), (e), (g), we can see obvious unexpected artifacts. For the unidirectional model [Fig. 7(f) and (h)], the original linear pattern with the same direction as the stripe has also been removed. The result of our HSI-DeNet is with satisfactory visual appearance. Moreover, the image structure has been preserved intactly. This result strongly demonstrates that the convolutional neural network has more powerful representation, which could better distinguish the stripe from the image structure.

3) Mixed Noise Removal: The random noise and the stripe noise always co-exist in the real HSI. In this section, we test the performance of our HSI-DeNet under the mixed noise case. The results are shown in Fig. 8. We observe an interesting phenomenon that the state-of-the-art HSI restoration method could only handle the random noise and alleviate the stripe noise. However, they can not totally remove the stripe noise, as shown in Fig. 8(c)-(i). On the contrary, in Fig. 8(j), the HSI-DeNet could simultaneously remove both the random and stripe noise with better visual appearance. This demonstrates that the explicitly modeling of the mixed noise is inappropriate. The implicit modeling via the CNN offers a new perspective for this complex problem. Moreover, their weak effect to the stripe noise heavily depends on the spectral correlation in HSIs. For the mixed noise in the single image, previous methods even do not consider this situation. We show in the next section that the HSI-DeNet could squeeze more spatial information and perfectly handle it.

4) Single Image Mixed Noise Removal: To the best of our knowledge, few works have considered the single image mixed stripe and random noise removal issue, except the work [74]. Such a mixed noise removal problem is extremely difficult. For
**Table III**

**Spatial and spectral quantitative assessments of different methods under different noise levels. The R20_S20 means the simultaneous random noise with variance 20 and stripe noise with intensity 20.**

<table>
<thead>
<tr>
<th>Sigma</th>
<th>Index</th>
<th>Methods</th>
<th>Noisy</th>
<th>BM3D</th>
<th>PARAFAC</th>
<th>LRMR</th>
<th>BM4D</th>
<th>TDL</th>
<th>ITSReg</th>
<th>LLRT</th>
<th>HSI-DeNet</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PSNR</td>
<td></td>
<td>22.07</td>
<td>35.06</td>
<td>29.54</td>
<td>30.04</td>
<td>38.54</td>
<td>37.88</td>
<td>39.56</td>
<td>40.15</td>
<td><strong>40.26</strong></td>
</tr>
<tr>
<td>20</td>
<td>SSIM</td>
<td>0.3519</td>
<td>0.9413</td>
<td>0.8658</td>
<td>0.7234</td>
<td>0.9682</td>
<td>0.9637</td>
<td>0.9674</td>
<td>0.9776</td>
<td>0.9776</td>
<td><strong>0.0088</strong></td>
</tr>
<tr>
<td></td>
<td>SAM</td>
<td>0.1726</td>
<td>0.0258</td>
<td>0.0156</td>
<td>0.0390</td>
<td>0.0157</td>
<td>0.0100</td>
<td>0.0112</td>
<td>0.0093</td>
<td>0.0093</td>
<td><strong>0.0088</strong></td>
</tr>
<tr>
<td></td>
<td>ERGAS</td>
<td>176.25</td>
<td>39.48</td>
<td>74.52</td>
<td>72.19</td>
<td>26.52</td>
<td>28.68</td>
<td>23.57</td>
<td>21.84</td>
<td>22.02</td>
<td><strong>22.02</strong></td>
</tr>
<tr>
<td>50</td>
<td>PSNR</td>
<td>14.12</td>
<td>30.80</td>
<td>28.07</td>
<td>23.52</td>
<td>33.79</td>
<td>33.34</td>
<td>32.30</td>
<td>35.75</td>
<td>36.17</td>
<td><strong>36.17</strong></td>
</tr>
<tr>
<td></td>
<td>SSIM</td>
<td>0.0777</td>
<td>0.8507</td>
<td>0.7711</td>
<td>0.3706</td>
<td>0.9040</td>
<td>0.8877</td>
<td>0.8062</td>
<td>0.9367</td>
<td>0.9379</td>
<td><strong>0.9379</strong></td>
</tr>
<tr>
<td></td>
<td>SAM</td>
<td>0.5604</td>
<td>0.0681</td>
<td>0.0200</td>
<td>0.0620</td>
<td>0.0350</td>
<td>0.0191</td>
<td>0.0514</td>
<td>0.0203</td>
<td>0.0168</td>
<td><strong>0.0168</strong></td>
</tr>
<tr>
<td></td>
<td>ERGAS</td>
<td>539.87</td>
<td>79.04</td>
<td>108.33</td>
<td>183.29</td>
<td>56.31</td>
<td>59.14</td>
<td>66.58</td>
<td>45.00</td>
<td>42.66</td>
<td><strong>42.66</strong></td>
</tr>
<tr>
<td>R20_S20</td>
<td>PSNR</td>
<td>20.84</td>
<td>27.84</td>
<td>25.43</td>
<td>29.15</td>
<td>29.65</td>
<td>28.76</td>
<td>28.26</td>
<td>30.65</td>
<td>37.79</td>
<td><strong>37.79</strong></td>
</tr>
<tr>
<td></td>
<td>SSIM</td>
<td>0.2064</td>
<td>0.3021</td>
<td>0.6351</td>
<td>0.5418</td>
<td>0.5575</td>
<td>0.5078</td>
<td>0.4775</td>
<td>0.6108</td>
<td>0.9467</td>
<td><strong>0.9467</strong></td>
</tr>
<tr>
<td></td>
<td>SAM</td>
<td>1.0120</td>
<td>0.6017</td>
<td>0.7734</td>
<td>0.7625</td>
<td>0.7874</td>
<td>0.8028</td>
<td>0.8322</td>
<td>0.8029</td>
<td>0.2212</td>
<td><strong>0.2212</strong></td>
</tr>
<tr>
<td></td>
<td>ERGAS</td>
<td>375.62</td>
<td>163.80</td>
<td>260.09</td>
<td>149.66</td>
<td>139.77</td>
<td>159.15</td>
<td>161.04</td>
<td>148.49</td>
<td>65.05</td>
<td><strong>65.05</strong></td>
</tr>
</tbody>
</table>

Figure 10. Real HSI image **urban** noise removal results. (a) Band 138. Restoration results by (b) BM3D, (c) PARAFAC, (d) LRMR, (e) BM4D, (f) TDL, (g) ITSReg, (h) LLRT, (i) HSI-DeNet with single band as input, and (j) HSI-DeNet with multiple bands as input.

Figure 11. Extension to the MODIS image mixed noise removal. (a) Terra MODIS image band 30. (b) HSI-DeNet.
channel of the model to accommodate the HSIs. For the fair comparison, both of our method and modified DnCNN are trained with same dataset on ICVL. The hyper-parameters are set as the rule in the original paper. Here, we give both the visual and quantitative comparison.

We compare the denoising results on 10 ICVL sub-images with the size of 380*380*10. We show one image as an example in Fig. 12. From the spatial visual performance, our method and DnCNN are quite similar to each other. From the spectral features, the spectral curve of our method is obviously closer to that of the original one. We could also obtain the same conclusion from the quantitative assessments. It is worth noting that, due to the additional multi-channel strategy, our model is more representative for the spectral features. That is the main reason why our model is more suitable for HSIs restoration task. And along with the dilated convolution, our method also obtains better performance in the spatial structural reconstruction.

6) Quantitative Comparison: We show the quantitative comparison results of Fig. 5 and Fig. 6 in Table 11. We can observe that the HSI-DeNet obtains the best results under different noise levels. Moreover, with the increase noise level, the advantages of HSI-DeNet over other methods becomes bigger. The main reason is that the other methods all utilize the local or non-local information from the degraded image itself. When the noise level increases, the internal information has been heavily damaged, resulting in the degeneration of their performances. On the contrary, our HSI-DeNet could benefit from the external dataset for better restoration performance.

C. Real Noise Removal

Here, we select the widely used urban dataset to test the performance of HSI-NeDet in real HSIs. Urban is one of the most widely used hyperspectral data. There are 307 × 307 pixels, each of which corresponds to a 2 × 2 m² area. In this image, there are 210 wavelengths ranging from 400nm to 2500nm, resulting in a spectral resolution of 10nm. We extract ten bands (bands from 129 to 138 as the input). Only a few bands in urban dataset are affected by slightly random and stripe noise (mostly horizontal). To better show the results, we increased the noise level by adding both random and stripe noises (vertical) in all bands, as shown in Fig. 10(a). It is worth noting that the result of Fig. 10(i) is corresponding to HSI-DeNet with only the single image band 138 as the input. The other methods are with ten image bands as the input. From Fig. 10 we have two main observations. First, our single image Fig. 10(i) and multiple image Fig. 10(j) based HSI-DeNet all obtain the best visual appearance. While the results of other methods are with unexpected artifacts, such as the stripe residual in Fig. 10(b),(f),(g),(h). Second, the result of Fig. 10(i) is a bit over-smoothed, compared with Fig. 10(j). This is a powerful proof that our HSI-DeNet could benefit from the spectral correlation in HSIs.

Further, in Fig. 11 we show our HSI-DeNet could also be applied to other remote sensing images, such as MODIS. Here we use the MODIS Terra image band 30 as the test image [Fig. 11(a)]. Moreover, from the satisfactory result in Fig. 11(b) and also Fig. 11(i), we could conclude that the HSI-DeNet fully utilizes the spatial information and benefits from the learned pattern from the external dataset. This might give a new insight for conventional model-based HSI restoration methods.

D. Study of HSI-DeNet

1) Effectiveness of the Discriminator: We performed an experiment to validate the effectiveness of the discriminator. As shown in Fig. 13 we compared the denoising result of HSI-DeNet [Fig. 13(d)] with its corresponding GAN [Fig. 13(e)]. Compared with other methods, the HSI-DeGAN obtained the image with the sharper edge and fewer artifacts from the visual appearance. However, the quantitative assessments of HSI-DeGAN are a little inferior to other methods. This is the main characteristic of the adversarial network that it could obtain photo-realistic image but poor quantitative result [64, 65].

2) Effectiveness for Post-processing: We use the unsupervised k-mean classification method to demonstrate that the proposed model can facilitate the subsequent processing. The number of the classification is five. The maximum iteration is set five times. Figure 14 shows the denoising and classification results. The top row is the degraded Salinas image and the recovery results by MP [75], LRMR [40], and the proposed method. The bottom row is the corresponding classification results. In Fig. 14(c) and (d), obvious classification errors can be seen in the original classification result. On the contrary, the restored classification image in Fig. 14(e) does not contain any random or line artifacts. It can be seen that the classification results are significantly improved after the destriping and denoising process. This demonstrates the restoration process is successfully applied.

3) Generalization Analysis: Note that, our model is trained on the ICVL dataset. We have tested our trained model for various datasets, including the ICVL, CAVE and urban. The trained model worked well for the test images such as the urban data with different spectral bands ranging from 1690nm to 1780nm. And our method could also well applied to other real HSI data, such as the Indian Pine shown in Fig. 16. These experiments demonstrate the generalization ability of
Figure 13. Effectiveness of the discriminator. (a) Original image. (b) Degraded image. Denoising results by (c) BM4D, (d) HSI-DeNet, and (e) HSI-DeGAN. The first row is the whole image, and the second row shows its zoom-in results.

Table IV

<table>
<thead>
<tr>
<th>Image Size</th>
<th>Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BM3D</td>
</tr>
<tr>
<td>180<em>180</em>10</td>
<td>1.863</td>
</tr>
<tr>
<td>380<em>380</em>10</td>
<td>14.233</td>
</tr>
<tr>
<td>512<em>512</em>10</td>
<td>24.821</td>
</tr>
</tbody>
</table>

Figure 14. Classification results by k-mean classification method. The first row show the images and the second row show the corresponding classification results.

Figure 15. Comparison with state-of-the-arts HSI denoising methods in terms of both speed and performance for image with size 180*180*10.
our model for various HSIs. In fact, our model is sensitive to the noise level and category. We will discuss the limitations in the next subsection.

Although the trained model on ICVL is robust to various HSIs, we perform an experiment to illustrate that the fine-tuning strategy could further boost the final restoration performance. We trained our model on ICVL and then fine-tuned it on the CAVE (32 scenes) where 90 percent is used as training and 10 percent used as test. We show the quantitative results Toy under noise level $\sigma = 20$ and $S = 20$ between without and with fine-tuning in table IV. We can observe that the fine-tune obviously improves the restoration performance in terms of the spectral information. That is to say the fine-tuned model is more adaptive to the specific imaging system. However, it is a little surprising that the PSNR value is even slightly inferior to that of the without fine-tuning, and the SSIM has been slightly improved. We speculate this is due to the lack of training samples in CAVE. Moreover, a large part of the images in CAVE is all dark with zero values. Thus, the fine-tuned model may be under-fitting due to the insufficient spatial information. The experiment can demonstrate that the fine-tuning strategy is a very effective way for HSIs dataset with different spectral bands ranging. These experiments could demonstrate the generalization ability of our model to some extent.

![Real AVIRIS image denoising results.](image)

**Figure 16.** Real AVIRIS image denoising results. Although our model is trained on the ICVL dataset, it works well for numerous simulated and real HSIs datasets with different spectral bands ranging. These experiments could demonstrate the generalization ability of our model to some extent.

<table>
<thead>
<tr>
<th>Table V</th>
<th>EFFECTIVENESS OF FINE-TUNING STRATEGY.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>PSNR</td>
</tr>
<tr>
<td>Without Fine-tune</td>
<td>37.79</td>
</tr>
<tr>
<td>Fine-tune</td>
<td>37.54</td>
</tr>
</tbody>
</table>

4) Depth of the Network: We also analyze the influence of the number of the convolutional layers. In Fig. 17 we show the training loss of the model with different depth. Here, we just choose the model with 3, 5, 10, and 18 layers as an representation. We have two observations here. First, at the first 50 epochs, the training loss of the model with fewer layers is usually lower. We guess that the shallow models are easier to be trained. Second, with the sufficiently training, the deeper the model is, the lower training loss it is. Moreover, we give the quantitative assessment in Table VI. That is to say the depth of the model do facilitate the training of the HSIs, and improve the restoration performance.

![The relationship between the training loss and depth of our network.](image)

**Figure 17.** The relationship between the training loss and depth of our network.

![The analysis of the regularization parameter $\lambda_2$. (a) The loss curve of the generator. (b) The loss curve of the discriminator.](image)

**Figure 18.** The analysis of the regularization parameter $\lambda_2$. (a) The loss curve of the generator. (b) The loss curve of the discriminator.

<table>
<thead>
<tr>
<th>Table VI</th>
<th>QUANTITATIVE ANALYSIS THE RELATIONSHIP BETWEEN THE TRAINING LOSS AND DEPTH OF OUR NETWORK.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model Depth</td>
<td>PSNR</td>
</tr>
<tr>
<td>3 Layer</td>
<td>40.80</td>
</tr>
<tr>
<td>5 Layer</td>
<td>41.05</td>
</tr>
<tr>
<td>10 Layer</td>
<td>41.30</td>
</tr>
<tr>
<td>18 Layer</td>
<td>41.58</td>
</tr>
</tbody>
</table>
for all methods, as shown in Fig. [15]. Our method achieves the best performance in both of the speed and PSNR value.

6) Regularization Parameter: The Eq. [7] contains two terms: the reconstruction term and the adversarial term. Looking back to the framework flowchart in Fig. [4], the physical meaning of first term is to guarantee the intensity similarity between the restored image and ground truth. The physical meaning of the second term is to work as a learnable discriminative prior, which desires the distribution of restored data to be adaptively similar to that of the ground truth. Keeping this in mind, it is a guidance for us to set the parameters in Eq. [7] and control the importance of each term. Normally, we set $\lambda_1 = 1$ and adjust the regularization parameter $\lambda_2$.

Here, we explore how the regularization parameter $\lambda_2$ influence the reconstruction performance. We show the training loss of the reconstruction term by changing the $\lambda_2$ in [0.001, 0.01, 0.1, 1]. In Fig. [18] with the increasing of $\lambda_2$, the training loss of the reconstruction term gradually increase. This is reasonable. With the increasing of $\lambda_2$, the model will pay less attention to the reconstruction error. That is to say, the reconstruction error becomes larger, and the PSNR value gradually decreases with the increasing of $\lambda_2$. For extreme case, when $\lambda_2 = 0$, Eq. [7] degenerates to the conventional reconstruction loss as in Eq. [5]. However, as the goal of the second term controlled by $\lambda_2$ is not PSNR value oriented, it is for the perceptual appearance, as shown in Fig. [13] As a tradeoff, we empirically set $\lambda_2 = 0.01$.

7) Training Convergence: We show the training loss of both the generator and discriminator to judge the training convergence. For the generator, we can observe from Fig. [17] and Fig. [18](a), the training loss decreases and converges to a very low value at different depth and different parameter. For the discriminator, we used the discriminator similar to that of the DCGAN [76]. However, the loss of the DCGAN is meaningless [77]. We plot the adversarial loss under different $\lambda_2$ in Fig. [18](b). As we expected, the loss curve does not exhibit any regularity. As far as we know, only the WGAN [77] could provide the reasonable curve/guidance for the discriminator convergence judgement. We would like to try more sophisticated GAN, such as WGAN as our future work.

8) Limitation: Right now, the trained HSI-DeNet is only suitable for the specific noise level. For example, as shown in Fig. [19] if we want to restore the degraded HSI with noise level $\sigma = 50$, we have to use the trained HSI-DeNet with HSI data degraded with $\sigma = 50$. The denoising result in Fig. [19](b) by HSI-DeNet_50 is with satisfactory appearance. However, if we import the noise image with either $\sigma = 25$ or $\sigma = 75$ to the $\sigma = 50$ based trained HSI-DeNet, we could observe that the results [Fig. [19](c) and (d)] are either over-smoothed or with residual noise. This heavily restricts the application of our HSI-DeNet for real HSIs with the unknown noise level. Fortunately, the recent researches have working toward this direction by training one single network for general restoration task [78], [79]. We will take this point as our future work.

The training of our HSI-DeNet has better to be performed on the GPU platform. Although our model could also be trained on the CPU, it requires several days to be completed. Larger training datasets are needed to improve the generalization of our model to accommodate all kinds of complex noise category and HSI data. Right now, we do not train one single model for all situations. Instead, we train each model for one specific task. The trained model obtains its best performance for the specific noise category.

V. CONCLUSION

In this work, we introduce the deep convolutional neural network to remove the mixed noise in HSIs. The proposed method significantly advances the HSIs restoration field in three aspects: complex noise modeling, spectral-spatial structure preserving and the running time. The implicit yet powerful representative ability of the CNN enables us to better model the mixed noise in HSIs. The learned 2D filters with multiple channels inherently match the multi-dimensional property of HSIs, better preserving the spatial-spectral structure correlation. Last but not least, the simple operations in HSI-DeNet make the algorithm extremely fast for testing. Our method has been tested on various simulated and real HSIs, and achieved better restoration performance than compared methods in terms of both quantitative and qualitative assessments. In the future, we would like to introduce the real 3D CNN [80] into the HSIs restoration, and also extend the CNN method to other interesting HSI tasks, such as super-resolution, unmixing and so on.

REFERENCES


