SelfS2: Self-Supervised Transfer Learning for Sentinel-2 Multispectral Image Super-Resolution

Xiao Qian, Tai-Xiang Jiang, and Xi-Le Zhao

***Abstract*—The multispectral image captured by the Sentinel-2 satellite contains 13 spectral bands with different resolutions, which may hider some of the subsequent applications. In this article, we design a novel method to super-resolve 20- and 60-m coarse bands of the S2 images to 10 m, achieving a complete dataset at the 10-m resolution. To tackle this inverse problem, we leverage the deep image prior expressed by the convolution neural network (CNN). Specifically, a plain ResNet architecture is adopted, and the 3-D separable convolution is utilized to better capture the spatial–spectral features. The loss function is tailored based on the degradation model, enforcing the network output obeying the degradation process. Meanwhile, a network parameter initialization strategy is designed to further mine the abundant fine information provided by existing 10-m bands. The network parameters are inferred solely from the observed S2 image in a self-supervised manner without involving any extra training data. Finally, the network outputs the super-resolution result. On the one hand, our method could utilize the high model capacity of CNNs and work without large amounts of training data required by many deep learning techniques. On the other hand, the degradation process is fully considered, and each module in our work is interpretable. Numerical results on synthetic and real data illustrate that our method could outperform compared state-of-the-art methods.**

***Index Terms*—Deep image prior, self-supervised learning, Sentinel-2 satellite, separable 3-D convolution, super-resolution.**

I. Introduction

SATELLITE remote sensing multispectral images (MSIs), which provide abundant spectral information, are widely employed for different applications, e.g., environment monitoring [1], [2], flood detection [3], land use and land cover classification [4], and change detection [5], [6], [7]. For many reasons, such as the physical limitations of the radiometric resolution of the imaging sensors, design considerations, and achieving a higher signal-to-noise ratio (SNR), the spatial resolution [or say, the ground sampling distance (GSD)] of a single remote-sensing MSI captured by some well-known satellites, including the Moderate Resolution Imaging Spectroradiometer [8], the Advanced Spaceborne Thermal Emission and Reflection Radiometer [9], and the recently deployed Sentinel-2 (S2) launched by the European Space Agency (ESA) [10], might be different across different spectral bands. As all such resolution differences would not go away with hardware improvements [11], computational super-resolving those coarser bands so as to have all bands available at the highest spatial resolution is of paramount importance.

Without loss of generality, this article focuses on the S2 satellite. An MSI collected by the S2 satellite contains 13 spectral bands (443–2190 nm), covering the visible-near infrared and short-wave infrared wavelengths with three different resolutions, as shown in Table I. The super-resolution problem of S2 MSIs is to increase the resolution of bands at 20- and 60-m spatial resolution to the maximal resolution (i.e., 10 m). The presence of fine bands makes super-resolving S2 images similar to the conventional pansharpening problem. In [12], existing component substitution [13], [14], [15] and multiresolution analysis [16], [17], [18], [19], [20] methods designed for pansharpening are extended to super-resolve bands at 20 m of S2 images via integrating four fine bands into a single band. A geostatistical approach, the area-to-point regression kriging approach [21], [22], is also extended. However, as pointed out in [11] and [23], the super-resolution of S2 images differs from the conventional pansharpening problem [20], [24], [25], [26], [27], [28] and the hyperspectral image/MSI fusion problem [28], [29], [30], [31] mainly for the high-spatial-resolution bands (i.e., the panchromatic image in pansharpening and the MSI in hyperspectral image/MSI fusion), spectrally overlapping the lower resolution bands in these two tasks, while this condition is not met by the images of S2 as the bands at the resolution of 10 m and the bands at the resolution of 20/60 m in S2 imagery are almost nonoverlapping (see Table I). Thus, novel methods tailored for S2 images super-resolution are expected.

Recent works super-resolve coarse bands via inverting the observation model that depicts the degradation (downsampling, blurring, and noise) of S2 images in the imaging process. Lanaras et al. [11] set up a joint observation model for S2 images across different bands. With the S2 images in a vector form,

Manuscript received 8 February 2022; revised 28 May 2022, 19 July 2022, and 16 September 2022; accepted 20 November 2022. Date of publication 28 November 2022; date of current version 7 December 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 12001446, Grant 61876203, and Grant 12171072, in part by the Natural Science Foundation of Sichuan, China, under Grant 2022NSFSC1798 and Grant 2022NSFSC0507, in part by the Applied Basic Research Project of Sichuan Province under Grant 2021YJ0107, in part by the Key Project of Applied Basic Research in Sichuan Province under Grant 2020YJ0216, in part by the National Key Research and Development Program of China under Grant 2020YFA0714001, in part by the Sichuan Science and Technology Project under Grant 2021ZYD0021, in part by the Fundamental Research Funds for the Central Universities, and in part by the Guanghua Talent Project. *(Corresponding author: Tai-Xiang Jiang.)*

Xiao Qian and Tai-Xiang Jiang are with the School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics, Chengdu 610074, China, and also with the Kash Institute of Electronics and Information Industry, Kashgar 844000, China (e-mail: qianxiao1047@gmail.com; taixiangjiang@gmail.com).

Xi-Le Zhao is with the Research Center for Image and Vision Computing, School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: xlzhao122003@163.com).

Digital Object Identifier 10.1109/JSTARS.2022.3224987

TABLE I
13 SENTINEL-2 BANDS [12]

Band	B1	B2	B3	B4	B5	B6	B7	B8	B8a	B9	B10	B11	B12
Center wavelength (nm)	443	490	560	665	705	740	783	842	865	945	1380	1610	2190
Width (nm)	20	65	35	30	15	15	20	115	20	20	30	90	180
Spatial Resolution (m)	60	10	10	10	20	20	20	10	20	60	60	20	20

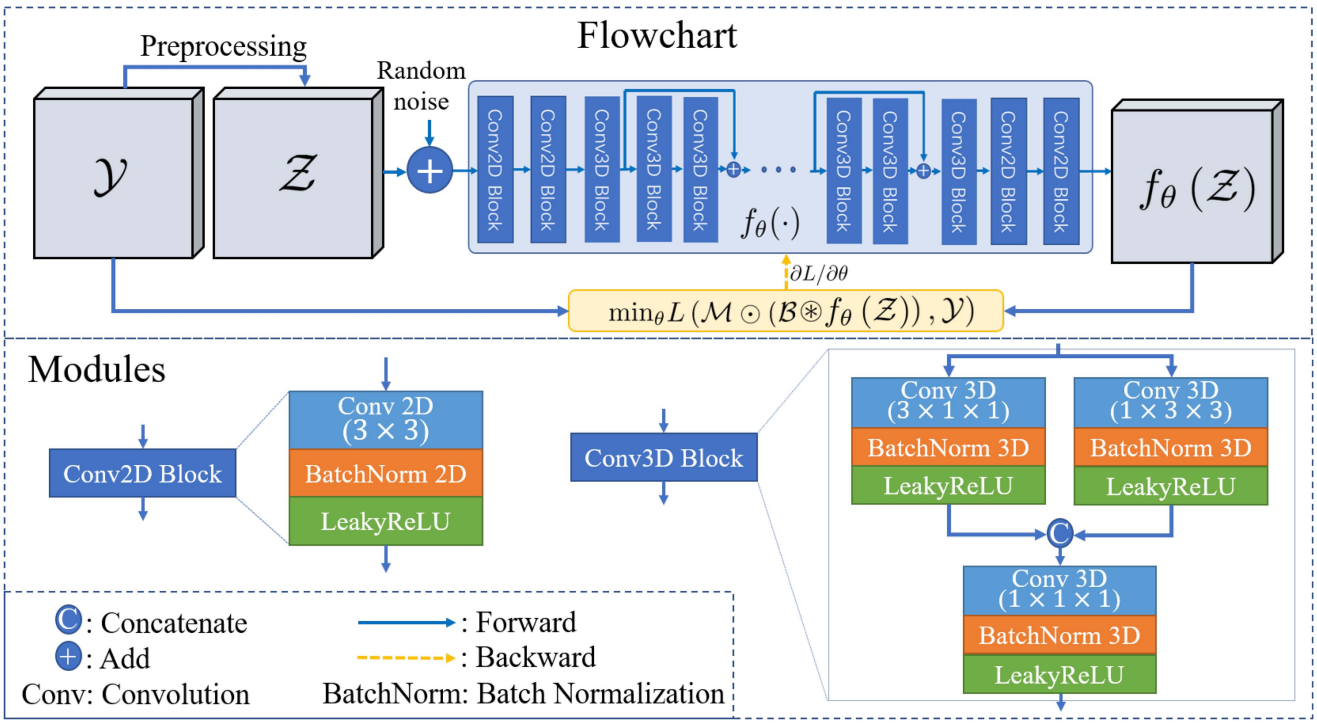

Fig. 1. Flowchart of the proposed method and the network architecture.

the downsampling and blurring processes are characterized as two block diagonal matrices with each block associated with a specific band. Especially, when the blur is assumed to be spatially invariant, each block of the blur matrix is a block-circulant circulant-block matrix associated with the point spread function (PSF) of the corresponding band. In [11], the variance of Gaussian blur is computed from the calibrated modulation transfer function released by the ESA [32]. This observation model has also been rewritten in the tensor form by Wang and Ji [33].

The inversion of the observation model is obviously ill-posed [34]. Therefore, introducing the prior knowledge of the underlying fine bands via regularization is required. Three types of prior knowledge can be found. In [11], the spatial *local* discontinuity of four fine bands in existence is encoded as a weighted total variation (TV) regularizer to sharpen the 20- and 60-m bands. Furthermore, Paris et al. [35] utilize the spatial *nonlocal* self-similarity under the plug-and-play framework. In [35], a CBM3D [36] denoiser is plugged in, and this version of CBM3D could compute the patch similarity from a reference image, which is obtained from the linear combination of four fine bands. In [23] and [33], the nonlocal self-similarity is exploited by formulating explicit regularizers. Another important property of the underlying high-resolution S2 images is that they are *globally* correlated. That is, MSIs are living in a low-dimensional subspace. Thus, low-rank subspace representation is employed in [11], [23], and [35], and Ulfarsson et al. [37] turn to reduce the rank of the data matrix.

Lanaras et al. [38] directly learn a deep convolutional neural network (CNN), which can be viewed as the inverse mapping of the observation model mentioned above, with the training data synthesized based on the scale invariance assumption. For example, when super-resolving the 20-m bands, the original 10-m bands and 20-m bands are downsampled by a factor of 2. Thus, the ground-truth "high-resolution" with the same resolution (at 20 m) across all bands at a reduced scale for training/validation/testing is then gathered together with the existing 20-m bands. Then, this dataset serves for the fully supervised training for 2× super-resolution. However, in [38], the 2× super-resolution and 6× super-resolution are separately conducted, and the network architecture for 6× super-resolution is much deeper. Palsson et al. [39], [40] and Gargiulo et al. [41] also adopt this strategy, and the deep residual network (ResNet) [42] and the generative adversarial network [43] are taken into consideration.

Generally, deep-learning-based methods are believed to maintain a high model capacity to represent information in need [44]. However, existing S2 super-resolution approaches using deep

CNNs, e.g., [38], [39], [40], face one unavoidable challenge of their generalization ability. Those model-based methods [11], [23], [33], [35], [37] could be scene adaptive as the prior knowledge (global/local/nonlocal) they considered is widespread in different scenes. Thus, to yield excellent performance with global applicability, methods in [38], [39], and [40] need to collect training data for different scenes as much as possible. Subsequently, this would also lead to a large volume of training data and a corresponding high computation burden for training.

To address the above issues, this article proposes a self-supervised learning method for S2 MSIs super-resolution. Our contributions could be summarized as follows.

1) First, to tackle the Sentinel-2 image super-resolution, we employ the deep image prior expressed by the CNN with a high model capacity. Specifically, we adopt a plain deep residual CNN architecture with the 3-D separable convolution. This structure is expected to adaptively capture a great deal of low-level MSI statistics for different scenes.
2) Second, the loss function is established based on the degradation model of S2 images. Thus, our results naturally conform to the degradation. Meanwhile, as the fine bands at 10-m resolution are available, we initially infer the network parameters with fine bands and then transfer them for super-resolving coarser bands. The overall learning stage is fully in a self-supervised manner, which means the inferring of network parameters is solely from the observed S2 MSI, and no extra training data are needed. Extensive experimental results are carried out to illustrate the effectiveness of the proposed method.

The rest of this article is organized as follows. Section II presents the proposed method in terms of the network architecture, the loss function, and details of learning. Section III reports experimental results and discussions. Finally, Section IV concludes this article.

II. Method

In this section, we first review the degradation model of S2 images. Then, we establish the framework of our super-resolution method, including the loss function and the specific network architecture. Subsequently, our learning scheme for S2 images is tailored.

A. Problem Formulation

As the input and output of our network are all in the tensor format in our implementation, we describe the degradation model of S2 images in the tensor format. Hence, before formulating the S2 super-resolution problem, we first introduce the tensor notations we used throughout this article. Lowercase letters are used for scalars, e.g., x; boldface lowercase letters are used for vectors, e.g., $\mathbf{x}$; boldface uppercase letters are used for matrices, e.g., $\mathbf{X}$; Euler script letters are used for tensors, e.g., $\mathcal{X}$. Given a third-order tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, we use $\mathcal{X}_{ijk}$ to denote its (i, j, k)th element. The kth frontal slice of $\mathcal{X}$ is denoted as $\mathcal{X}^{(k)}$ (or $\mathcal{X}(:, :, k)$, $\mathbf{X}^k$).

Note that the spatial size and the PSF of blurring matrices for different bands of S2 images could be different. Formulating the degradation model of S2 images is tricky [23]. In the following, we reformulate the degradation model proposed in [11] in the tensor format with referring to [33]. For notational simplicity, we first conduct plain upsampling via the Kronecker product to ensure that the observed images are of the same spatial size. For example, for a band at the 20-m resolution, denoted as $\mathbf{X} \in \mathbb{R}^{\frac{M}{2} \times \frac{N}{2}}$, we upsample it via

$$\mathbf{X} \otimes \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{M \times N} \tag{1}$$

where $\otimes$ denotes the Kronecker product. Then, for S2 images having $B = 12$ spectral bands[1] and the spatial size $M \times N$ (at 10-m resolution), after the plain upsampling, it can be written as a third-order tensor $\mathcal{Y} \in \mathbb{R}^{M \times N \times B}$. The order of the bands in $\mathcal{Y}$ is from B1 to B12 (except B10), in accord with Table I. That is, $\mathcal{Y}(:, :, i)$ is at the spatial resolution of 60 m for $i \in \{1, 10\}$ and 20 m for $i \in \{5, 6, 7, 9, 11, 12\}$, respectively. For simplicity, we denote sets as

$$s_1 = \{2, 3, 4, 8\},\ s_2 = \{5, 6, 7, 9, 11, 12\},\ \text{and}\ s_6 = \{1, 10\}.$$

Then, the degradation model, up to the noise, could be expressed as

$$\mathcal{Y} = \mathcal{M} \odot (\mathcal{B} \circledast \mathcal{X}) \tag{2}$$

where $\mathcal{Y} \in \mathbb{R}^{M \times N \times B}$ is the observed tensor, $\mathcal{M} \in \mathbb{R}^{M \times N \times B}$ is the binary mask tensor representing the downsampling process, $\odot$ denotes the elementwise multiplication, $\mathcal{B} \in \mathbb{R}^{h \times h \times B}$ is the Gaussian kernel tensor, $\circledast$ indicates the slicewise convolution, and $\mathcal{X} \in \mathbb{R}^{M \times N \times B}$ is the underlying MSI with the highest spatial resolution across each band. The ith frontal slice of $\mathcal{B}$ (i.e., $\mathcal{B}(:, :, i) \in \mathbb{R}^{h \times h}$) is the Gaussian blur kernel corresponding to the ith S2 band. For $i \in s_1$, $\mathcal{B}(:, :, i)$ is the unit discrete impulse function. That is, the convolution between $\mathcal{X}(:, :, i)$ and $\mathcal{B}(:, :, i)$ returns $\mathcal{X}(:, :, i)$ for $i \in s_1$. Similar to $\mathcal{Y}$, frontal slices of $\mathcal{M}$ are constructed via

$$\mathcal{M}(:, :, i) = \begin{cases} \texttt{ones}(M, N), & i \in s_1 \\ \texttt{ones}(\frac{M}{2}, \frac{N}{2}) \otimes \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, & i \in s_2 \\ \texttt{ones}(\frac{M}{6}, \frac{N}{6}) \otimes \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, & i \in s_6 \end{cases} \tag{3}$$

where $\texttt{ones}(m, n)$ indicates generating a matrix of the size $m \times n$ with all entries equaling to 1. The construction of $\mathcal{M}$ also indicates that the downsampling the conducted via selecting the element in the top-left corner.

B. Proposed Method

The super-resolution of S2 images aims at recovering the high-resolution MSI $\mathcal{X}$ from the observation $\mathcal{Y}$. On the one hand,

[1] In this article, we do not consider the B10 band at 60-m resolution, which is used for atmospheric correction [45].

with pre-estimated $\mathcal{B}$ and a given $\mathcal{M}$, previous model-based S2 super-resolution methods [11], [23], [33], [35], [37] can be formulated as

$$\hat{\mathcal{X}} = \arg\min_{\mathcal{X}} L\left(\mathcal{M} \odot (\mathcal{B} \circledast \mathcal{X}), \mathcal{Y}\right) + \phi\left(\mathcal{X}\right) \tag{4}$$

where L is the fidelity term and $\phi(\mathcal{X})$ is the regularization term. L enforces the super-resolution result being in accordance with the degradation model, and the regularization term is formulated to express the prior knowledge of the underlying results. On the other hand, those deep-learning-based methods can be viewed as learning the mapping function

$$\hat{\mathcal{X}} = \texttt{CNN}\left(\mathcal{Y}\right) \tag{5}$$

from a large amount of training data.

Our method attacks the S2 super-resolution problem in another way and can be formulated as

$$\min_{\theta} L\left(\mathcal{M} \odot (\mathcal{B} \circledast f_\theta\left(\mathcal{Z}\right)), \mathcal{Y}\right) \tag{6}$$

where $f_\theta(\cdot)$ is a specific CNN with the network parameter θ, $L(\cdot)$ is the loss function, and $\mathcal{Z}$ is the network input. As for the network input $\mathcal{Z}$, we follow the strategy of using the degraded image in [46]. However, directly inputting the $\mathcal{Y}$ constructed via the Kronecker product in (2) would not be helpful for extracting spatial and spectral features since there are too many zero entries. Therefore, instead of directly using $\mathcal{Y}$, we use a simple spatial bicubic interpolation [47] to fill those zeros entries within 20- and 60-m bands. Meanwhile, to better utilize the global low dimensionality of the MSI, the bicubic interpolation result is then projected to a low-dimensional subspace detected via the HySime algorithm [48] and projected back. For simplification, this preprocessing step can be denoted as $\mathcal{Z} = \text{Preprocessing}(\mathcal{Y})$. As we will show in the experimental part, other preprocessing techniques also work.

We then explain the distinctions and connections of our method to (4) and (5). For convenience, we rewrite (6) as

$$\min_{\theta} L\left(\mathcal{M} \odot (\mathcal{B} \circledast \mathcal{X}), \mathcal{Y}\right), \text{s.t. } f_\theta\left(\mathcal{Z}\right) = \mathcal{X} \tag{7}$$

where $\mathcal{X}$ is an auxiliary variable to represent the network output.

On the one hand, we can see that our method is different from previous CNN-based methods, which directly map the observed low-resolution image to a high-resolution result. The parameters of the CNN in (5) are learned from large datasets of images via computing the loss between the network output and the synthetic high-resolution via gradient descent and backpropagation [49]. Our method is to infer the network parameters solely from a single observed low-resolution image by minimizing the distance between the network output after degradation and the low-resolution image. That is, the network input $\mathcal{Z}$, which is obtained from the observed MSI via simple preprocessing techniques, in (7) is always fixed across the overall training procedure. It can be viewed as our training set containing only one image, and this single image would be repeatedly used many times. Thus, our method is in a self-supervised learning manner, being different from previous fully supervised deep learning methods.

On the other hand, our method is closer to the model in (4). The first term $L(\mathcal{M} \odot (\mathcal{B} \circledast \mathcal{Z}), \mathcal{Y})$ is in the same manner as the fidelity term in (4). Both of them can enforce the result $\mathcal{X}$ to be in accordance with the degradation model (2). The main difference lies in the regularization term. In (4), $\phi(\cdot)$ represents the regularizer to express prior knowledge, such as TV for spatial local discontinuity and CBM3D for nonlocal self-similarity. In our method, $f_\theta(\cdot)$ is herein to express the deep image prior [44], which widely exists in many types of visual data. That is, the CNN $f_\theta(\cdot)$ itself serves as the regularization term. Meanwhile, (4) can be solved via traditional optimization methods, while this is not suitable for (7).

The super-resolution result is obtained by $\hat{\mathcal{X}} = f_\theta(\mathcal{Z})$ after (6) is optimized.[2] The flowchart of our method is shown in Fig. 1.

C. Network Architecture and Loss Function

Those two parts, i.e., $f_\theta(\cdot)$ and $L(\cdot)$, corresponding to the fidelity term and regularization term, are vital for successful recovery. In the following, we will introduce the specific architecture of $f_\theta(\cdot)$ and tailor the loss function $L(\cdot)$.

1) Network Architecture: As pointed out by Ulyanov et al. [44], the image statistics prior could be well captured by the structure of a CNN independent of learning. That is, the structure of the CNN could express the prior knowledge of the natural images. Hence, the selection of CNN's structure is of great importance. In this article, we adopt a simple deep ResNet [42] structure with separable 3-D convolutional blocks. Meanwhile, the input of our network is $\mathcal{Z}$ constructed from the observation $\mathcal{Y}$. These are different from [44], in which a generator network with a U-net [50] like structure is employed, and the random noise is taken as the input. For an input $\mathcal{Z} \in \mathbb{R}^{M \times N \times 12}$, the first two 2-D convolution blocks would increase the 2-D features and extend the feature dimension to the size 256. Then, a ResNet structure with 34 separable 3-D convolutional blocks follows. Finally, two 2-D convolution blocks are in series to decrease the feature dimension to 12. The 2-D convolution block consists of a cascade of 2-D convolution layer, a 2-D batch normalization [51] layer, and the LeakyReLU [52] as nonlinear activation function.

Considering that the cubic nature of the underlying MSI and the pattern along the spectral direction would be different from that along with spatial modes, we exploit the separable 3-D convolution [53] blocks[3] to construct the main pipeline of $f_\theta(\cdot)$. We would illustrate that the separable 3-D convolution is superior to 2-D convolution or plain 3-D convolution in the experimental part. The (separable) 3-D convolution block (Conv3D Block) is shown in Fig. 1. It is implemented via separating a 3-D convolution operation into a 2-D spatial convolution and a 1-D spectral convolution in sequence. The separable 3-D convolution is expected to simultaneously capture the spatial and spectral information with fewer filter coefficients.

[2]Generally, only local minima could be achieved.

[3]Different from the initial form of separable 3-D convolution in [53], we omit the skip connection within the block as the overall network structure has already been in a ResNet style.

2) Loss Function: In our work, minimizing the loss function $L(\mathcal{M} \odot (\mathcal{B} \circledast f_\theta(\mathcal{Z})), \mathcal{Y})$ rectifies the deviation between the network output after degradation and the observation. It would enhance the network output $f_\theta(\mathcal{Z})$, which involves the deep image prior expressed by the network structure to comply with the observation model. From the perspective of maximum *a posteriori*, the fidelity term should be designed based on the probability of the observation, given the underlying high-quality MSI related to the noise distribution. For example, the ℓ_2 norm in [11], [23], and [35] or the Frobenius norm in [33] accounts for the independent and identically distributed Gaussian noise. In this article, inspired by a recent work for the image super-resolution [54], in which the ℓ_1 norm has been proved to be better in preserving image edges and textures, we adopt the ℓ_1 norm in our loss function. In the meantime, the structural similarity (SSIM) [55] loss is added to the loss function to enhance the structural feature extraction ability of f_θ for a better preservation of the structural information. Thus, our loss function turns to be

$$\begin{aligned}\mathcal{L} &= \alpha_1 \mathcal{L}_1 + \alpha_2 \mathcal{L}_{\text{SSIM}} \\ &= \alpha_1 \left\| \mathcal{M} \odot (\mathcal{B} \circledast f_\theta(\mathcal{Z})) - \mathcal{Y} \right\|_1 \\ &\quad + \alpha_2 \text{SSIM}\left(\mathcal{M} \odot (\mathcal{B} \circledast f_\theta(\mathcal{Z})), \mathcal{Y}\right) \end{aligned} \tag{8}$$

where $\|\cdot\|_1$ is the ℓ_1 norm defined as the sum of absolute values and α_1 and α_2 are set to 1. The choice of α_1 and α_2 is discussed in Section III-C5.

D. Parameter Initialization and Implementation Details

The ultimate goal of our work is to infer a nonlinear mapping function $f_\theta(\cdot)$, which could map the observed MSI with different spatial resolutions to a desired MSI with the highest resolution across different bands. Empirically, the network f_θ is expected to learn high-resolution features from 10-m bands and help to super-resolve the 20- and 60-m bands. Considering that the observation contains four 10-m bands, we proposed the following network parameter initialization strategy to further take advantage of those high-resolution bands.

First, we simulate pseudo low-resolution bands from those 10-m bands. For the jth band ($j \in s_2 \cup s_6$) of the observation, i.e., $\mathcal{Y}^{(j)}$ or $\mathcal{Y}(:,:,j)$, we solve the following linear regression problem:

$$\min_{\beta_2^j, \beta_3^j, \beta_4^j, \beta_8^j} \left\| \sum_{i \in s_1} \beta_i^j \left(\mathcal{M}^{(i)} \odot \left(\mathcal{B}^{(i)} \circledast \mathcal{Y}^{(i)} \right) \right) - \mathcal{Y}^{(j)} \right\|_F^2 \tag{9}$$

where $\mathcal{M}^{(i)}, \mathcal{B}^{(i)}$, and $\mathcal{Y}^{(i)}$ are the ith frontal slice of $\mathcal{M}, \mathcal{B}$, and $\mathcal{X}$, respectively. Equation (9) aims at finding the optimal linear combination of high-resolution bands to express the low-resolution bands. For a given $j \in s_2 \cup s_6$, we obtain the regression coefficients $\beta_2^j, \beta_3^j, \beta_4^j$, and β_8^j. Then, a pseudo low-resolution input $\mathcal{Y}_{\text{low-pseudo}}$ is simulated as

$$\mathcal{Y}_{\text{low-pseudo}}^{(j)} = \begin{cases} \sum_{i \in s_1} \beta_i^j \left(\mathcal{M}^{(i)} \odot \left(\mathcal{B}^{(i)} \circledast \mathcal{Y}^{(i)} \right) \right), & \text{if } j \in s_2 \cup s_6 \\ \mathcal{Y}^{(j)}, & \text{if } j \in s_1 \end{cases}. \tag{10}$$

Correspondingly, we could simulate a pseudo high-resolution MSI $\mathcal{Y}_{\text{high-pseudo}}$ as

$$\mathcal{Y}_{\text{high-pseudo}}^{(j)} = \begin{cases} \sum_{i \in s_1} \beta_i^j \left(\mathcal{Y}^{(i)} \right), & \text{if } j \in s_2 \cup s_6 \\ \mathcal{Y}^{(j)}, & \text{if } j \in s_1 \end{cases}. \tag{11}$$

The above strategy to simulate the pseudo high-resolution MSI is indeed the band-synthesized scheme of the hypersharpening framework proposed in [56].

Then, the parameters of our network f_θ are random initialized and subsequently trained via minimizing the following loss function:

$$\| f_\theta(\mathcal{Y}_{\text{low-pseudo}}) - \mathcal{Y}_{\text{high-pseudo}} \|_1 \tag{12}$$

using the ADAM optimizer [57]. After 1000 epochs of training with a learning rate of 0.02, we stop it and use the network parameters at this time as the initialization for minimizing (8) in the main training phase. This initialization stage directly enforces the network output close to a (pseudo) high-resolution MSI when inputting the (pseudo) low-resolution observation. As we will illustrate in the experiment part, it significantly improves the super-resolution ability of f_θ.

Our method is implemented with the PyTorch [58] framework on a desktop of GPU NVIDIA GeForce RTX 2080Ti with 11-GB GDDR6 RAM. For each iteration (epoch), following the setting in [44], an independently generated zero-mean Gaussian noise with the standard deviation $\sigma = \frac{1}{30}$ is added to the input (after the bicubic interpolation and the subspace projection). Finally, in the main training stage, the learning rate is set to be 0.02 with 1000 epochs.

III. Experiments

In this section, we compare our method with other state-of-the-art methods on the synthetic data and real data. Compared methods are listed as follows:

1) the simple bicubic interpolation;
2) SupReME [11], a method utilizing the discontinuities of higher resolution bands via a weighted TV regularizer;
3) MuSA [35], a method employing the C-BM3D denoiser to express the nonlocal self-similarity;
4) NSTMR [33], a method explicitly formulating the nonlocal self-similarity regularization regularizer in the tensor format;
5) DSen2 [38], a CNN-based method directly learning the mapping from degraded images to high-resolution results.

DSen2 is implemented on the TensorFlow framework with the pretrained network[4] provided by the authors. As DSen2 is pretrained on data synthesized from the real S2 images with the scale invariance assumption, directly applying DSen2 on our synthetic data would be unfair. Thus, we have retrained DSen2 following the training scheme provided in [38]. The only difference lies in that we use the degradation model (2) to generate low-resolution and high-resolution MSI pairs. The remaining model-based methods are implemented on MATLAB.

[4][Online]. Available: https://github.com/lanha/DSen2

TABLE II
QUANTITATIVE ASSESSMENTS OF ALL COMPETING METHODS ON THE SYNTHETIC DATA

Data	Index	Bicubic	SupReME	MuSA	DSen2	NSTMR	SelfS2
AVIRIS-City	RMSE	0.0823	0.0288	0.0278	0.0329	0.0210	**0.0206**
	SSIM	0.6462	0.9461	0.9492	0.9294	**0.9767**	0.9753
	SRE	18.26	28.35	28.66	27.44	31.12	**31.30**
	UIQI	0.6572	0.9504	0.9538	0.9302	0.9778	**0.9782**
	SAM	4.4060	2.2140	2.2609	2.2298	1.6999	**1.6443**
AVIRIS-Coast	RMSE	0.0511	0.0235	0.0237	0.0377	**0.0176**	0.0181
	SSIM	0.8179	0.9438	0.9475	0.8952	**0.9795**	0.9707
	SRE	22.33	29.71	29.07	26.01	**32.22**	31.6
	UIQI	0.5996	0.7425	0.7705	0.7937	0.7806	**0.8015**
	SAM	2.8045	1.7535	1.6647	2.3718	**1.217**	1.2266
AVIRIS-Crops	RMSE	0.0313	0.0123	0.0126	0.0138	0.0122	**0.0089**
	SSIM	0.8827	0.9598	0.9767	0.9515	0.9754	**0.9835**
	SRE	27.61	35.18	34.81	33.96	35.42	**37.79**
	UIQI	0.6561	0.7914	0.8486	0.8035	0.8266	**0.8813**
	SAM	1.8797	0.8869	0.8717	1.0967	0.8105	**0.5506**
AVIRIS-Montain	RMSE	0.0578	0.0221	0.0220	0.0388	**0.0169**	0.0379
	SSIM	0.7534	0.9564	0.9660	0.941	**0.9826**	0.9776
	SRE	21.77	30.62	30.24	27.84	**32.93**	25.97
	UIQI	0.7557	0.9615	0.9711	0.9427	**0.9846**	0.9807
	SAM	2.9471	1.5876	1.5787	2.3755	1.3586	**1.3404**
HYDICE-WDC	RMSE	0.0553	0.0158	0.0284	0.0218	0.0165	**0.0099**
	SSIM	0.7193	0.9767	0.9234	0.9505	0.9811	**0.9906**
	SRE	17.67	28.93	25.37	23.1993	28.96	**32.62**
	UIQI	0.7147	0.9785	0.9329	0.9498	0.9803	**0.9915**
	SAM	4.6437	1.8820	4.3418	3.0139	2.0127	**1.2653**
HYDICE-Terrain	RMSE	0.0552	0.0204	0.0316	0.0199	0.0164	**0.0112**
	SSIM	0.7212	0.9558	0.9073	0.9774	0.9806	**0.9917**
	SRE	20.48	29.98	27.58	26.20	31.60	**34.40**
	UIQI	0.7194	0.9593	0.9155	0.9774	0.9784	**0.9917**
	SAM	3.0761	1.7337	3.0386	1.3551	1.5858	**1.0370**
APEX-Baden	RMSE	0.0764	0.0302	0.0304	0.0414	0.0304	**0.0204**
	SSIM	0.4320	0.8908	0.8812	0.8708	0.8711	**0.9527**
	SRE	12.11	21.87	22.51	19.36	22.91	**25.68**
	UIQI	0.4068	0.8929	0.8815	0.8732	0.8653	**0.9550**
	SAM	8.8328	4.8797	5.1757	5.3612	5.3846	**3.7550**

The best values and the second-best values are respectively highlighted by bolder fonts and underlines.

For all the model-based methods, their parameters are manually tuned for their best performances. We would exhibit the results on the synthetic data and real data and then conduct ablation studies to examine the effect from every single pipeline of the proposed method.

A. Results on the Synthetic Data

As the ground-truth super-resolved S2 images with 10-m resolution at all 12 bands are inaccessible, it is needed to simulate the synthetic high-resolution S2 images. However, the simulation is tricky since both the spectral and spatial properties should be simultaneously considered. Fortunately, previous research works have addressed this in a reasonable way. For example, Paris et al. [35] employ the hyperspectral images captured by the NASA Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor [59]. The detailed steps can be seen in [35, Sec. IV-A] and [23, Sec. IV-B]. We restate key steps here for readers' convenience. In this work, being the same as [35], we select four AVIRIS images corresponding to spatial regions of city (at 3.5-m GSD), coast (at 5-m GSD), crop (at 3.2-m GSD), and mountain (at 5-m GSD). The first step is to spatially filter those AVIRIS images with a Gaussian kernel (with standard deviation of 1.5 for city data, 1.2 for coast and mountain data, and 1.6 for crop data). Then, the second step is the spatial downsampling (with a factor of 3 for city data, 2 for coast and mountain data, and 3 for crop data). After this stage, the hyperspectral images are approximately reaching the 10-m GSD. The AVIRIS hyperspectral sensor provides 224 narrow contiguous spectral bands from 0.4 to 2.5 μm [59], covering that of Sentinel-2 images. Thus, the final step is to simulate the spectral properties of S2 images by applying its spectral response to the 224-band hyperspectral images. Consequently, we can obtain 12-band simulated ground-truth S2 images (respectively, denoted as "AVIRIS-City," "AVIRIS-Coast," "AVIRIS-Crops," and "AVIRIS-Montain"). Their spatial sizes are all 406×108 at the 10-m bands.

In the meantime, other three simulated ground-truth images[5] in [33] are also taken as the ground truth . Two of them are generated from the HYDICE images of Washington DC Mall and Terrain (respectively, denoted as "HYDICE-WDC" and "HYDICE-Terrain") with 2.8-m spatial resolution, and the remaining one comes from the airborne prism experiment (APEX) [60] image of Baden (denoted as "APEX-Baden") with 1.8-m spatial resolution. The spatial sizes of these three synthetic S2 images at the 10-m bands are 96×96. Those three ground-truth images are generated by downsampling hyperspectral images to obtain a spatial resolution of approximately 10 m and selecting 12 bands with the same wavelength as the S2 satellite.

Then, simulated S2 images are generated from the synthetic ground-truth data following the degradation model in (2). Specifically, the size of the Gaussian blur kernels is 10×10 across different low-resolution bands, and standard deviations are set according to ESA's data quality report on Sentinel-2 satellite products [32] per spectrum. We remark here that we also set the kernel size as 10×10 when running compared methods in [11], [33], and [35] for a fair comparison. For each band, the Gaussian noise is added to control the SNR to 40 dB.

After having the pairs of ground-truth images and simulated S2 images, we utilize five quality metrics to quantitatively measure the results by different methods. They are: 1) the SSIM [55]; 2) the root-mean-square error (RMSE); 3) the signal to reconstruction error ratio (SRE); 4) the universal image quality index [61] (UIQI); and 5) the spectral angle mapper (SAM). For $i \in s_2 \cup s_6$, we denote the ith band of the ground truth as $\mathbf{X}^i \in \mathbb{R}^{M \times N}$ and the ith band of the super-resolved result as $\hat{\mathbf{X}}^{\in} \mathbb{R}^{M \times N}$. Then, the SSIM value of the ith band is computed via

$$\text{SSIM} = \frac{(2\mu_x \mu_{\hat{x}} + c_1)(2\sigma_{x\hat{x}} + c_2)}{(\mu_x^2 + \mu_{\hat{x}}^2 + c_1)(\sigma_x^2 + \sigma_{\hat{x}}^2 + c_2)} \tag{13}$$

where μ_x and $\mu_{\hat{x}}$ are the average values of $\mathbf{X}_i$ and $\hat{\mathbf{X}}_i$, respectively, σ_x, σ_y, and σ_{xy} are the standard deviation of $\mathbf{X}_i$, $\hat{\mathbf{X}}_i$, and covariance of $\mathbf{X}_i \hat{\mathbf{X}}_i$, respectively, and c_1 and c_2 are two constants, respectively, defined as $(k_1 L)^2$ and $(k_2 L)^2$ with the

[5]The code of [33] is kindly provided by the authors.

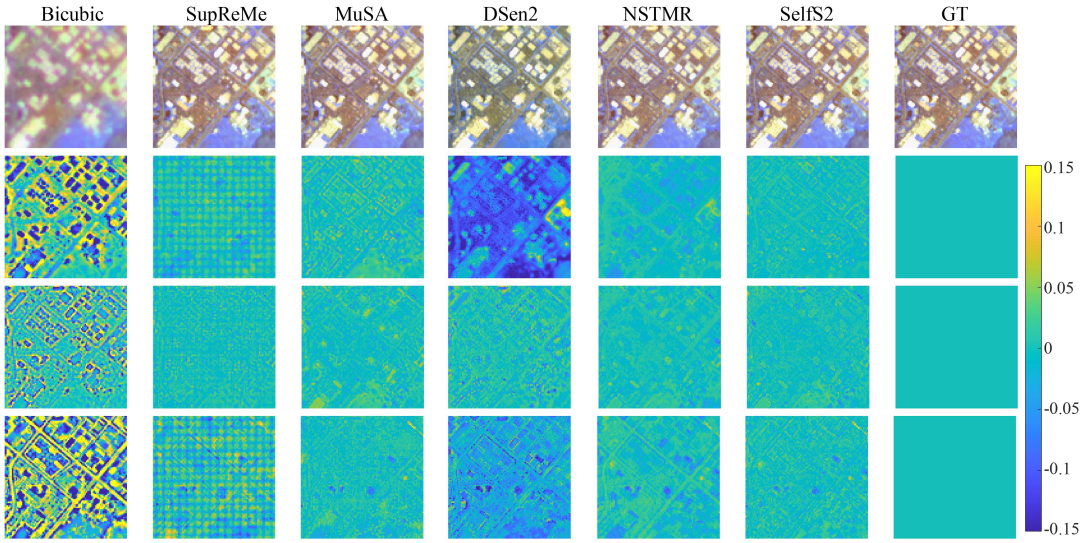

Fig. 2. Top row shows the pseudo-color images composed of B1 (red), B5 (green), and B9 (blue) bands of results by different methods on the data AVIRIS-City and the ground truth (GT). The second to bottom rows illustrate the residual images of these three bands, respectively.

range of per pixel L and size of sliding windows k_1 and k_2[6]. Then, the SRE value of the ith band is computed as

$$\text{SRE} = 10\log_{10}\frac{\|\mathbf{X}^i\|_F^2}{\|\hat{\mathbf{X}}^i - \mathbf{X}^i\|_F^2}. \tag{14}$$

Next, the UIQI value of the ith band is computed as

$$\text{UIQI} = \frac{4\sigma_{x\hat{x}}\mu_x\mu_{\hat{x}}}{(\sigma_x^2+\sigma_{\hat{x}}^2)(\mu_x^2+\mu_{\hat{x}}^2)}. \tag{15}$$

We compute the average values of the SSIM, SRE, and UIQI across bands in $s_2 \cup s_6$. After denoting the ground truth as $\mathcal{X} \in \mathbb{R}^{M\times N\times B}$ and the super-resolved result as $\mathcal{X} \in \mathbb{R}^{M\times N\times B}$, the RMSE and SAM values are, respectively, computed via

$$\text{RMSE} = \sqrt{\frac{1}{MNB}\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{k=1}^{B}\left(\mathcal{X}_{ijk} - \hat{\mathcal{X}}_{ijk}\right)^2} \tag{16}$$

and

$$\text{SAM} = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\arccos\left(\frac{\left\langle\hat{\mathcal{X}}(i,j,:),\mathcal{X}(i,j,:)\right\rangle}{\|\hat{\mathcal{X}}(i,j,:)\|_2\|\mathcal{X}(i,j,:)\|_2}\right). \tag{17}$$

In Table II, we exhibit quantitative metrics of the super-resolved results by different methods. We can see that our method is comparable to NSTMR on the data simulated from the AVIRIS hyperspectral images, as they rank the first and second places in most cases. As for the data simulated from HYDICE and APEX hyperspectral images, our method achieves the best performance on account of all the quantitative metrics, while NSTMR and SupReME alternately obtain the second-best values. We can also see that the performance of DSen2 is not the best, although it employs a CNN with massive parameters. This collaboratively validates our analysis in the previous part that DSen2 could not well fit the degradation model (2) well since all the data in this subsection are degenerated in accord with (2).

In Figs. 2–5, we exhibit the pseudo-color images composed of three bands of results on AVIRIS-City, AVIRIS-Crops, HYDICE-WDC, and HYDICE-Terrain, respectively. Considering that the spatial sizes of AVIRIS-City and AVIRIS-Crops are a little bit big, we only clip square areas of the size 108×108 from the results for better visualization and space saving. The selected bands to make the pseudo-color images always consist of, at least, one 60-m band. In the meantime, the corresponding error images are also shown. From Figs. 2 and 3, we can see that errors are generally related to edges. For AVIRIS-City, SupReMe and our SelfS2 perform well on the B5 band, whereas MuSA and our SelS2 obtain better results on the B9 band. For the 60-m band B1, the error of the result by our method is the lowest. For AVIRIS-Crops, all the compared methods have a good performance on the B6 band, while our method obtains the lowest error on both the B1 and B11 bands. In Fig. 4, results of the B6 and B7 bands by our method and SupReMe are the best and our method super-resolves the B9 band (60 m) of HYDICE-WDC better than compared methods. It can be easily found in Fig. 5 that our method achieves the best performance on all of the illustrated bands. From those visualizations, we can see that the superior of our method is more obvious on the 60-m bands.

B. Results on Real Data

In this part, we test all the methods on two real datasets: Verona and Malmo. The spatial size of them is 180×180 at the 10-m bands. For the real data, we only display the visual results. The pseudo-color images composed of three bands on Malmo and Verona are shown in Figs. 6 and 7, respectively. We can see that

[6] L, k_1, and k_2 are using the default settings of the python package *scikit-image*. See https://scikit-image.org/docs/stable/auto_examples/transform/plot_ssim.html?highlight=ssim for more details.

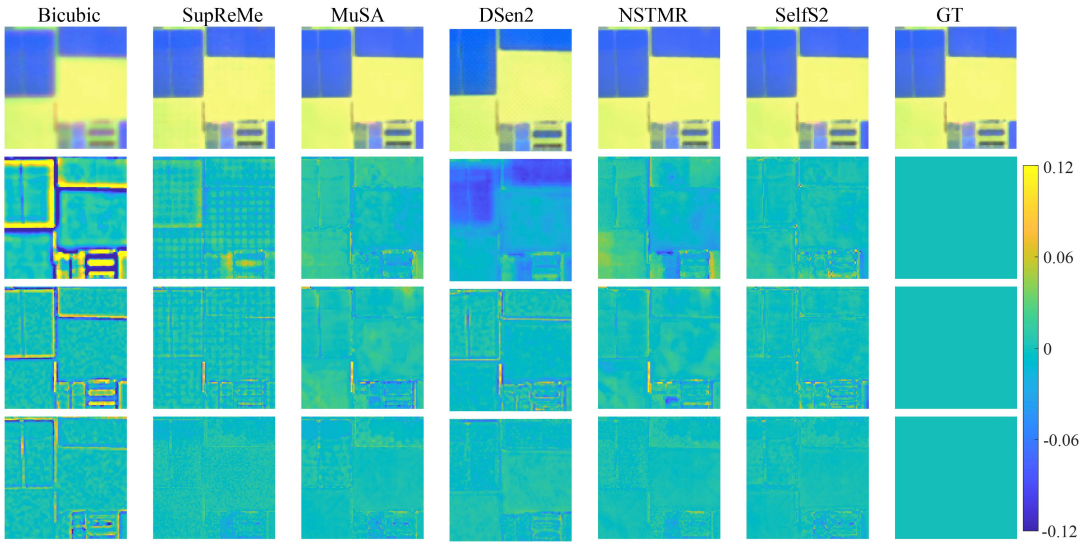

Fig. 3. Top row shows the pseudo-color images composed of B1 (red), B11 (green), and B6 (blue) bands of results by different methods on the data AVIRIS-Crops and the ground truth (GT). The second to bottom rows illustrate the residual images of these three bands, respectively.

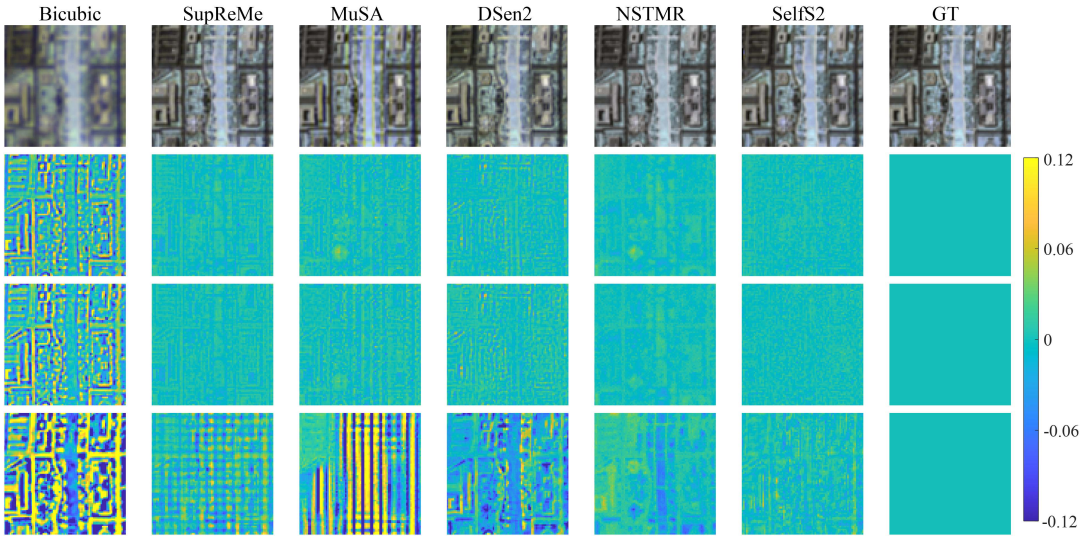

Fig. 4. Top row shows the pseudo-color images composed of B6 (red), B7 (green), and B9 (blue) bands of results by different methods on the data HYDICE-WDC and the ground truth (GT). The second to bottom rows illustrate the residual images of these three bands, respectively.

all the methods work and generate sharper results compared with the bicubic interpolation for the real data.

From Fig. 6, it can be found that the result by MuSA is too smooth that some image details are blurred. Also, we can see color distortions of the result by DSen2, and its result looks a little bit blurry. The color styles of the results by the bicubic interpolation, MuSA, and DSen2 tend to be similar, and this color difference would be related to their spatial blurring as these three results are visually blurry. SupReMe, NSTMR, and our SelfS2 all obtain good results. We can see from the area boxed by the red-dashed line that our result is more clear.

In Fig. 7, we highlighted two areas of all the results with red-dashed circles. The results by DSen2 and MuSA are more blurry for this real data compared with other methods. We can see from the center of large red-dashed circles that MuSA, NSTMR, and our SelfS2 preserve the orange and round area well, and only our method and NSTMR fully reconstruct the small line, which is composed of red and cyan blocks, right above the orange and round area. From the small red-dashed circles, the color distortion of the result by NSTMR is obvious compared with the bicubic interpolation. More specifically, the dot, which is in the orange color of the bicubic interpolation, tends to be red in the result of NSTMR.

C. *Ablation Study and Parameter Analysis*

In this part, we conduct the ablation study and parameter analysis to test the effects of important modules of our framework

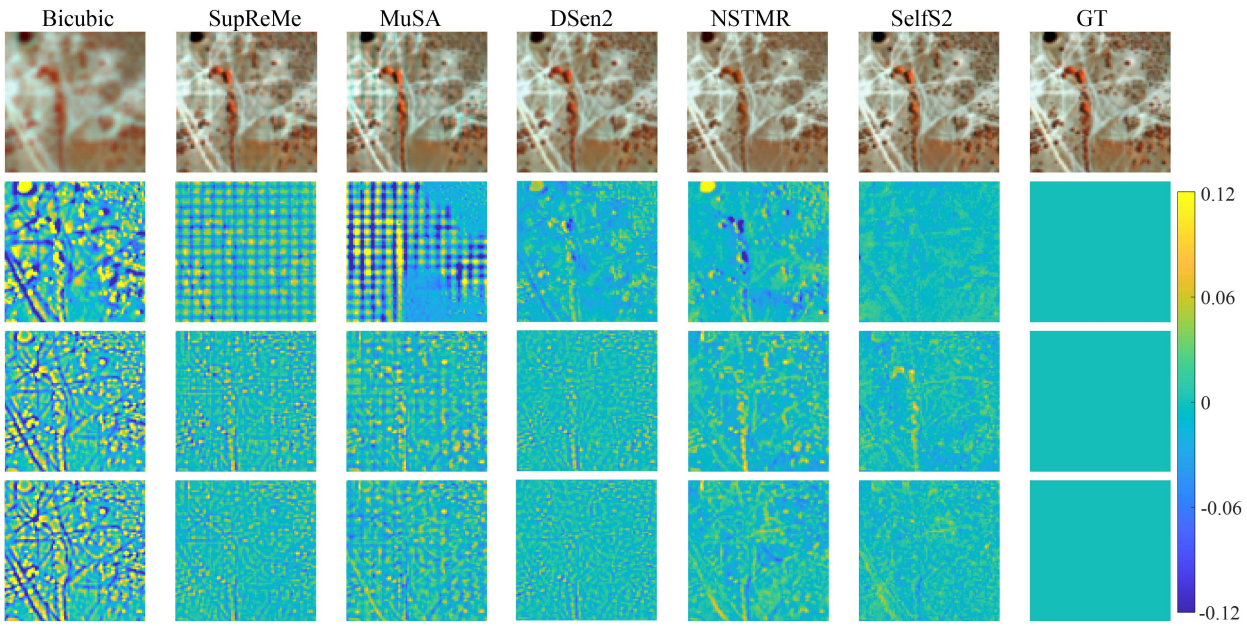

Fig. 5. Top row shows the pseudo-color images composed of B9 (red), B11 (green), and B12 (blue) bands of results by different methods on the data HYDICE-Terrain and the ground truth (GT). The second to bottom rows illustrate the residual images of these three bands, respectively.

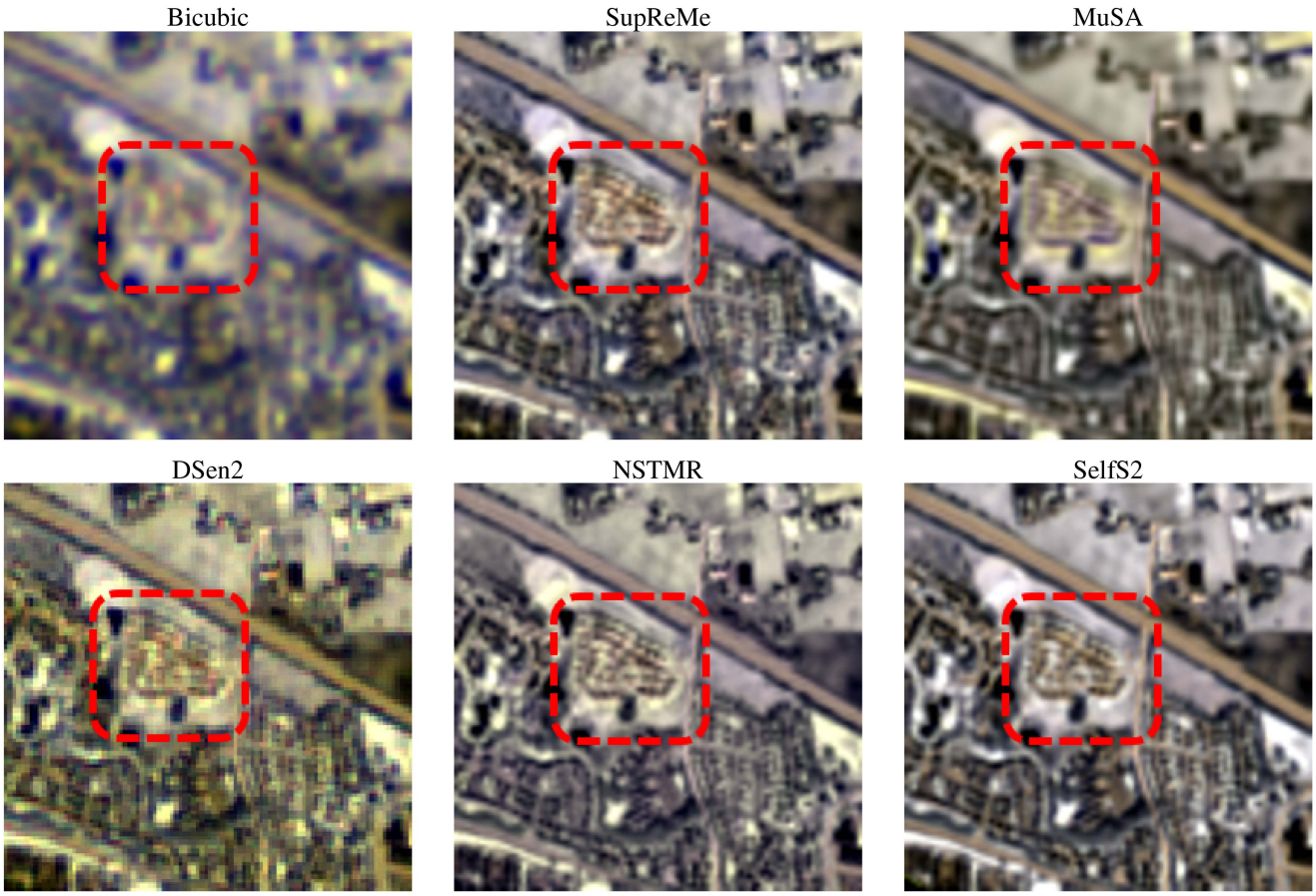

Fig. 6. Pseudo-color images created with bands B6, B7, and B10 by different methods for Malmo. The bicubic, the input, results by DSen2, MuSA, SupReME, S2Sharp, NSTMR, and SelfS2.

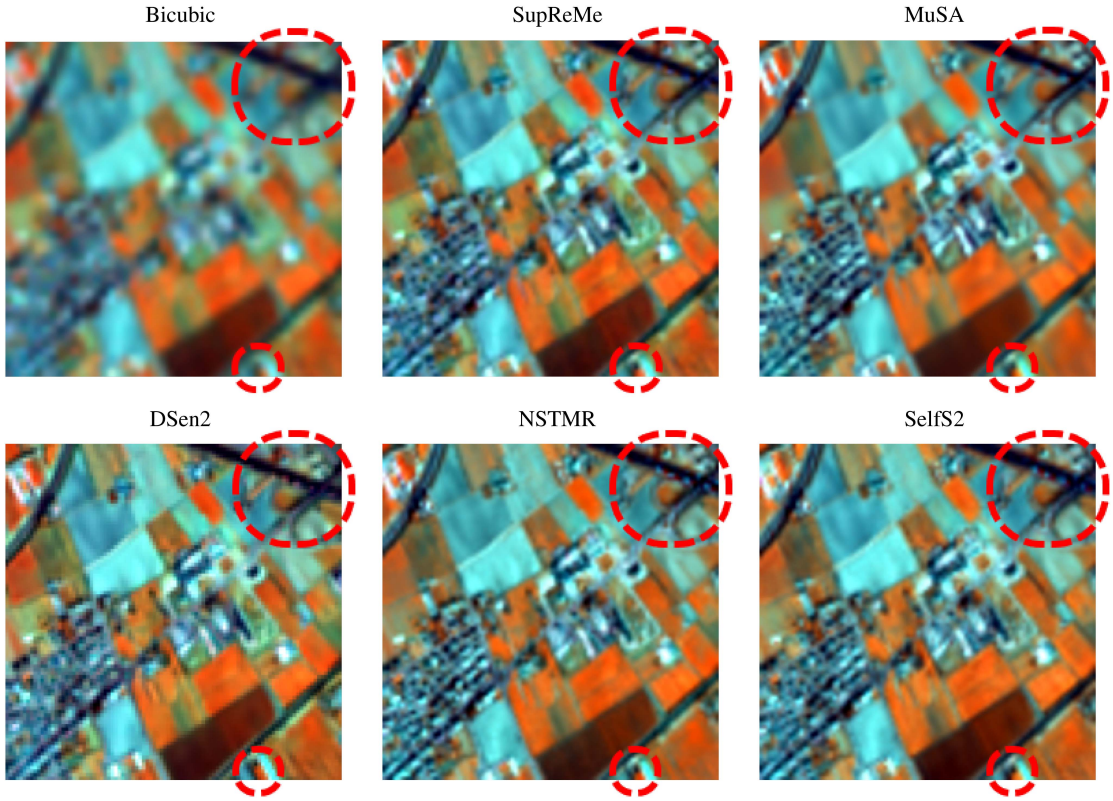

Fig. 7. Pseudo-color images created with bands B6, B11, and B12 by different methods for Verona. The bicubic, the input, results by DSen2, MuSA, SupReME, S2Sharp, NSTMR, and SelfS2.

and the sensibility of parameters. Modules and parameters to be examined are: 1) the network input; 2) the network structure; 3) the loss function; 4) the network parameter initialization strategy; 5) the weighting parameters α_1 and α_2 in the loss function (8); and 6) the value of the network input regularization parameter σ. All the experiments in the part are conducted on the synthetic data HYDICE-WDC and HYDICE-Terrain.

1) Network Input: As the initial work of deep image prior [44] adopts a generator network with the random input, we test our method with the network input randomly produced obeying the Gaussian distribution with the standard deviation $\sigma = 1/10$. Meanwhile, we also adopt the bicubic interpolation and results by MuSA, SupReMe, and NSTMR as the network input. Table III shows the results of our method with different network inputs. We can see that for this S2 super-resolution problem, the bicubic interpolation is more suitable than the random input. After projected into the low-dimensional subspace detected by HiSime [48], although this projection step may delete some spectral anomalies, the bicubic interpolation could help our method to obtain the best performance.

2) Loss Function: The loss function rectifies the deviation between the network output after degradation and the observation, enforcing the network output to obey the degradation model. To test the effect of different types of loss functions, we select four frequently used loss functions, i.e., the ℓ_1 loss, the ℓ_2 (or known as the mean square error) loss, the SSIM loss, and the perceptual loss[7] [63]. All these mentioned loss functions are computed with $\mathcal{M} \odot (\mathcal{B} \circledast f_\theta(\mathcal{Y})) - \mathcal{Y}$ in (8). In Table IV, we report the quantitative metrics of the results by our method with different loss functions and their combinations. We can see that the SSIM loss is helpful for obtaining better SSIM and UIQI values. Using the combination of the ℓ_1 loss and the SSIM loss could achieve the best performance.

TABLE III
QUANTITATIVE ASSESSMENTS OF OUR METHOD WITH DIFFERENT *NETWORK INPUTS* ON THE SYNTHETIC DATA HYDICE-WDC AND HYDICE-TERRAIN

Data	Index	Random (Gaussian)	Bicubic	MuSA	SupReME	NSTMR	Bicubic+ HySime
HYDICE-WDC	RMSE	0.0202	0.0110	0.0128	0.0112	0.0118	**0.0099**
	SSIM	0.9665	0.9888	0.9850	0.9883	0.9882	**0.9906**
	SRE	26.28	31.65	30.22	31.62	31.14	**32.62**
	UIQI	0.9715	0.9900	0.9867	0.9898	0.9895	**0.9915**
	SAM	2.1615	1.3574	1.4170	1.4039	1.4373	**1.2653**
HYDICE-Terrain	RMSE	0.0745	0.0158	0.0126	0.0128	0.0119	**0.0112**
	SSIM	0.9815	0.9861	0.9898	0.9912	0.9901	**0.9917**
	SRE	31.40	33.04	33.27	17.61	33.63	**34.40**
	UIQI	0.9824	0.9881	0.9897	0.9912	0.9904	**0.9917**
	SAM	1.4510	1.0811	1.1608	1.3999	1.0796	**1.0370**

The best values and the second-best values are highlighted by bolder fonts and underlines, respectively.

[7]We adopt the output of last Conv2d layer before ReLU from a pretrained VGG16 [62] to calculate the perceptual loss.

TABLE IV
QUANTITATIVE ASSESSMENTS OF OUR METHOD WITH DIFFERENT *LOSS FUNCTIONS* ON THE SYNTHETIC DATA HYDICE-WDC AND HYDICE-TERRAIN

Data	Index	ℓ_1	ℓ_1+ℓ_2	ℓ_1+ℓ_2 +SSIM	ℓ_1+ℓ_2+SSIM +Perception	ℓ_1+SSIM	ℓ_1+SSIM +Perception	ℓ_2	ℓ_2+SSIM	ℓ_2+SSIM +Perception	SSIM
HYDICE-WDC	RMSE	0.0218	0.0139	0.0107	0.0105	**0.0099**	0.0112	0.0124	0.0103	0.0106	0.0103
	SSIM	0.9747	0.9873	0.9904	0.9895	0.9906	0.9892	0.9804	0.9901	0.9890	**0.9908**
	SRE	29.00	31.30	31.86	32.01	**32.62**	32.02	29.90	32.09	32.10	32.45
	UIQI	0.9777	0.9885	0.9913	0.9906	0.9915	0.9902	0.9831	0.9914	0.9900	**0.9918**
	SAM	1.9218	1.4242	1.3334	1.3055	**1.2653**	1.3552	1.7305	1.3478	1.3769	1.2967
HYDICE-Terrain	RMSE	0.0167	0.0117	0.0486	0.0117	**0.0112**	0.0118	0.0120	0.0124	0.0119	0.0116
	SSIM	0.9784	0.9914	0.9905	0.9914	**0.9917**	0.9913	0.9907	0.9900	0.9909	0.9915
	SRE	30.80	33.95	21.21	33.93	**34.40**	33.83	33.70	33.60	33.80	34.15
	UIQI	0.9805	0.9915	0.9876	0.9916	**0.9917**	0.9915	0.9908	0.9899	0.9910	0.9916
	SAM	1.4018	1.0744	1.1870	1.0642	**1.0370**	1.0642	1.1049	1.1665	1.0717	1.0770

The best values and the second-best values are highlighted by bolder fonts and underlines, respectively.

TABLE V
QUANTITATIVE ASSESSMENTS OF OUR METHOD WITH DIFFERENT *NETWORK STRUCTURES* ON THE SYNTHETIC DATA HYDICE-WDC AND HYDICE-TERRAIN

Data	Index	RMSE	SSIM	SRE	UIQI	SAM
HYDICE-WDC	Hourglass+2D Conv	0.0527	0.8904	18.15	0.9000	3.2958
	Hourglass + 3D Conv	0.0109	0.9893	31.79	0.9903	1.3222
	Hourglass + Sep 3D Conv	0.0110	0.9882	31.72	0.9893	1.3570
	ResNet + 2D Conv	0.0628	0.9384	25.49	0.9660	16.0164
	ResNet + 3D Conv	0.0110	0.9895	31.90	0.9907	1.3687
	ResNet + Sep. 3D Conv	**0.0099**	**0.9906**	**32.62**	**0.9915**	**1.2653**
HYDICE-Terrain	Hourglass + 2D Conv	0.0146	0.9856	31.89	0.9860	1.2696
	Hourglass + 3D conv	0.0117	0.9914	33.98	0.9913	1.0962
	Hourglass + Sep. 3D Conv	0.0122	0.9891	33.67	0.9895	1.1378
	ResNet + 2D Conv	0.0140	0.9908	33.59	0.9908	1.1263
	ResNet + 3D Conv	0.0122	0.9912	34.30	0.9860	1.1045
	ResNet + Sep. 3D Conv	**0.0112**	**0.9917**	**34.40**	**0.9917**	**1.0370**

The best values and the second-best values are highlighted by bolder fonts and underlines, respectively.

TABLE VI
QUANTITATIVE ASSESSMENTS OF OUR METHOD WITH DIFFERENT *NETWORK PARAMETER INITIALIZATIONS* ON THE SYNTHETIC DATA HYDICE-WDC AND HYDICE-TERRAIN

Data	Index	RMSE	SSIM	SRE	UIQI	SAM
HYDICE-WDC	Random initialization	0.0157	0.9747	28.94	0.9772	1.9714
	With parameter initialization	**0.0099**	**0.9906**	**32.62**	**0.9915**	**1.2653**
HYDICE-Terrain	Radom initialization	0.0130	0.9863	33.03	0.9865	1.1738
	With parameter initialization	**0.0112**	**0.9917**	**34.40**	**0.9917**	**1.0370**

The best values are highlighted by bolder fonts.

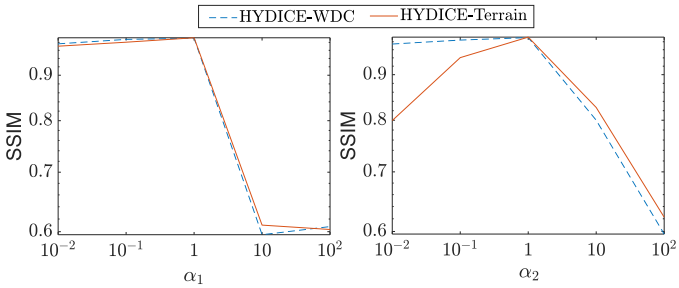

Fig. 8. SSIM values of the results on synthetic data HYDICE-WDC and HYDICE-Terrain with respect to different α_1 and α_2 in the loss function (8).

3) Network Structure: As reported in [44], the priors expressed by different network structures could be slightly different, and both the U-net like (or say hourglass) network work and the ResNet are adequate for natural images. Thus, in this part, we test our method with these two typical network structures. In the meantime, three types of the basic convolution block are tested. They are the common 2-D convolution, the 3-D convolution, and the separable 3-D convolution. Table V shows the performance of our method with different network structures. We can see that, when the overall structure is the hourglass, using the 3-D convolution could achieve better performance than using separable 3-D convolution. The ResNet could outperform the hourglass network, and it could obtain the best result when using separable 3-D convolution.

4) Network Parameter Initialization: In Section II-D, we elaborate a parameter initialization strategy, in which the network parameters are first initialized by fully utilizing high-resolution bands. Table VI shows the results of our method with this strategy or directly using the random initialization of network parameters. We can see that random initialization also works, while using our initialization strategy can largely promote performance. This comparison reveals that how to resort to high-resolution bands for super-resolving low-resolution bands could be of great importance in the S2 super-resolution task.

5) Weighting Parameters in Loss Function: Our loss function is constructed as a combination of the ℓ_1 loss and the SSIM loss. The weighting parameters α_1 and α_2 are set to 1 throughout all the experiments except for this part. We conduct an experiment to examine the performance of our method with different values of parameters α_1 and α_2. First, we fix $\alpha_2 = 1$ and vary the value of α_1 from 10^{-2} to 10^2. Then, α_1 is fixed as 1 and the value of α_2 varies. The SSIM values of the results are shown in Fig. 8. We can see our method is more sensitive to the value of α_2, and the best performance is obtained when $\alpha_1 = \alpha_2 = 1$.

6) Network Input Regularization Parameter: In this article, we take the same network input regularization strategy as [44], i.e., adding a zero-mean Gaussian noise with a fixed standard deviation to the input. This strategy has been shown to generate better image recovery or super-resolution results [44]. In this part, we conduct experiments with different values of the standard deviation σ. Meanwhile, we also consider parameterizing the input noise with the SNR in dB. Table VII shows the performance of our model with different standard deviations

TABLE VII
QUANTITATIVE ASSESSMENTS OF OUR METHOD WITH DIFFERENT σS OR SNRS (IN DB) ON THE SYNTHETIC DATA HYDICE-WDC AND HYDICE-TERRAIN

Data		RMSE	SSIM	SRE	UIQI	SAM
HYDICE-WDC	$\sigma = 1/10$	0.2710	0.0583	3.59	0.0139	15.7494
	$\sigma = 1/20$	0.2686	0.0672	3.66	0.0083	15.5776
	$\sigma = 1/30$	0.0099	0.9906	32.62	0.9915	1.2653
	$\sigma = 1/40$	0.0246	0.9595	27.57	0.9616	1.7962
	$\sigma = 1/50$	0.0243	0.9599	27.58	0.9614	1.7998
	SNR = 20 dB	**0.0092**	**0.9913**	**33.3**	**0.9921**	**1.2080**
	SNR = 30 dB	0.0107	0.9898	32.13	0.9907	1.3459
	SNR = 40 dB	0.0100	0.9904	32.59	0.9912	1.2887
	SNR = 50 dB	0.0104	0.9885	32.23	0.9895	1.3592
HYDICE-Terrain	$\sigma = 1/10$	0.2707	0.0624	6.40	0.0098	15.7415
	$\sigma = 1/20$	0.2711	0.0605	6.39	0.0105	15.8014
	$\sigma = 1/30$	0.0112	**0.9917**	34.40	**0.9917**	1.0370
	$\sigma = 1/40$	0.0162	0.9908	34.21	0.9902	1.0523
	$\sigma = 1/50$	0.0114	0.9907	34.07	0.9905	1.0578
	SNR = 20 dB	0.0119	0.9910	34.01	0.9911	1.1267
	SNR = 30 dB	0.0112	0.9907	34.53	0.9910	1.0211
	SNR = 40 dB	0.0152	0.9904	31.72	0.9901	1.2558
	SNR = 50 dB	**0.0110**	0.9912	**34.68**	0.9915	**1.0162**

The best values and the second-best values are highlighted by bolder fonts and underlines, respectively.

TABLE VIII
RUNNING TIME (IN SECONDS) OF ALL METHODS ON THE SYNTHETIC DATA

Data	Bicubic	SupReME	MuSA	DSen2	NSTMR	SelfS2
AVIRIS-City	0.11	4.17	289.62	1.71	556.81	1543.3
AVIRIS-Coast	0.1	4.64	301.87	1.2	457.83	1579.84
AVIRIS-Crops	0.1	3.81	295.55	1.38	554.39	1551.4
AVIRIS-Montain	0.1	4.42	287.81	1.26	459.97	1533.4
HYDICE-WDC	0.02	0.46	40.12	0.72	140.22	768.75
HYDICE-Terrain	0.13	0.43	44.64	0.69	90.56	775.44
APEX-Baden	0.06	0.46	53.6	6.22	291.37	776.73

($\sigma = [1/10, 1/20, 1/30, 1/40, 1/50]$) and SNRs ([20 dB, 30 dB, 40 dB, 50 dB]). We can see that large σs would bring about inferior results, while small σs help our method generate satisfactory results. Although using the strategy of parameterizing the noise with different SNRs can obtain the best results, we still choose to add the zero-mean Gaussian noise with a fixed standard deviation since this strategy is more stable for different datasets.

7) Running Time: In Table VIII, we report the running time (in seconds) of all the competing methods on the synthetic datasets. As a self-supervised learning method, our method learns the network parameters solely from the observed data, which takes a lot of time for the network to converge, thus being time consuming.

IV. CONCLUSION

In this article, to fulfill the super-resolution of S2 images, we suggest a novel method resorting to the deep image prior. The structure of the selected CNN could well capture the low-level statistics, and network parameters are obtained solely from the observed MSI, being adaptive to different scenes. The degradation of S2 images is taken into account via the loss function, making the network output in accord with the degradation model. Moreover, as the observed S2 images contain four high-resolution bands, a novel network parameter initialization strategy is designed to further utilize fine features within those bands. Then, after our initialization, the network parameters are self-supervisedly learned solely from the observed S2 image. Experiments are conducted on simulated and real S2 data. From the comparison with state-of-the-art methods, we can see the effectiveness of our method. However, as our method needs to infer all the parameters solely from the observed MSI in a self-supervised learning manner, it is quite time consuming. Thus, we will consider accelerating our method in the future.

ACKNOWLEDGMENT

The authors would like to thank the editor and reviewers for many comments and suggestions, which are of great value for improving the quality of this article. The authors would like to thank the authors of [11], [23], [33], [35], and [38] for their generous sharing of their codes or data.

REFERENCES

[1] M. J. Carlotto, M. B. Lazaroff, and M. W. Brennan, "Multispectral image processing for environmental monitoring," in *Proc. Digit. Image Process. Vis. Commun. Technol. Earth Atmos. Sci. II*, 1993, pp. 113–124.
[2] D. Stow, A. Hope, A. T. Nguyen, S. Phinn, and C. A. Benkelman, "Monitoring detailed land surface changes using an airborne multispectral digital camera system," *IEEE Trans. Geosci. Remote Sens.*, vol. 34, no. 5, pp. 1191–1203, Sep. 1996.
[3] T. Bhadra, A. Chouhan, D. Chutia, A. Bhowmick, and P. Raju, "Flood detection using multispectral images and SAR data," in *Proc. Int. Conf. Mach. Learn., Image Process., Netw. Secur. Data Sci.*, 2020, pp. 294–303.
[4] A. Vali, S. Comai, and M. Matteucci, "Deep learning for land use and land cover classification based on hyperspectral and multispectral earth observation data: A review," *Remote Sens.*, vol. 12, no. 15, 2020, Art. no. 2495.
[5] A. A. Nielsen, K. Conradsen, and J. J. Simpson, "Multivariate alteration detection (MAD) and MAF postprocessing in multispectral, bitemporal image data: New approaches to change detection studies," *Remote Sens. Environ.*, vol. 64, no. 1, pp. 1–19, 1998.
[6] Y. Bazi, F. Melgani, and H. D. Al-Sharari, "Unsupervised change detection in multispectral remotely sensed imagery with level set methods," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 8, pp. 3178–3187, Aug. 2010.
[7] C. Wu, B. Du, and L. Zhang, "Slow feature analysis for change detection in multispectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 5, pp. 2858–2874, May 2014.
[8] T. S. Pagano and R. M. Durham, "Moderate resolution imaging spectroradiometer (MODIS)," in *Proc. Sens. Syst. Early Earth Observing Syst. Platforms*, 1993, pp. 2–17.
[9] Y. Yamaguchi, A. B. Kahle, H. Tsu, T. Kawakami, and M. Pniel, "Overview of advanced spaceborne thermal emission and reflection radiometer (ASTER)," *IEEE Trans. Geosci. Remote Sens.*, vol. 36, no. 4, pp. 1062–1071, Jul. 1998.
[10] M. Drusch et al., "Sentinel-2: ESA's optical high-resolution mission for GMES operational services," *Remote Sens. Environ.*, vol. 120, pp. 25–36, 2012.
[11] C. Lanaras, J. Bioucas-Dias, E. Baltsavias, and K. Schindler, "Super-resolution of multispectral multiresolution images from a single sensor," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2017, pp. 20–28.
[12] Q. Wang, W. Shi, Z. Li, and P. M. Atkinson, "Fusion of Sentinel-2 images," *Remote Sens. Environ.*, vol. 187, pp. 241–252, 2016.
[13] B. Aiazzi, S. Baronti, and M. Selva, "Improving component substitution pansharpening through multivariate regression of MS + Pan data," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 10, pp. 3230–3239, Oct. 2007.
[14] B. Aiazzi, S. Baronti, F. Lotti, and M. Selva, "A comparison between global and context-adaptive pansharpening of multispectral images," *IEEE Geosci. Remote Sens. Lett.*, vol. 6, no. 2, pp. 302–306, Apr. 2009.
[15] J. Choi, K. Yu, and Y. Kim, "A new adaptive component-substitution-based satellite image fusion by using partial replacement," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 1, pp. 295–309, Jan. 2011.

[16] T. Ranchin and L. Wald, "Fusion of high spatial and spectral resolution images: The ARSIS concept and its implementation," *Photogrammetric Eng. Remote Sens.*, vol. 66, no. 1, pp. 49–61, 2000.

[17] B. Aiazzi, L. Alparone, S. Baronti, A. Garzelli, and M. Selva, "MTF-tailored multiscale fusion of high-resolution MS and PAN imagery," *Photogrammetric Eng. Remote Sens.*, vol. 72, no. 5, pp. 591–596, 2006.

[18] L. Alparone, L. Wald, J. Chanussot, C. Thomas, P. Gamba, and L. M. Bruce, "Comparison of pansharpening algorithms: Outcome of the 2006 GRS-S data-fusion contest," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 10, pp. 3012–3021, Oct. 2007.

[19] B. Aiazzi, L. Alparone, S. Baronti, A. Garzelli, and M. Selva, "An MTF-based spectral distortion minimizing model for pan-sharpening of very high resolution multispectral images of urban areas," in *Proc. 2nd GRSS/ISPRS Joint Workshop Remote Sens. Data Fusion Over Urban Areas*, 2003, pp. 90–94.

[20] G. Vivone et al., "A critical comparison among pansharpening algorithms," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2565–2586, May 2015.

[21] Q. Wang, W. Shi, P. M. Atkinson, and E. Pardo-Igúzquiza, "A new geostatistical solution to remote sensing image downscaling," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 386–396, Jan. 2015.

[22] Q. Wang, W. Shi, P. M. Atkinson, and Y. Zhao, "Downscaling MODIS images with area-to-point regression kriging," *Remote Sens. Environ.*, vol. 166, pp. 191–204, 2015.

[23] C.-H. Lin and J. M. Bioucas-Dias, "An explicit and scene-adapted definition of convex self-similarity prior with application to unsupervised Sentinel-2 super-resolution," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3352–3365, May 2020.

[24] L. Loncan et al., "Hyperspectral pansharpening: A review," *IEEE Trans. Geosci. Remote Sens.*, vol. 3, no. 3, pp. 27–46, Sep. 2015.

[25] X. Fu, W. Wang, Y. Huang, X. Ding, and J. Paisley, "Deep multiscale detail networks for multiband spectral image sharpening," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 5, pp. 2090–2104, May 2021.

[26] J. Yang, X. Fu, Y. Hu, Y. Huang, X. Ding, and J. Paisley, "PanNet: A deep network architecture for pan-sharpening," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5449–5457.

[27] L.-J. Deng, M. Feng, and X.-C. Tai, "The fusion of panchromatic and multispectral remote sensing images via tensor-based sparse modeling and hyper-Laplacian prior," *Inf. Fusion*, vol. 52, pp. 76–89, 2019.

[28] L.-J. Deng, G. Vivone, C. Jin, and J. Chanussot, "Detail injection-based deep convolutional neural networks for pansharpening," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 8, pp. 6995–7010, Aug. 2021.

[29] J.-F. Hu, T.-Z. Huang, L.-J. Deng, T.-X. Jiang, G. Vivone, and J. Chanussot, "Hyperspectral image super-resolution via deep spatiospectral attention convolutional neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, doi: 10.1109/TNNLS.2021.3084682.

[30] S. Li, R. Dian, L. Fang, and J. M. Bioucas-Dias, "Fusing hyperspectral and multispectral images via coupled sparse tensor factorization," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 4118–4130, Aug. 2018.

[31] R. Dian, S. Li, B. Sun, and A. Guo, "Recent advances and new guidelines on hyperspectral and multispectral image fusion," *Inf. Fusion*, vol. 69, pp. 40–51, 2021.

[32] S. Clerc and MPC Team, "S2 MPC—Data quality report," 2017. Accessed: Feb. 6, 2017. [Online]. Available: http://earth.esa.int/documents/247904/685211/Sentinel-2-Data-Quality-Report

[33] X.-Q. Wang and T.-Y. Ji, "NSTMR: Super resolution of Sentinel-2 images using nonlocal nonconvex surrogate of tensor multirank," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 5694–5706, 2021.

[34] M. Bertero and P. Boccacci, *Introduction to Inverse Problems in Imaging*. Boca Raton, FL, USA: CRC Press, 2020.

[35] C. Paris, J. Bioucas-Dias, and L. Bruzzone, "A novel sharpening approach for superresolving multiresolution optical images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 3, pp. 1545–1560, Mar. 2018.

[36] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space," in *Proc. IEEE Int. Conf. Image Process.*, 2007, pp. I–313.

[37] M. O. Ulfarsson, F. Palsson, M. D. Mura, and J. R. Sveinsson, "Sentinel-2 sharpening using a reduced-rank method," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6408–6420, Sep. 2019.

[38] C. Lanaras, J. Bioucas-Dias, S. Galliani, E. Baltsavias, and K. Schindler, "Super-resolution of Sentinel-2 images: Learning a globally applicable deep neural network," *ISPRS J. Photogrammetry Remote Sens.*, vol. 146, pp. 305–319, 2018.

[39] F. Palsson, J. R. Sveinsson, and M. O. Ulfarsson, "Sentinel-2 image fusion using a deep residual network," *Remote Sens.*, vol. 10, no. 8, 2018, Art. no. 1290.

[40] F. Palsson, J. R. Sveinsson, and M. O. Ulfarsson, "Single sensor image fusion using a deep convolutional generative adversarial network," in *Proc. 9th Workshop Hyperspectral Image Signal Process.: Evol. Remote Sens.*, 2018, pp. 1–5.

[41] M. Gargiulo, A. Mazza, R. Gaetano, G. Ruello, and G. Scarpa, "Fast super-resolution of 20 m Sentinel-2 bands using convolutional neural networks," *Remote Sens.*, vol. 11, no. 22, 2019, Art. no. 2635.

[42] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[43] I. Goodfellow et al., "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[44] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," *Int. J. Comput. Vis.* vol. 30, pp. 1867–1888, 2020.

[45] U. Muller-Wilm, J. Louis, R. Richter, F. Gascon, and M. Niezette, "Sentinel-2 level 2A prototype processor: Architecture, algorithms and first results," in *Proc. ESA Living Planet Symp.*, 2013, pp. 9–13.

[46] Y.-S. Luo, X.-L. Zhao, T.-X. Jiang, Y.-B. Zheng, and Y. Chang, "Hyperspectral mixed noise removal via spatial-spectral constrained unsupervised deep image prior," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 9435–9449, 2021.

[47] B. Aiazzi, S. Baronti, M. Selva, and L. Alparone, "Bi-cubic interpolation for shift-free pan-sharpening," *ISPRS J. Photogrammetry Remote Sens.*, vol. 86, pp. 65–76, 2013.

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

[49] Y. LeCun et al., "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989.

[50] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervention*, 2015, pp. 234–241.

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

[52] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. Int. Conf. Mach. Learn.*, 2013, Art. no. 3.

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

[54] T. Huang, W. Dong, X. Yuan, J. Wu, and G. Shi, "Deep Gaussian scale mixture prior for spectral compressive imaging," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 16216–16225.

[55] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[56] M. Selva, B. Aiazzi, F. Butera, L. Chiarantini, and S. Baronti, "Hyper-sharpening: A first approach on SIM-GA data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 3008–3024, Jun. 2015.

[57] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[58] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2019, vol. 32, pp. 8026–8037.

[59] G. Vane, R. O. Green, T. G. Chrien, H. T. Enmark, E. G. Hansen, and W. M. Porter, "The airborne visible/infrared imaging spectrometer (AVIRIS)," *Remote Sens. Environ.*, vol. 44, no. 2/3, pp. 127–143, 1993.

[60] M. E. Schaepman et al., "Advanced radiometry measurements and earth science applications with the airborne prism experiment (APEX)," *Remote Sens. Environ.*, vol. 158, pp. 207–219, 2015.

[61] Z. Wang and A. Bovik, "A universal image quality index," *IEEE Signal Process. Lett.*, vol. 9, no. 3, pp. 81–84, Mar. 2002.

[62] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[63] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 694–711.