Optimization Algorithm Unfolding Deep Networks of Detail Injection Model for Pansharpening

Yunqiao Feng, Junmin Liu, Kun Chen, Bo Wang, and Zixiang Zhao

***Abstract*— *Pansharpening* aims at integrating a high-spatial-resolution panchromatic (PAN) image with a low-spatial-resolution multispectral (MS) image to generate a high-resolution MS (HRMS) image. It is a fundamental and significant task in the field of remotely sensed images. Classic and convolutional neural network (CNN)-based algorithms have been developed, over the last decades, for pansharpening based on the spatial detail injection model. However, these algorithms have difficulties in extracting sufficient details or lack interpretability. In this letter, we present an algorithm unfolding pansharpening (AUP) for this task. In the proposed AUP, a two-step optimization model is first designed based on the spatial detail decomposition model. Then, the iteration processes induced by an optimization model are mapped to several *detailed convolution* (dc) blocks to solve the detail injection by a trainable neural network. Finally, the desired MS details are obtained in end-to-end manners through a decoder. The superiority of the proposed AUP is demonstrated by extensive experiments on datasets acquired by two different kinds of satellites. Each module of the AUP is interpretable, and its fused results are with fewer spectral and spatial distortions.**

***Index Terms*— Algorithm unfolding, deep neural networks, details injection, pansharpening.**

I. Introduction

DUE to the severe constraint about the signal-to-noise ratio of satellite products [1], it becomes a hard task for current satellites to obtain the high-spatial-resolution multispectral (MS) images; generally, only an image pair is acquired with complementary features, i.e., a high-spatial-resolution panchromatic (PAN) image and a low-spatial-resolution MS image with rich spectral information. Nevertheless, high-spatial-resolution MS images are required in many applications, such as the *visual interpretation*, the *land cover change monitor*, and *object recognition*. A way to get high-quality products through signal processing is classically referred to as pansharpening, which can be cast as a typical kind of *image fusion* [2].

Over the past decades, various pansharpening methods have been proposed in the literature of remote sensing. The classical two categories [3] include component substitution (CS)-based methods and multiresolution analysis (MRA)-based methods. They hold the same objective to extract MS details from the PAN and MS image as much as possible while preventing spectral distortion. The main differences between the two categories are in extracting spatial details from the PAN image, turning the spatial details to MS details, and then injecting them into the low-resolution MS (LRMS) image.

There has been recently significant progress in convolutional neural network (CNN)-based methods in reducing the spatial and spectral distortions of the fused images compared to classical methods. The reason may be the end-to-end learning strategy and their powerful feature extraction capacity. For instance, a CNN architecture called pansharpening neural network (PNN) was designed and trained by Masi *et al.* [4] with three layers. Zhou *et al.* [5] proposed pyramid fully convolutional network for the pansharpening. Although CNN-based methods have represented state-of-the-art (SOTA) performance, they have no interpretability and are more like a black-box game compared to the classic methods.

Recently, a novel detail injection pansharpening framework based on CNN was proposed to address the limitations of existing CNN-based methods by combining the classic pansharpening with CNN. For example, Yang *et al.* [6] first introduced detail injection neural networks taking both PAN and MS images to extract spatial details. Deng *et al.* [7] combined CNNs and traditional CS and MRA fusion schemes to estimate the nonlinear injection models. The detail injection CNN models have better spectral quality compared to the general CNN-based methods and greatly reduce the uncertainty of fusion results. Unfortunately, no such studies considered designing an interpretable network, and only a general convolution structure was exploited.

With the development of research on the interpretability of deep learning (DL), a novel method was presented called *algorithm unfolding* to build interpretable deep networks. For example, the fast sparse coding was proposed by Gregor and LeCun [8] to train a nonlinear, feedforward predictor with a specific architecture and a fixed depth. The main idea of algorithm unfolding is to extend the traditional sparse coding problem into deep networks and solve it iteratively

Manuscript received January 7, 2021; revised March 21, 2021 and April 25, 2021; accepted April 26, 2021. This work was supported in part by the National Key Research and Development Program of China under Grant 2020AAA0105601, in part by the National Natural Science Foundation of China under Grant 61877049 and Grant 51709065, in part by the Natural Science Foundation of Heilongjiang Province under Grant JJ2020LH1535, and in part by the Foundation under Grant 61403120105. *(Corresponding authors: Junmin Liu; Bo Wang.)*

Yunqiao Feng, Junmin Liu, Kun Chen, and Zixiang Zhao are with the School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: junminliu@mail.xjtu.edu.cn).

Bo Wang is with the National Key Laboratory of Science and Technology on Underwater Vehicle, Harbin Engineering University, Harbin 150001, China (e-mail: wb@hrbeu.edu.cn).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/LGRS.2021.3077183.

Color versions of one or more figures in this letter are available at https://doi.org/10.1109/LGRS.2021.3077183.

Digital Object Identifier 10.1109/LGRS.2021.3077183

to increment the interpretability of the model while reducing the calculation burden and a large number of parameters.

It is difficult for the current pansharpening algorithms, especially the classic detail injection framework, to extract sufficient spatial details through simple filters or manually designed network architectures. In addition, as for the general CNN-based frameworks, they are more like a black-box game and lack interpretability. Although the uncertainty of learning is reduced by the detail injection-based CNN [6], [7], [14], it is not clear how to extract the detailed information by deep networks with interpretability. In this letter, a decomposition model is first established based on the mechanism of MS detail extraction, and then, the model is formulated into an optimization problem. Note that decomposition here is to decompose detailed information from PAN images. Through the principle of algorithm unfolding, we extend the optimization-based decomposition algorithm into an interpretable detail injection CNN, called algorithm unfolding pansharpening (AUP). Experiments conducted on WorldView2 and QuickBird datasets demonstrate that the proposed AUP method can acquire pleasant fusion results at both full and reduced scales.

II. Related Work

Notations: The LRMS image is denoted by $\boldsymbol{\mathcal{M}} \in \mathbb{R}^{H\times W\times C}$, where H, W, and C are the height, the width, and the number of spectral bands of the LRMS image, respectively. We denote the high-resolution PAN image by $\mathbf{P} \in \mathbb{R}^{rH\times rW}$, in which r represents the spatial resolution ratio between MS and PAN, denoted by $\widehat{\boldsymbol{\mathcal{M}}} \in \mathbb{R}^{rH\times rW\times C}$ the reconstructed high-resolution MS (HRMS) image. Let $\mathbf{M}_k$ denote the kth band of the LRMS image and $\widetilde{\mathbf{M}}_k \in \mathbb{R}^{rH\times rW}$ denote the upsampled version of $\mathbf{M}_k$ by ratio r. Based on these symbols, we briefly introduce the main idea of detail injection and CNN frameworks in pansharpening.

A. Detail Injection Framework

During the last decades, numerous research efforts have been devoted to developing pansharpening algorithms. Classic methods belong to CS- and MRA-based classes. Generally, the CS class has the following form:

$$\widehat{\mathbf{M}}_k = \widetilde{\mathbf{M}}_k + g_k(\mathbf{P} - \mathbf{I}_L), \quad k = 1, \ldots, C \tag{1}$$

where $g_1, \ldots, g_C$ denote the injection gains, and $\mathbf{I}_L$ is a linear combination of the upsampled LRMS image bands. It is often called an *intensity component*, which is determined as $\mathbf{I}_L = \sum_{k=1}^{C} w_k \widetilde{\mathbf{M}}_k$, where $w_1, \ldots, w_C$ usually correspond to the first row of the forward transformation matrix for measuring the degrees of spectral overlap between the MS and PAN channels.

For the MRA class, it can be formulated as

$$\widehat{\mathbf{M}}_k = \widetilde{\mathbf{M}}_k + g_k(\mathbf{P} - \mathbf{P}_L) \tag{2}$$

where $\mathbf{P}_L$ represents the low-frequency component of the PAN image.

Based on the fusion process shown in (1) and (2), traditional CS-based methods and MRA-based methods can be mathematically classified as the *detail injection framework*. Normally, they both include two following steps.

1) Generating the spatial details extracted from PAN and/or MS image and then calculating the injection gains to obtain the MS details.
2) Injecting the MS details into an upsampled MS image to produce the desired HRMS image. Thus, (1) and (2) can be equivalently created as

$$\widehat{\mathbf{M}}_k = \widetilde{\mathbf{M}}_k + \mathbf{D}_k \tag{3}$$

where $\mathbf{D}_k$ represents spatial details created by both PAN and MS image based on each class and is used for injecting into the upsampled MS image $\widetilde{\mathbf{M}}_k$.

B. CNN Framework

Recently, the CNN-based approaches have been utilized for pansharpening and obtaining the SOTA performances [4]. Since pansharpening is different from image super-resolution and other image fusion problems, detail injection CNN-based methods for pansharpening were proposed to further improve the performances of CNN-based methods [6], [7], [14]. The network is different from the hand-crafted designed MS details, while detail injection CNN-based methods try to learn how to extract MS details through the neural network within an adaptive way. Compared to the classic methods and general CNN-based methods, detail injection-based CNN methods have clear interpretability in the detail injection context and are greatly able to reduce the uncertainty of learning, which is similar to the mechanism of residual learning.

As the ideal HRMS image is not available, a common trick of the CNN-based methods is the scale-invariant assumption based on Wald's protocol [9]. Hence, the training image pairs are generated by downsampling both the PAN and MS images with ratio r by using the modulation transfer functions (MTF)-matched low-pass filters [10].

Generally, let the ideal HRMS image be $\boldsymbol{\mathcal{Y}}^i$; CNN-based methods can be obtained by minimizing the following loss function:

$$\mathcal{L}(\boldsymbol{\Theta}) = \sum_{i=1}^{n} \left\| \boldsymbol{\mathcal{Y}}^i - f\left(\boldsymbol{\mathcal{X}}^i; \boldsymbol{\Theta}\right) - \widetilde{\boldsymbol{\mathcal{M}}}^i \right\|_F^2 \tag{4}$$

where $f(\boldsymbol{\mathcal{X}}^i; \boldsymbol{\Theta})$ represents the reconstructed MS details taking $\boldsymbol{\Theta}$ as parameters and with the input $\boldsymbol{\mathcal{X}}^i$ (PAN and MS image pairs or just PAN image). Here, n denotes the number of training samples, and $||\cdot||_F$ is the Frobenius norm.

III. Proposed AUP Method

To avoid the incomplete extraction of detailed information in traditional methods and the poor interpretability of CNN-based methods, we first introduce an image decomposition model to extract MS details. Then, we formulate the model into an optimization problem and ultimately solve it by unfolding an iteration process into deep neural networks.

A. Optimization Algorithm

For an observed PAN image, the spatial details can be represented as $\boldsymbol{\mathcal{D}} \in \mathbb{R}^{rH\times rW\times C}$, in which $\boldsymbol{\mathcal{D}}$ is a tensor composed by $\mathbf{D}_1, \ldots, \mathbf{D}_C$. Here, to extract different spatial information for each channel of MS image, we use the tensor form of spatial

detail. The classic- and CNN-based methods are mainly aim to estimate the details from $\mathbf{P}$ and $\boldsymbol{\mathcal{M}}$ (expressed by linear/filter transformation for classic-based methods and a deep network for CNN-based approaches).

Contrary to classically making spatial details with low capacity in separating the high- and low-frequency information or heuristically constructing a complex deep network architecture, we try to discover a way to take the advantages of both the high feature extraction capacity of CNN-based methods and the high interpretability of classic-based methods. Therefore, we consider the detailed injection formulation of MRA class. Since the methodology of information separation of MRA is interpretable, it can be simply integrated with CNN framework. Hence, based on the formulation (2), spatial detail can be obtained as

$$\mathbf{D}_k = \mathbf{P} - \mathbf{P} * f_{\text{low}}^k \tag{5}$$

where $\mathbf{D}_k$ represents the kth channel of $\boldsymbol{\mathcal{D}}$, f_{low}^k is the low-pass filter able to separate low-frequency information from PAN image, and $*$ represents the 2-D convolutional operation. For convenience, we rewrite (5) as

$$\boldsymbol{\mathcal{D}} = \boldsymbol{\mathcal{P}} - \boldsymbol{\mathcal{P}} * \mathcal{F}_{\text{low}} \tag{6}$$

where $\mathcal{F}_{\text{low}}$ represents the tensor form of $f_{\text{low}}^k, k = 1, 2, \ldots, C$, and $\boldsymbol{\mathcal{P}} \in \mathbb{R}^{rH \times rW \times C}$ denotes the tensor form of $\mathbf{P}$, indicating that each channel of $\boldsymbol{\mathcal{P}}$ equals to $\mathbf{P}$. We perform convolution between $\boldsymbol{\mathcal{P}}$ and $\mathcal{F}_{\text{low}}$ for each channel.

Moreover, based on (3), once we obtain spatial details, it is necessary to create MS details by injection gains. Contrary to classic approaches, here, we suppose that MS details can be obtained by a function of spatial details, i.e., the MS details are represented as $\boldsymbol{\mathcal{D}}_M = g(\boldsymbol{\mathcal{D}})$, where g is an unknown function; thus, we can rewrite (3) as

$$\widehat{\boldsymbol{\mathcal{M}}} = \widetilde{\boldsymbol{\mathcal{M}}} + g(\boldsymbol{\mathcal{D}}). \tag{7}$$

Since our objective is to discover the MS details, with the defined detail reconstruction model, we should formulate the energy function. According to maximum *a posteriori* (MAP) probability, the energy function can be obtained as

$$\boldsymbol{\mathcal{L}} = \min_{\boldsymbol{\mathcal{D}}, \widehat{\boldsymbol{\mathcal{M}}}} \|\boldsymbol{\mathcal{P}} - \boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{P}} * \mathcal{F}_{\text{low}}\|_F^2 + \lambda \|\widehat{\boldsymbol{\mathcal{M}}} - g(\boldsymbol{\mathcal{D}}) - \widetilde{\boldsymbol{\mathcal{M}}}\|_F^2 \tag{8}$$

where $\|\boldsymbol{\mathcal{P}} - \boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{P}} * \mathcal{F}_{\text{low}}\|_F^2$ represents the data fidelity term created by the reconstruction model. $\|\widehat{\boldsymbol{\mathcal{M}}} - g(\boldsymbol{\mathcal{D}}) - \widetilde{\boldsymbol{\mathcal{M}}}\|_F^2$ denotes the regularization (prior) term, and λ represents the regularization parameter.

For solving (8), traditional techniques normally contain definite complex operations, i.e., the Fourier transform and the inverse Fourier transform [11] that are difficult to accomplish. Since we try to unfold these equations in deep network for solving the image decomposition model step-by-step, it is essential to create an algorithm containing only simple computations with simple transformation to network modules. Therefore, we prefer to utilize gradient descent algorithm for solving the above problems. As there are two target variables in (8), the gradient calculation of (8) should be separated into two steps, one of which is for spatial details

$$\frac{\partial \boldsymbol{\mathcal{L}}}{\partial \boldsymbol{\mathcal{D}}} = (\boldsymbol{\mathcal{P}} - \boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{P}} * \mathcal{F}_{\text{low}}) + \lambda g'(\boldsymbol{\mathcal{D}})(\widehat{\boldsymbol{\mathcal{M}}} - g(\boldsymbol{\mathcal{D}}) - \widetilde{\boldsymbol{\mathcal{M}}}) \tag{9}$$

and the other one is for $\widehat{\boldsymbol{\mathcal{M}}}$

$$\frac{\partial \boldsymbol{\mathcal{L}}}{\partial \widehat{\boldsymbol{\mathcal{M}}}} = \widehat{\boldsymbol{\mathcal{M}}} - g(\boldsymbol{\mathcal{D}}) - \widetilde{\boldsymbol{\mathcal{M}}}. \tag{10}$$

Thus, the update rule of gradient descent can be divided into the following processes:

$$\begin{aligned} \boldsymbol{\mathcal{D}}_{(t)} = \boldsymbol{\mathcal{D}}_{(t-1)} - \eta\big[&\big(\boldsymbol{\mathcal{P}} - \boldsymbol{\mathcal{D}}_{(t-1)} - \boldsymbol{\mathcal{P}} * \mathcal{F}_{\text{low}}\big) \\ &+ \lambda g'\big(\boldsymbol{\mathcal{D}}_{(t-1)}\big)\big(\widehat{\boldsymbol{\mathcal{M}}}_{(t-1)} - g\big(\boldsymbol{\mathcal{D}}_{(t-1)}\big) - \widetilde{\boldsymbol{\mathcal{M}}}\big)\big] \end{aligned} \tag{11}$$

$$\widehat{\boldsymbol{\mathcal{M}}}_{(t)} = \widehat{\boldsymbol{\mathcal{M}}}_{(t-1)} - \eta(\widehat{\boldsymbol{\mathcal{M}}}_{(t-1)} - g(\boldsymbol{\mathcal{D}}_{(t)}) - \widetilde{\boldsymbol{\mathcal{M}}}) \tag{12}$$

where η represents the step size. It is worth noting that, at the t stage for the two step optimization, first, the spatial details $\boldsymbol{\mathcal{D}}_{(t-1)}$ are updated; then, $\boldsymbol{\mathcal{D}}_{(t)}$ is obtained and used to update $\widehat{\boldsymbol{\mathcal{M}}}_{(t-1)}$

B. Algorithm Unfolding

Inspired by the work [12], we try to unfold the optimization problem in a CNN. Indeed, (11) and (12) comprise filters $\mathcal{F}_{\text{low}}$, and unknown functions $g(\cdot)$ and $g'(\cdot)$ possess the same function with the convolution operation in CNN. Therefore, filters $\mathcal{F}_{\text{low}}$ and the unknown function $g(\cdot)$ are substituted by convolutional units, and (11) and (12) can be rewritten as

$$\text{D-term} : \begin{cases} \boldsymbol{\mathcal{E}}_{(t)} = \boldsymbol{\mathcal{P}} - \boldsymbol{\mathcal{D}}_{(t-1)} - \text{Conv}_l(\boldsymbol{\mathcal{P}}) \\ \widehat{\boldsymbol{\mathcal{R}}}_{(t)} = \widehat{\boldsymbol{\mathcal{M}}}_{(t-1)} - \text{Net}_1\big(\boldsymbol{\mathcal{D}}_{(t-1)}\big) - \widetilde{\boldsymbol{\mathcal{M}}} \\ \boldsymbol{\mathcal{H}}_{(t)} = \boldsymbol{\mathcal{E}}_{(t)} + \text{Net}_2(\boldsymbol{\mathcal{D}}_{(t-1)})\widehat{\boldsymbol{\mathcal{R}}}_{(t)} \\ \boldsymbol{\mathcal{D}}_{(t)} = \boldsymbol{\mathcal{D}}_{(t-1)} - \eta\boldsymbol{\mathcal{H}}_{(t)} \end{cases} \tag{13}$$

$$\text{R-term} : \begin{cases} \boldsymbol{\mathcal{R}}_{(t)} = \widehat{\boldsymbol{\mathcal{M}}}_{(t-1)} - \text{Net}_1\big(\boldsymbol{\mathcal{D}}_{(t)}\big) - \widetilde{\boldsymbol{\mathcal{M}}} \\ \widehat{\boldsymbol{\mathcal{M}}}_{(t)} = \widehat{\boldsymbol{\mathcal{M}}}_{(t-1)} - \eta\boldsymbol{\mathcal{R}}_{(t)} \end{cases} \tag{14}$$

where Conv_l, $\text{Net}_1(\cdot)$, and $\text{Net}_2(\cdot)$ represent the convolutional unit with a kernel of size k. In this letter, k is adjusted to 3.

Equations (13) and (14) can be considered as a detail convolution block (DC block), including two parts: one for updating the data term and the other one for updating the regular term. Three convolution units exist in each DC block; a batch normalization is conducted with parametric rectified linear unit (PReLU) as activation function after the convolution of Conv_l, $\text{Net}_1(\cdot)$, and $\text{Net}_2(\cdot)$, to avoid the vanishing gradient problem. Thus, feature extraction capability can be further enhanced. Note that Conv_l, $\text{Net}_1(\cdot)$, and $\text{Net}_2(\cdot)$ are general 2-D convolution layers, where the 3×3 convolution kernel is generated by random initialization (see the Supplementary Material for more details about their network architectures). In contrast to traditional algorithms predefining the hyperparameters, λ, step size η, the filter $\mathcal{F}_{\text{low}}$, and unknown functions $g(\cdot)$ and $g'(\cdot)$, here, we utilize the neural network to make them learnable for extracting MS details in an adaptive way.

It is worth noting that each term in the DC block is very interpretable. For the data term, first, the residual information $\boldsymbol{\mathcal{E}}_{(t)}$ and $\widehat{\boldsymbol{\mathcal{R}}}_{(t)}$ of spatial details and HRMS image is,

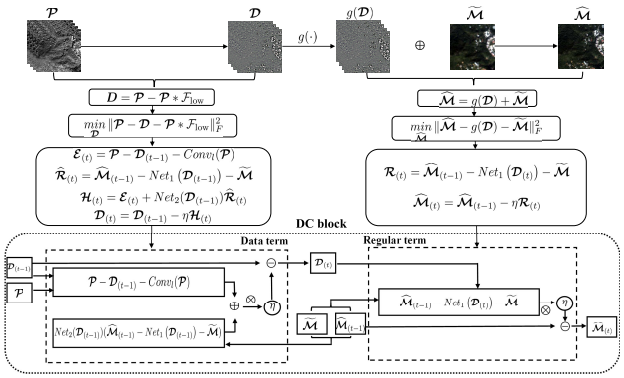

Fig. 1. Illustration of a single DC block in the AUP network.

respectively, estimated by (6) and (7). Then, the gradient information $\mathcal{H}_{(t)}$ of $\mathcal{D}_{(t-1)}$ is determined with the two residual information and utilized for updating the spatial details. During the regular term, the updated $\mathcal{D}_{(t)}$ was considered as the input for calculating the new residual information $\mathcal{R}_{(t)}$ and updating the estimated HRMS image $\widehat{\mathcal{M}}_{(t-1)}$.

C. Network Architecture

The proposed AUP can be split into encoder and decoder parts. In the encoder part, we stack N DC blocks for PAN images to iteratively decompose and update spatial feature maps. N is set to 9 in this letter. The output of the encoder is the last spatial feature map, which is taken as the input of the decoder after convolved with the convolution unit $\mathrm{Net}_1(\cdot)$. The decoder part is used to rebuild the desired MS details $\mathcal{D}_M$. Finally, the estimated HRMS image is created by (3). It should be noted that, since the MS details are acquired by unfolding neural network with automatic adjustment capability, thus, g_k is set to 1, here. Due to the limited space, the process of AUP for creating the MS details and the decoder part are displayed in the Supplementary Material.

The detailed architecture of a single DC block is presented in Fig. 1. The DC block in encoder takes $\mathcal{P}$, $\mathcal{M}$, and $\widetilde{\mathcal{M}}$ as input and then creates the initial spatial details $\mathcal{D}_{(0)}$ and $\widehat{\mathcal{M}}_{(0)}$ by employing Laplace filters to $\mathcal{P}$ and bicubic interpolation to $\mathcal{M}$, respectively.

The numbers of channels $\left(c^{\text{input}}, c^{\text{output}}\right)$ for the convolution units Conv_l, $\mathrm{Net}_1(\cdot)$, and $\mathrm{Net}_2(\cdot)$ are (C, O), (C, C), and (C, C). It is worth noting that, to match the number of channels of the image before and after Conv_l, an additional convolutional layer [the number of channels $\left(c^{\text{input}}, c^{\text{output}}\right)$ as (O, C)] is designed after Conv_l. Here, we set O as 64.

To learn the model parameters, the reconstruction errors are determined as

$$\mathcal{L} = \sum_{i=1}^{n} \alpha ||\mathcal{D}^i + \widetilde{\mathcal{M}}^i - \mathcal{Y}^i||_F^2 + \beta ||\widehat{\mathcal{M}}^i - \mathcal{Y}^i||_F^2 \quad (15)$$

where $\mathcal{D}^i$ represents the MS detail of the ith train image, $\widehat{\mathcal{M}}^i$ is the predicted HRMS image in the last DC block of the ith train image, and α and β are hyperparameters, which are, respectively, set to 0.9 and 0.1 in the experiments. We used ℓ_2 loss to measure the pixel intensity between the reconstructed image and its corresponding ideal one.

TABLE I
QUANTITATIVE COMPARISON FOR THE QUICKBIRD DATASET AT THE REDUCED SCALE

method	DRPNN	FusionNet	PanNet	DiCNN2	AUP
Q_avg(1)	0.9131	0.9211	0.9320	0.9364	**0.9404**
SAM(0)	4.8361	4.5683	4.1936	4.1058	**4.0085**
ERGAS(0)	3.9849	3.6260	3.2746	3.3183	**3.1391**
SCC(1)	0.8974	0.9046	0.9210	0.9220	**0.9274**
Q(1)	0.9131	0.9205	0.9333	0.9361	**0.9396**

IV. EXPERIMENTS

By applying the proposed AUP algorithm to two remote sensing image datasets, we compare it with some state-of-art methods, including deep residual pansharpening neural network (DRPNN) [13], FusionNet [7], the PanNet [6], and the detail injection-based CNN mode 2 method, DiCNN2 [14]. The experiments are performed on a computer with an NVIDIA GeForce RTX 2080ti GPU and an Intel i7-9700K CPU at 3.60 GHz.

We examined the proposed AUP and compared it with baselines on two image datasets gathered from QuickBird and WorldView2, respectively. To avoid overfitting, we divide each dataset into two nonoverlapping subsets, i.e., a test dataset and a training dataset. There are 224 and 32 image pairs for WorldView2 and 529 and 60 image pairs for QuickBird datasets in training and test datasets, respectively. The size of each sample is 64 $\times$ 64 for the MS image, including four spectral bands (blue, green, red, and near-infrared bands) and 256 $\times$ 256 for the PAN image.

A. Reduced Scale Experiments

Following Wald's protocol [9], we first degrade PAN and MS images in the available datasets to a reduced scale and apply fusion processes to the degraded PAN and MS images. Hence, the original MS image can be considered as an ideal HRMS image for assessing the performance of fusion methods. Five extensively used indicators are taken to quantitatively assess the performance of the presented and compared approaches that are universal image quality index [15] averaged over the bands (Q_avg) and its four-band extension, Q4 [3], spectral angle mapper (SAM) [16], Erreur Relative Globale Adimensionnelle de Synthése (ERAGS) [17], and the spatial correlation coefficient (sCC) [9].

Table I represents the values of the five indexes of the baselines and the proposed AUP for the QuickBird dataset. We highlight the best results in bold black. Fig. 2 displays a typical test image selected from QuickBird datasets. According to Table I, it is observed that the proposed AUP obtains the best performance in terms of all the five indexes on the QuickBird dataset. From Fig. 2, it is indicated that the proposed AUP can obtain the fused image with less significant spectral distortions or noticeable artifacts and looks more similar to the ideal HRMS image.

B. Full-Scale Experiments

In this experiment, to generate HRMS images, the original PAN and MS images are inserted into the models. Thus, the simulation procedures are avoided, which are required

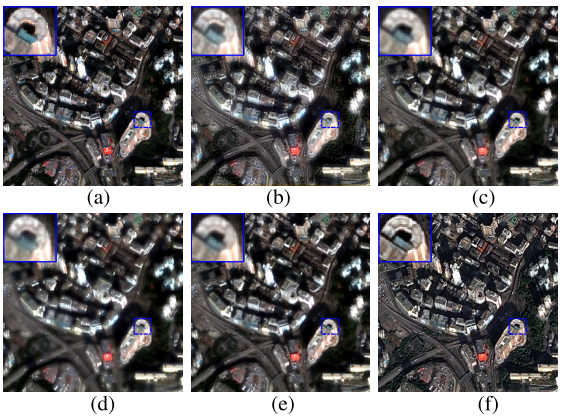

Fig. 2. Visual comparison at reduced scale on the QuickBird image. (a) Original. (b) DRPNN. (c) FusionNet. (d) PanNet. (e) DiCNN2. (f) AUP.

TABLE II

QUANTITATIVE COMPARISON FOR THE WORLDVIEW2 DATASET AT THE FULL SCALE

method	DRPNN	FusionNet	PanNet	DiCNN2	AUP
D_λ	0.0541	0.1087	0.0249	0.0365	**0.0207**
D_S	0.0827	0.2166	0.0530	0.1426	**0.0317**
QNRI	0.8677	0.6982	0.9234	0.8261	**0.9482**

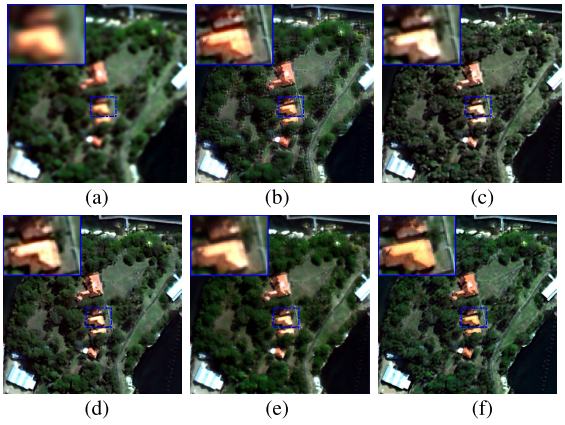

Fig. 3. Visual comparison at the full scale on the WorldView2 image. (a) LRMS. (b) DRPNN. (c) FusionNet. (d) PanNet. (e) DiCNN2. (f) AUP.

by the reduced scale experiments possibly leading to bias. During the quantitative and qualitative assessments, no reference image exists for direct comparison. For the full-scale experiments, the quality with no reference (QNR) metric [18] and its spatial index (D_S) and spectral index (D_λ) is used for measuring the quality of the fused product.

Table II represents the values of D_S, D_λ, and QNR for quantitative comparison, in which the best results are highlighted in bold black. We can see from the table that the proposed AUP shows the best performances. Fig. 3 represents the scenes of full-scale pansharpened results in the WorldView2 dataset. Based on the results generated by DRPNN, a bad performance is observed on the WorldView2 dataset. The reason is that the continuous forest area is discretized owing to not recovering structured spatial details, while, for FusionNet, PanNet, and DiCNN2, the spectral distortion is obvious. The proposed AUP owned the best performance as it can obtain the missing high-frequency contents by adaptively decomposing the PAN image through neural networks and reduce spectral distortions to a certain extent.

V. CONCLUSION

In this letter, we presented an AUP method network for fusing PAN and MS images. It was aimed to improve the interpretability of detail injection CNN-based technique in a convenient way. By the proposed AUP, the iteration steps of the optimization model are extended into a deep neural network to solve the optimization model. Thus, the interpretability of detailed information extraction can be increased while reducing calculations. The experiments on two datasets demonstrate that the proposed method can create pansharpened images with promising spectral and spatial qualities.

REFERENCES

[1] G. A. Shaw and H.-H. K. Burke, "Spectral imaging for remote sensing," *Lincoln Lab. J.*, vol. 14, no. 1, pp. 3–28, 2003.

[2] M. Ehlers, S. Klonus, P. J. Åstrand, and P. Rosso, "Multi-sensor image fusion for pansharpening in remote sensing," *Int. J. Image Data Fusion*, vol. 1, no. 1, pp. 25–45, Mar. 2010.

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

[4] G. Masi, D. Cozzolino, L. Verdoliva, and G. Scarpa, "Pansharpening by convolutional neural networks," *Remote Sens.*, vol. 8, no. 7, p. 594, Jul. 2016.

[5] F. Zhou, R. Hang, Q. Liu, and X. Yuan, "Pyramid fully convolutional network for hyperspectral and multispectral image fusion," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 5, pp. 1549–1558, May 2019.

[6] 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. (ICCV)*, Oct. 2017, pp. 1753–1761.

[7] L.-J. Deng, G. Vivone, C. Jin, and J. Chanussot, "Detail injection-based deep convolutional neural networks for pansharpening," *IEEE Trans. Geosci. Remote Sens.*, early access, Oct. 27, 2020, doi: 10.1109/TGRS.2020.3031366.

[8] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proc. 27th Int. Conf. Int. Conf. Mach. Learn.*, Jun. 2010, pp. 399–406.

[9] L. Wald, T. Ranchin, and M. Mangolini, "Fusion of satellite images of different spatial resolutions: Assessing the quality of resulting imagem," *Photogramm. Eng. Remote Sens.*, vol. 63, no. 6, pp. 691–699, 1997.

[10] B. Aiazzi *et al.*, "MTF-tailored multiscale fusion of high-resolution MS and PAN imagery," *Photogramm. Eng. Remote Sens.*, vol. 72, no. 5, p. 591–596, May 2006.

[11] F. Huang and A. Anandkumar, "Convolutional dictionary learning through tensor factorization," *Comput. Sci.*, vol. 4, pp. 116–129, Oct. 2015.

[12] H. Sreter and R. Giryes, "Learned convolutional sparse coding," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 2191–2195.

[13] Y. Wei, Q. Yuan, H. Shen, and L. Zhang, "Boosting the accuracy of multispectral image pansharpening by learning a deep residual network," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 10, pp. 1795–1799, Oct. 2017.

[14] L. He *et al.*, "Pansharpening via detail injection based convolutional neural networks," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 4, pp. 1188–1204, Apr. 2019.

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

[16] R. H. Yuhas, A. F. Goetz, and J. W. Boardman, "Discrimination among semi-arid landscape endmembers using the spectral angle mapper (SAM) algorithm," in *Proc. 3rd Annu. JPL Airborne Geosci. Workshop*, vol. 1, 1992, pp. 147–149.

[17] J. Zhou, D. L. Civco, and J. A. Silander, "A wavelet transform method to merge landsat TM and SPOT panchromatic data," *Int. J. Remote Sens.*, vol. 19, no. 4, pp. 743–757, Jan. 1998.

[18] L. Alparone, B. Aiazzi, S. Baronti, A. Garzelli, F. Nencini, and M. Selva, "Multispectral and panchromatic data fusion assessment without reference," *Photogramm. Eng. Remote Sens.*, vol. 74, no. 2, pp. 193–200, Feb. 2008.