When Pansharpening Meets Graph Convolution Network and Knowledge Distillation

Keyu Yan, Man Zhou, Liu Liu, Chengjun Xie, and Danfeng Hong

***Abstract*—In this article, we propose a novel graph convolutional network (GCN) for pansharpening, defined as GCPNet, which consists of three main modules: the spatial GCN module (SGCN), the spectral band GCN module (BGCN), and the atrous spatial pyramid module (ASPM). Specifically, due to the nature of GCN, the proposed SGCN and BGCN are capable of exploring the long-range relationship between the object and the global state in the spatial and spectral aspects, which benefits pansharpened results and has not been fully investigated before. In addition, the designed ASPM is equipped with multiscale atrous convolutions and learns richer local feature information, so as to cover the objects of different sizes in satellite images. To further enhance the representation of our proposed GCPNet, asynchronous knowledge distillation is introduced to provide compact features by heterogeneous task imitation in a teacher–student paradigm. In the paradigm, the teacher network acts as a variational autoencoder to extract compact features of the ground-truth MS images. The student network, devised for pansharpening, is trained with the assistance of the teacher network to transfer the important information of the expected ground-truth MS images. Extensive experimental results on different satellite datasets demonstrate that our proposed network outperforms the state-of-the-art methods both visually and quantitatively. The source code is released at https://github.com/Keyu-Yan/GCPNet.**

***Index Terms*—Asynchronous knowledge distillation, atrous convolution, graph convolutional network (GCN), pansharpening.**

I. Introduction

In THE field of remote sensing, with the development of imaging systems and satellite technology, abundant satellite images are available in daily life. However, limited by the hardware conditions of multispectral imaging devices, currently, common optical satellites (such as WordView and GaoFen) usually provide two types of remote sensing images: multispectral image low-resolution multispectral (LMS) with rich spectral information but low spatial resolution, and panchromatic image (PAN) with rich spatial details but only gray information [1]. In order to obtain a high spatial quality multispectral image, the goal of pansharpening is to fuse LMS and PAN by integrating their complementary advantages for the purpose of improving the spatial quality of the fused image (MS) on the premise of preserving multispectral information as much as possible. Considering the above utility, the pansharpening task is generally regarded as a key preprocessing step for many remote sensing data applications [2]–[4], such as object detection [5]–[7], land cover classification [8], urban impervious surface extraction [9], and change detection [10]. Traditional pansharpening methods usually require reasonable assumptions based on prior knowledge; otherwise, it is easy to cause distortion of fused images. In addition, these methods of traditional transformation only have the ability of shallow nonlinear expression; therefore, it is difficult to achieve a good balance between the improvement of spatial quality and the maintenance of spectral quality.

Afterward, inspired by the success of deep learning over natural image processing, deep learning has been introduced to the field of pansharpening in recent years by virtue of its powerful ability to significantly represent local complex structures. Pansharpening networks based on the convolutional neural network (CNN) [11], [12] have been proposed that can learn nonlinear mapping and high semantic features from a large number of paired images, significantly improving the performance and robustness of the pansharpening process. Despite remarkable results, several commonly recognized issues remain to be solved.

1) *Insufficient Feature Extraction:* Most of the existing methods only stack pure feedforward convolution operations to extract the features without fully exploring its potentials like long-rang information and cross-spectral relationship, thus limiting the model performance. Like the latest advance, Cai and Huang [2] tend to improve performance by continuously deepening the network, which leads to the introduction of more parameters in the training process of the deep network, resulting in the increase of memory and computation.

2) *Scale Variance:* Different satellite images have different resolutions, and the scale of imaging objects will be different, so the problem of multiscale cannot be ignored. Some solutions exploit multiple convolution kernels with different receptive fields but bring huge computational costs. Furthermore, the unsupervised pansharpening method [13] that aims to handle differences between features according to the characteristics of the spectral information and multiscale information does not require a large dataset and the availability

Manuscript received October 3, 2021; revised December 1, 2021, March 15, 2022, and April 12, 2022; accepted April 12, 2022. Date of publication April 18, 2022; date of current version May 2, 2022. This work was supported by the National Natural Science Foundation of China under Grant 32171888. *(Keyu Yan and Man Zhou are co-first authors.) (Corresponding author: Chengjun Xie.)*

Keyu Yan and Man Zhou are with the Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei 230031, China, and also with the Science Island Branch, University of Science and Technology of China, Hefei 230026, China (e-mail: keyu@mail.ustc.edu.cn; manman@mail.ustc.edu.cn).

Liu Liu is with the Department of Computer Science, Shanghai Jiao Tong University, Shanghai 200240, China.

Chengjun Xie is with the Institute of Intelligent Machines, Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei 230031, China (e-mail: cjxie@iim.ac.cn).

Danfeng Hong is with the Key Laboratory of Computational Optical Imaging Technology, Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094, China (e-mail: hongdf@aircas.ac.cn).

Digital Object Identifier 10.1109/TGRS.2022.3168192

of ground-truth MS but needs to design appropriate constraints to obtain useful feature representation.

3) *Inefficient Utilization of Ground Truth:* As recognized, the ground-truth multispectral (MS) images possess the complementary information (e.g., high-frequency component) of low-resolution (LR) MS images, which can be considered as privileged information to alleviate the spectral distortion and insufficient spatial texture enhancement. Since existing pansharpening methods only utilize the ground-truth MS image to supervise the network training, its potential value has not been fully explored.

To solve the problems mentioned above, we propose the novel graph convolutional network (GCN) for pansharpening, defined as GCPNet, which consists of three main modules: the spatial GCN module (SGCN), the spectral band GCN module (BGCN), and the atrous spatial pyramid module (ASPM). The proposed GCPNet aims to integrate the long-range information through GCN, make full use of the internal relationship in the spatial and spectral dimensions, and support image reconstruction by obtaining global spatial information and cross-spectral relationship. This just makes up for the disadvantage of CNN's focusing on local information, which is not conducive to image reconstruction and leads to the loss of feature information due to prior geometric shapes. The designed ASPM learns multiscale feature information and obtains different receptive fields through atrous convolutions of different sizes in series and parallel, so as to adapt to objects of different sizes in satellite images. In addition, to fully explore the potential of ground truth, we also adopt a new method of knowledge distillation, asynchronous knowledge distillation, where the teacher and the student deal with different tasks, but the teacher can learn more compact information and transfer knowledge to the student through feature distillation to further enhance the student's pansharpening ability. In Fig. 1, the point closer to the left suggests higher speed or fewer parameters, and it can be seen that GCPNet is located in the upper left corner of the coordinate system and achieves state-of-the-art performance.

The main contributions of the proposed GCPNet are given as follows.

1) We propose an efficient GCPNet model for pansharpening, which uses SGCN and spectral BGCN to explore the long-range spatial and spectral relations, and ASPM to learn multiscale information. Because of the innovative design of the GCN module that can effectively capture features, the whole model also has absolute advantages over the latest models in terms of the number of parameters and computation.
2) An ASPM is designed, which obtains a variety of receptive field sizes through serial and parallel connections of empty convolution, with stronger nonlinear expression ability and avoids information loss caused by upsampling and downsampling operations commonly used in previous multiscale modules. The ASPM can aggregate multiscale features and significantly improve the representation ability of neural networks.

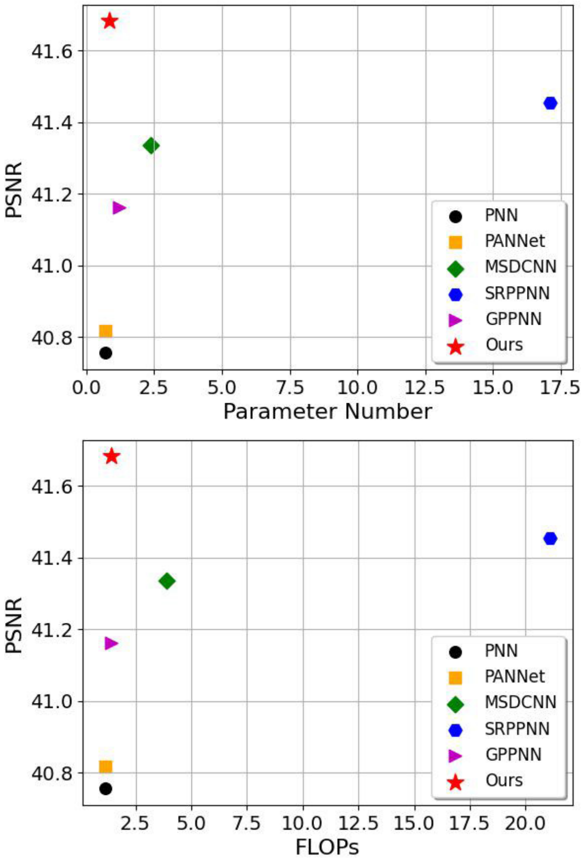

Fig. 1. Performance versus speed and trainable parameter numbers. Points closer to the left suggest higher speed or fewer parameters, while points closer to the right suggest better performance.

3) We redevelop a new knowledge distillation framework for pansharpening and devise an encoder–decoder teacher network of knowledge distillation to transfer the important knowledge of the ground-truth MS images to the student network to enhance its representation.

The remaining part of this article is organized as follows. Section II gives a brief review of related works. Section III describes, in detail, the proposed network and asynchronous knowledge distillation. Section IV illustrates comparative experiments and ablation experiments. Section V contains the conclusion.

II. Related Work

A. Pansharpening

Now that remote sensing image fusion is helpful for many remote sensing applications; many pansharpening algorithms have been proposed by researchers in recent years. There are many traditional image fusion methods, mainly including the component substitution method and the multiresolution analysis method. Of course, there are other methods based on model optimization, the hybrid method [14], the variation method [15], and the sparse-representation method [16]. As the most traditional fusion method, the component substitution method first projects the multispectral image into a new space, then replaces its structural components with PANs in whole or in part, and then obtains the final fusion result through the inverse transformation of space. It mainly includes the intensity–hue–saturation (IHS) methods [17], the principal component analysis (PCA) methods [18], [19], the Gram–Schmidt (GS) methods [20], the Brovey transformation methods [21], and so on. However, as a certain component of MS is considered to be all spatial information, it actually contains certain spectral information, so the fusion results obtained by simply replacing these components with PAN often have spectral distortion. The multiresolution analysis methods mainly include the Wavelet transform [22], the Laplacian pyramid transform [23], the nonsubsampled contourlet transform [24], the curvelet transform [25], and so on. By extracting spatial details from PANs and injecting them into the upsampled multispectral images at different scales, spatial resolutions, and decomposition layers, the final high-resolution multispectral images can be obtained. For the most part, these methods can maintain good spectral features, but there will be a distortion of spatial structure in the satellite images.

These traditional methods mentioned above are mostly linear model fusion, which can only reflect the limited prior knowledge of images, so it is difficult to achieve a good balance between improving spatial quality and maintaining spectral quality. The complex transformations between spectral and spatial domains should be considered highly nonlinear. In order to maintain the fidelity of these images observed, we need highly nonlinear functions for simulation. With the development of deep learning, CNN has been applied to the study of satellite image fusion, and a good fusion effect has been achieved depending on its nonlinear advantages in the mapping process [26]–[29]. For example, Zhong *et al.* [30] proposed a pansharpening method combining the super-resolution convolution neural network SRCNN [31] model and GS transformation. Although this method achieves good results, it is not an end-to-end mapping process and does not completely leave the traditional method. In order to model the pansharpening process as end-to-end mapping, Masi *et al.* [12] proposed a network named PNN. However, direct learning of the relationship between low-resolution images and high-resolution images would have more redundancy, making it difficult to learn the model well. Yang *et al.* [32] designed PANNet architecture by adding upsampled multi-spectral images to the network's output, transmitting spectral information directly to reconstructed images, and training the network's parameters in high-pass filtering domain. However, these methods are only simply stacking models in image classification; even so, the final effect of the model is far superior to the traditional methods. Cai and Huang [2] have achieved good results in pansharpening by using the method of image super-resolution, but the amount of computation and parameter is greatly increased. Both MSDCNN [33], which is proposed to extract multiscale features, and GPPNN [34], which has two optimization problems regularized by the deep prior, are carefully designed for pansharpening to improve efficiency and generate the high-quality image. Besides, Qu *et al.* [13] adopted the fully connected layers in the pansharpening network, but the number of network parameters will increase significantly, and the network consumes more testing time.

B. Graph Convolutional Network

Research on the graph neural network (GNN) can be traced back to the pioneering work of Scarselli *et al.* [35]. They designed mapping functions from graph structure space to the m-dimensional Euclidean space and proposed a supervised learning algorithm that can update parameters in GNN models. However, this model does not use convolution. Later, Bruna *et al.* [36] combined the idea of convolution in the spectrum graph theory with GNN and the proposed graph convolution network (GCN). Different from CNN, which relies heavily on the geometry of prior conditions, GCN eases the assumption of prior conditions, which takes the research object as the node and the correlation or similarity between objects as the edge. It can deal with complex paired interactions and integrate global spatial data, make full use of the internal relations between objects, and mine invisible relations between objects. In recent years, the graph convolution theory has developed rapidly. It has not only been widely applied to various high-level vision tasks, such as action recognition [37] and semantic segmentation [38], [39], but also started to be used to solve low-level vision tasks, such as image inpainting [40], image deraining [41], and image denoising [42]. Furthermore, dual GCNs [43] with different mapping strategies become popular. Bandara *et al.* [44] proposed spatial and interaction space graph reasoning to extract roads from aerial images. As far as we know, GCN is currently used for very little hyperspectral imagery. Qin *et al.* [45] and Wan *et al.* [46] have related work, but it is limited to the task of hyperspectral image classification [47]–[49]. We are the first to apply GCN to pansharpening to improve the quality of results by proposing SGCN and BGCN in a cascaded manner to efficiently and effectively capture long-range and global spatial and spectral information.

C. Knowledge Distillation

Early knowledge distillation framework [50] uses a large and cumbersome teacher model to supervise the learning process of a smaller and faster student model to achieve the purpose of the compression model. Nowadays, knowledge distillation [51]–[53] usually transfers knowledge between two deep models, transferring the representation ability of the teacher model to the student model to improve the performance of the student model. Inspired by this idea, we propose an asynchronous knowledge distillation method to improve the performance of the lightweight student model. Specifically, the teacher and the student handle different tasks, and the teacher can learn more potential knowledge while working on his own task and then pass this information on to the student through distillation to help the student complete the pansharpening task.

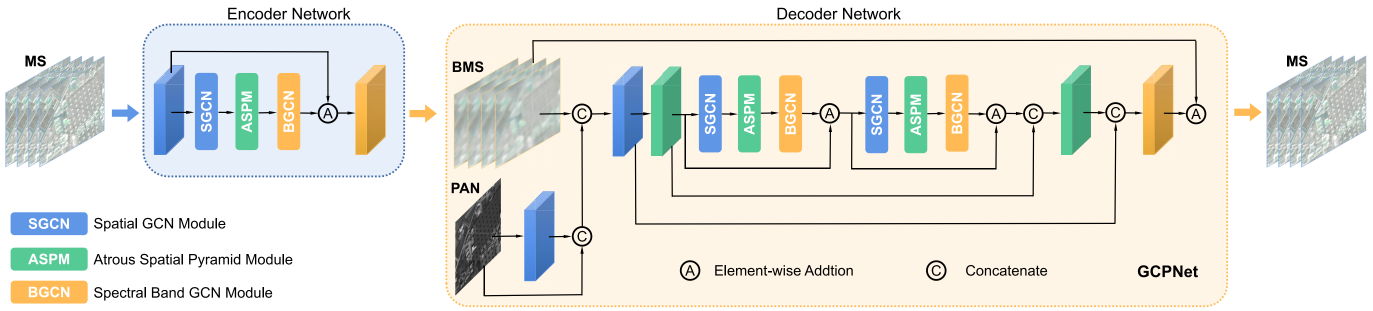

Fig. 2. Overall structure of the proposed graph convolutional framework consists of two main parts: the encoder network and the decoder network, including SGCN, spectral BGCN, and ASPM. Note that we have defined the decoder network here as GCPNet.

III. Proposed Method

In this section, we outline the proposed graph convolutional framework (GCPNet) and elaborate on the three key components: the SGCN module, the spectral BGCN module, and ASPM. In addition, we introduce the realization method of asynchronous knowledge distillation.

A. Overall Network Architecture

CNN can only process data of local structures, but we desire to extract long-range and global features effectively. Therefore, GCNs, which can describe one-to-many relationships of data, have become the focus of research. However, the most existing deep learning methods to handle pansharpening tasks usually just use CNN blocks as the main feature extraction units. We design a new network that can extract spatial and spectral information very efficiently through GCN.

Fig. 2 presents the flowchart of the proposed method that consists of two major components: the encoder network and the decoder network. In the training process of asynchronous knowledge distillation, the encoder–decoder network will be used, but we only need the decoder network called GCPNet for the actual pansharpening task.

We denote input image as $\mathbf{X}$ and the network output image as $\mathbf{Y}$. Thus, the inputs and outputs of the decoder network can be expressed as $\mathbf{X}_{\text{BMS}} \in \mathbb{R}^{W\times H\times C}$, $\mathbf{X}_{\text{PAN}} \in \mathbb{R}^{W\times H\times 1}$, and $\mathbf{Y}_{\text{MS}} \in \mathbb{R}^{W\times H\times C}$, where W and H represent the width and height of the image, C is the number of image bands, and $\mathbf{X}_{\text{BMS}}$ is obtained by $\mathbf{X}_{\text{LMS}} \in \mathbb{R}^{w\times h\times C}$ through bicubic interpolation upsampling. The original goal is to generate the high spatial resolution and high spectral resolution image from the input image $\mathbf{X}_{\text{LMS}}$ and $\mathbf{X}_{\text{PAN}}$ via solving the following optimization problem:

$$\min_{\mathbf{X}} \mathcal{L}(f(\mathbf{X}_{\text{LMS}}, \mathbf{X}_{\text{PAN}}), \mathbf{X}_{\text{MS}}). \tag{1}$$

We turn the problem into an approximate optimization problem

$$\min_{\mathbf{X}} \mathcal{L}(f(\mathbf{X}_{\text{BMS}}, \mathbf{X}_{\text{PAN}}), \mathbf{X}_{\text{MS}}) \tag{2}$$

where $f(.)$ refers to the operation of the proposed GCPN method and $\mathcal{L}$ is a loss function.

In the following, we give details of GCPNet. Large convolution kernel size is adopted in both input and output parts of the network, which is a common technique used in existing methods [12], [32]. Because of its large receptive field, it can obtain global features and maintain the original image structure, which is conducive to extracting features by GCN and reconstructing the network's output image. In the intermediate structure of the network, a small convolution kernel size is used to pay attention to details and reduce the number of parameters.

We deploy the SGCN module to capture local-to-global spatial information. Then, this spatial information is sent into the ASPM to assist the network to extract multiscale local spatial features. The shapes of different landforms are different at different spatial scales, so this effect must also be taken into account during panchromatic sharpening. To obtain spectral information that is complementary to spatial information, we employ the spectral BGCN module to explore the correlation among the features that contain rich global and local spectral representations. Furthermore, we adopt symmetric skip connections to link shallow and deep layers. This can not only avoid the gradient vanishing but also propagate image detail to improve the pansharpening performance.

B. Spatial GCN Module (SGCN)

Graph convolution allows the model to aggregate the global pixels at all spatial positions as the response at a position. The purpose of this module is to explore the relationship between one pixel and all pixels in the feature map. Let a feature map be $\mathbf{F} \in \mathbb{R}^{N\times W\times H}$, where N is the number of channel, and W and H are the width and height of $\mathbf{F}$, respectively. The graph convolution is defined as the simple form by Kipf and Welling [54]

$$\mathbf{Z} = \hat{\mathbf{A}}\mathbf{F}\Theta \tag{3}$$

where $\mathbf{Z}$ is the convolved signal matrix, $\hat{\mathbf{A}}$ is the adjacency matrix, and Θ is a matrix of filter parameters. Similar to the nonlocal network [55] which can be thought of as a form of fully connected GCN, we use three convolution layers on the input feature map $\mathbf{F}_{\text{in}} \in \mathbb{R}^{N\times W\times H}$ to reduce the channel number from N to $(N/2)$. As shown in Fig. 3, we use $\varphi(\cdot)$, $\theta(\cdot)$, and $\delta(\cdot)$ to represent the three convolution layers mentioned above. The new feature is defined in the form of spatial graph

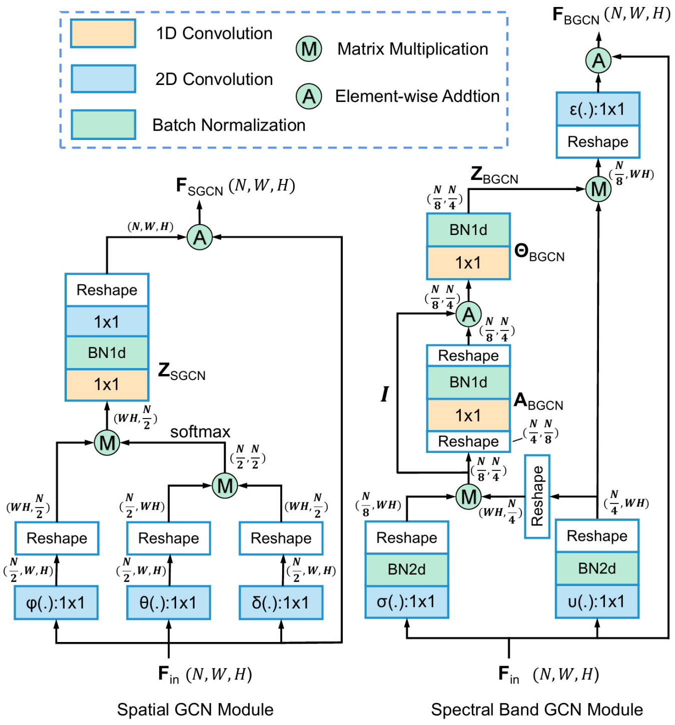

Fig. 3. Architecture of proposed SGCN and BGCN modules that are used to form the GCPNet. The spatial and spectral BGCNs focus on exploring space and spectral contextual information along two orthogonal dimensions. (Best viewed in color.)

convolution learning

$$\begin{aligned}\mathbf{Z}_{\mathrm{SGCN}} &= \hat{\mathbf{A}}_{\mathrm{SGCN}}\mathbf{F}\Theta_{\mathrm{SGCN}}\\ &= \varphi(\mathbf{F}_{\mathrm{in}})\theta(\mathbf{F}_{\mathrm{in}})^{\mathsf{T}}\delta(\mathbf{F}_{\mathrm{in}})\Theta_{\mathrm{SGCN}}\end{aligned} \tag{4}$$

where $\mathbf{Z}_{\mathrm{SGCN}}$ is output of SGCN and T is the transpose operation. $\varphi(\cdot)\theta(\cdot)^{\mathsf{T}}$ is performed by matrix multiplication and can be seen as the adjacency matrix $\hat{\mathbf{A}}_{\mathrm{SGCN}}$. To simulate block operations, these 1×1 convolution layers [$\varphi(\cdot)$, $\theta(\cdot)$, and $\delta(\cdot)$] are implemented to replace the $n \times n$ sliding window, which is used in the traditional nonlocal algorithms. According to the associative rule, we replace the original term $(\varphi(\cdot)\theta(\cdot)^{\mathsf{T}})\delta(\cdot)$ with $\varphi(\cdot)(\theta(\cdot)^{\mathsf{T}}\delta(\cdot))$. By doing so, the computational complexity of the measurement matrix can be reduced from $O((WH)^2)$ to $O((WH))$ compared with the generic nonlocal module [55]. In order to obtain the global feature correlation, we multiply the features of the output of the $\delta(\cdot)$ convolution layer and $\theta(\cdot)$ convolution layer by the matrix and then multiply the features of the output of the $\varphi(\cdot)$ convolution layer. The *softmax* operation is used to avoid numerical instabilities and is found to give better convergence [56]. The weighting process of Θ, a hidden-to-output weight matrix, is conducted by using one 1×1 convolution layer to perform the operation.

Finally, before output, the features are further tuned via the 1×1 convolution block by the following formula:

$$\mathbf{F}_{\mathrm{SGCN}} = \mathrm{BN}(\mathbf{Z}_{\mathrm{SGCN}})\mathbf{W}_{\mathrm{SGCN}} + \mathbf{F}_{\mathrm{in}} \tag{5}$$

where $\mathrm{BN}(\cdot)$ refers to the batch normalization operation and $\mathbf{W}_{\mathrm{SGCN}}$ denotes the weight of the output convolution layer. For residual learning, we add the item of $\mathbf{F}_{\mathrm{in}}$.

C. Spectral Band GCN Module (BGCN)

Different from natural images, MS images have a unique near infrared band (NIR) in addition to red, green, and blue color channels. PAN images, however, have only single-band channel. As for the pansharpening problem, panchromatic (PAN) and multispectral (MS) images need to trade off not only in spatial resolution but also in spectral space. In order to effectively utilize the internal relationship between PAN images and MS images of different bands, we designed the spectral BGCN module to reason the spectral correlation. We model our spectral BGCN as

$$\mathbf{Z}_{\mathrm{BGCN}} = (\mathbf{I} + \mathbf{A}_{\mathrm{BGCN}})\mathbf{F}\Theta_{\mathrm{BGCN}} \tag{6}$$

where $\mathbf{A}_{\mathrm{BGCN}} \in \mathbb{R}^{(N/8)\times(N/8)}$ is the adjacency matrix measuring the relations of the graph, and $\Theta_{\mathrm{BGCN}} \in \mathbb{R}^{(N/4)\times(N/4)}$ is the weight matrix. Note that both of these matrices are implemented by 1×1 1-D convolutions and learned from data. We utilize identity matrix $\mathbf{I}$ to propagate the node features over the graph to perform Laplacian smoothing, which is also used in [57], [58].

In practice, we first adopt two 1×1 2-D convolution layers [$\sigma(\cdot)$ and $\upsilon(\cdot)$] on the input feature $\mathbf{F}_{\mathrm{in}} \in \mathbb{R}^{N\times W\times H}$. $\sigma(\cdot)$ aims to reduce the dimension, which can reduce the computation and the number of parameters. $\upsilon(\cdot)^{\mathsf{T}}$ is projection weights, which can map original features to the spectral interaction space [57]. Thus, the size of feature $\mathbf{F}$ in (6) is $(N/8) \times (N/4)$. From the perspective of a graph, this means that there are $(N/8)$ nodes, and the dimension of each node is $(N/4)$. Through $\mathbf{A}_{\mathrm{BGCN}}$ and Θ_{BGCN}, we construct a fully connected graph on the $\mathbf{F}$ to obtain the spectral relationship. As shown in Fig. 3, the procedure of this module can be expressed as

$$\mathbf{Z}_{\mathrm{BGCN}} = (\mathbf{I} + \mathbf{A}_{\mathrm{BGCN}})\sigma(\mathbf{F}_{\mathrm{in}})\upsilon(\mathbf{F}_{\mathrm{in}})^{\mathsf{T}}\Theta_{\mathrm{BGCN}} \tag{7}$$

$$\mathbf{F}_{\mathrm{BGCN}} = f_R(\mathbf{Z}_{\mathrm{BGCN}}) + \mathbf{F}_{\mathrm{in}} \tag{8}$$

where function $f_R(\cdot)$ denotes the hidden-to-output operation used for the image reconstruction. Since the dimension of generated graph $\mathbf{Z}_{\mathrm{BGCN}}$ is $(N/8) \times (N/4)$, we add one function $f_R(\cdot)$ to reverse project and readjust feature to the form of feature map. Specifically, we first multiply the generated graph $\mathbf{Z}_{\mathrm{BGCN}}$ by $\upsilon(\mathbf{F}_{\mathrm{in}})$ and then utilize one 2-D convolutional layer $\varepsilon(\cdot)$ to transform the number of bands to N. Finally, the output feature map $\mathbf{F}_{\mathrm{BGCN}} \in \mathbb{R}^{N\times W\times H}$ can participate in subsequent operations, whose shape is the same as $\mathbf{F}_{\mathrm{in}}$. By deploying the spectral BGCN module into the GCPNet, our model can focus on the correlation between different spectral in the generated MS images.

D. Atrous Spatial Pyramid Module (ASPM)

In remote sensing images, the subjects of images are usually vehicles, houses, fields, mountains, and so on. However, there are often large-scale differences among these objects. Therefore, spatial scale is a relatively important factor in restoring high spatial resolution images. For multiscale problems, the pyramid pooling module (PMM) [59] and atrous spatial pyramid pooling (ASPP) [60] methods are widely used in related tasks. Different from previous methods, we specially designed

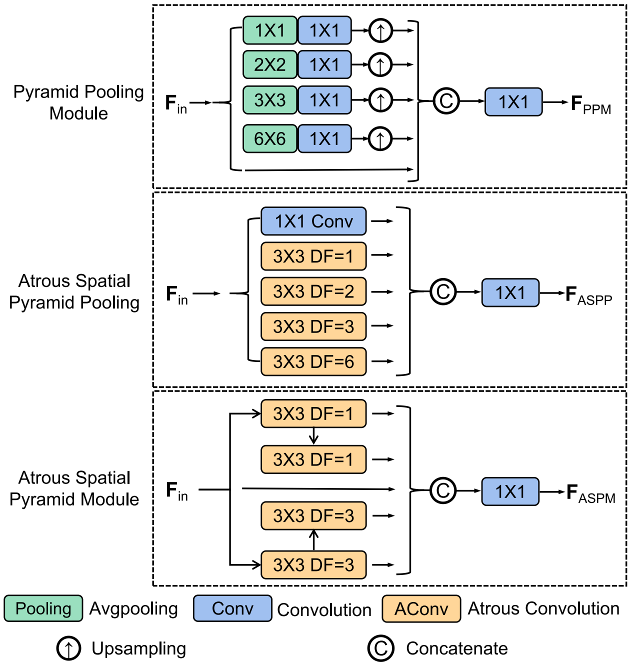

Fig. 4. Architecture of the pyramid pooling module (PMM), ASPP, and our ASPM. *DF* denotes the dilation factor of atrous convolution. (Best viewed in color.)

for pansharpening the ASPM structure, which is responsible for the feature extraction in different scales.

PMM extracts features of different scales by pooling layers of different sizes and, finally, restores the original input feature map size by bilinear interpolation. However, using the pooling layer in the module for downsampling can cause the loss of information. ASPP abandons the pooling layer and uses the parallel connection of atrous convolutions to generate multiple types of features from different spatial scales so that the obtained features keep the same size as the original image, avoiding downsampling and upsampling operations in PMM.

As shown in Fig. 4, our ASPM also takes atrous convolutions, acquiring different receptive fields by combining parallel and cascade connections of atrous convolutions. In this way, we only need two types of atrous convolution to make the fused feature $\mathbf{F}_{\text{ASPM}}$ have the same receptive fields as ASPP, i.e., 3 × 3, 5 × 5, 7 × 7, and 13 × 13. Finally, a 1 × 1 convolution operation is used to fuse different receptive field features to enhance feature representation ability.

This module can obtain the multiscale features from local spatial regions contained in the fusion features, which is exactly compensated with the two GCN modules proposed to learn the global space and spectral bands, respectively.

E. Asynchronous Knowledge Distillation

As shown in Fig. 5, we propose an asynchronous knowledge distillation framework, which contains an encoder–decoder teacher network and a pansharpening network (GCPNet). The student and teacher's decoder networks are designed to have the same architecture, but the tasks that they should handle are different. In order to help train the pansharpening network later, we let the teacher network learn how to reconstruct ground truth from high-resolution multispectral (MS) images and high-resolution single-band panchromatic (PAN) images. In the experiments, the teacher network can accurately recover the ground truth, so we consider that the teacher learned the distribution of high-resolution multispectral images and can provide the student with favorable prior knowledge to help it learn how to restore high-resolution multispectral images (MS) from low-resolution multispectral (LMS) images and high-resolution single-band panchromatic (PAN) images.

Since the input of the teacher network is MS image $\mathbf{X}_{\text{MS}}$, the output $\mathbf{Y}^T_{\text{MS}}$ should be as similar as possible to the input image $\mathbf{X}_{\text{MS}}$ so that teacher network only learns to copy the input image to rebuild MS image but cannot extract useful features. Therefore, we exploit the encoder–decoder architecture to promote the teacher to extract valuable information and facilitate the transfer of favorable prior knowledge to students via the decoder network. It first projects the MS image into a low-dimensional feature space and then uses the generated BMS and PAN images to restore the original MS image so that teacher can learn to extract better feature representation to complete the task of MS image reconstruction. We apply pansharpening loss $\mathcal{L}^T_p$ and imitation loss $\mathcal{L}^T_i$ to train the teacher network. Specifically, the $\mathcal{L}^T_p$ loss is defined as the mean square error between $\mathbf{X}_{\text{MS}}$ and $\mathbf{Y}^T_{\text{MS}}$

$$\mathcal{L}^T_p = \frac{1}{2N}\sum_{n=1}^{N}\sum_{i}^{W}\sum_{j}^{H}\left(\mathbf{X}^n_{\text{MS}}(i,j) - \mathbf{Y}^{nT}_{\text{MS}}(i,j)\right)^2 \tag{9}$$

where W and H are the width and the height of the MS image, respectively. N denotes the number of images in a training batch. $\mathbf{X}^n_{\text{MS}}(i,j)$ denotes the intensity value of the nth $\mathbf{X}_{\text{MS}}$ at position (i,j) and so is $\mathbf{Y}^{nT}_{\text{MS}}(i,j)$. The $\mathcal{L}^T_i$ term limits the representation ability of the encoder and makes the encoder's output close to $\mathbf{X}_{\text{BMS}}$ image. The imitation loss computes mean absolute error between $\mathbf{X}_{\text{BMS}}$ and $\mathbf{Y}_{\text{BMS}}$ defined as

$$\mathcal{L}^T_i = \frac{1}{N}\sum_{n=1}^{N}\sum_{i}^{W'}\sum_{j}^{H'}\left|\mathbf{X}^n_{\text{BMS}}(i,j) - \mathbf{Y}^n_{\text{BMS}}(i,j)\right| \tag{10}$$

where W' and H' are width and height of the BMS image, respectively. In summary, the final teacher's loss function is the sum of the two losses

$$\mathcal{L}^T_{\text{sum}} = \mathcal{L}^T_p + \lambda^T\mathcal{L}^T_i \tag{11}$$

where λ^T is applied to balance the contributions between $\mathcal{L}^T_p$ and $\mathcal{L}^T_i$.

After training the teacher network, we initialize the weight of the student network with the weight of the decoder in the teacher, so as to transfer the teacher's reconstruction ability to the students [51], [61]. Then, we fix the parameters of the teacher network and further train the student network with pansharpening loss $\mathcal{L}^S_p$ and distillation loss $\mathcal{L}^S_d$. $\mathcal{L}^S_p$ is similarly defined as $\mathcal{L}^T_p$

$$\mathcal{L}^S_p = \frac{1}{2N}\sum_{n=1}^{N}\sum_{i}^{W}\sum_{j}^{H}\left(\mathbf{X}^n_{\text{MS}}(i,j) - \mathbf{Y}^{nS}_{\text{MS}}(i,j)\right)^2. \tag{12}$$

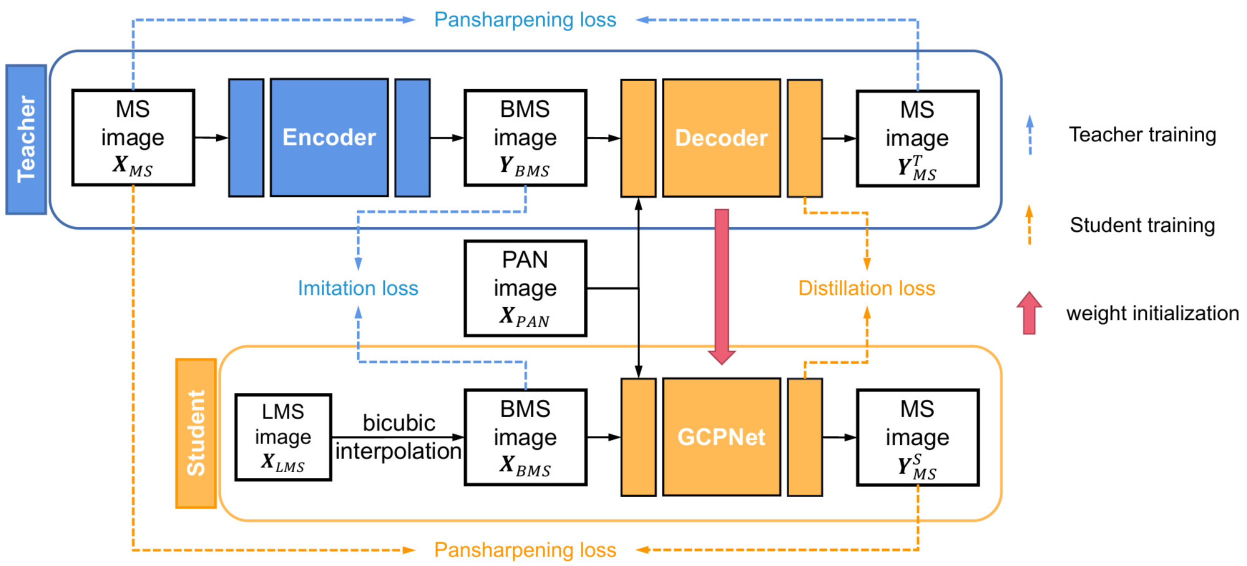

Fig. 5. Overview of our asynchronous knowledge distillation framework. The teacher network inputs an MS image and extracts a feature representation $\mathbf{Y}_{\text{BMS}}$ of an approximate BMS image using an encoder. Then, the decoder network reconstructs an MS image output. To train the teacher network, we use imitation loss and pansharpening loss. After training the teacher, the student network is initialized with the weights of the decoder in the teacher network (red line). Note that the student network and the decoder share the same network architecture. To train the student network, we use distillation loss and pansharpening loss. (Best viewed in color.)

According to the experimental results, teacher has stronger pansharpening ability, so the term distillation is used to impart teacher's knowledge to student. Specifically, the $\mathcal{L}_d^S$ loss minimizes the featurewise error between the teacher's feature map and the student's feature map. The feature maps from penultimate convolution layer in the decoder are used to calculate distillation loss. The $\mathcal{L}_d^S$ loss is computed by

$$\mathcal{L}_d^S = \frac{1}{2N}\sum_{n=1}^{N}\sum_{i}^{W'}\sum_{j}^{H'}\left(\mathbf{F}_n^T(i,j) - \mathbf{F}_n^S(i,j)\right)^2 \tag{13}$$

where $\mathbf{F}_n^T$ and $\mathbf{F}_n^S$ are the feature maps of teacher and student networks, respectively. Overall, we use the following loss to train the student network:

$$\mathcal{L}_{\text{sum}}^S = \mathcal{L}_p^S + \lambda^S \mathcal{L}_d^S \tag{14}$$

where λ^S is a distillation parameter. The experimental results show that the performance of our GCPNet can be further improved by asynchronous knowledge distillation.

IV. Experiments

A. Datasets and Implementation Details

We conduct several experiments on datasets collected by GaoFen-2 (GF2), WordView II (WV2), and WordView III (WV3). The spatial resolutions of WV2, WV3, and GF2 on MS and PAN images were 0.5 and 1.8 m, 0.31 and 1.24 m, and 1 and 4 m, respectively. Following the previous works [33], [62], we crop the original satellite images into MS image patches of 128 × 128 × 4 and PAN image patches of 128 × 128 × 1, and further downsample the MS images into LMS image patches of 32 × 32 × 4 through bicubic interpolation. We split the datasets of the three satellites into 90% for training and 10% for validation approximately.

First, in order to train the teacher network, we use stochastic gradient descent (SGD) with a momentum equal to 0.9 to minimize the objective function in (11), where λ^T of (11) is set to 1e-4. We set the minibatch size to 8, the initial learning rate to 0.01, and the number of total training epochs to 2000, and decrease the learning rate by a factor of 10 in the 1000th epoch. The numbers of iterations in one epoch on GaoFen-2 (GF2), WorldView III (WV3), and WorldView II (WV2) are 1017, 807, and 285, respectively. In addition, we set the threshold of gradient clipping to 0.1. Although the model convergence will be slower, this can make the training process more stable.

After training the teacher network, we also use SGD with a momentum equal to 0.9 to minimize the objective function in (13), where λ^S is set to 1e-6. We set the minibatch size to 4, the threshold of gradient clipping to 0.2, the initial learning rate to 0.01, and the number of total training epochs to 2000 and decrease the learning rate by a factor of 10 in the 1600th epoch.

Models are implemented via PyTorch on GTX TITAN X GPUs on the desktop with Ubuntu 18.04, CUDA 10.2, and CUDNN 7.5.

We evaluate the algorithm performance using the following six image quality assessment (IQA) metrics that can be calculated with references and are widely used in pansharpening missions: the peak signal-to-noise ratio (PSNR), the structural similarity index (SSIM), the spectral angle mapper (SAM) [63], the relative dimensionless global error in synthesis (ERGAS), the spatial correlation coefficient (SCC), and the four-band extension of Q (Q4). Furthermore, we adopt the spectral distortion index D_λ, the spatial distortion index D_S, and the quality with no reference (QNR) method to evaluate the results of reference-free measure. The best values for these metrics are $+\infty$, 1, 0, 0, 1, 1, 0, 0, and 1, respectively.

TABLE I

QUANTITATIVE COMPARISON OF NINE METHODS ON THE GAOFEN-2 DATASETS. THE BEST, SECOND BEST, AND THIRD BEST RESULTS ARE HIGHLIGHTED BY RED, BLUE, AND UNDERLINE, RESPECTIVELY. ↑ INDICATES THAT THE LARGER THE VALUE, THE BETTER THE PERFORMANCE, AND ↓ INDICATES THAT THE SMALLER THE VALUE, THE BETTER THE PERFORMANCE

Methods	PSNR ↑	SSIM ↑	SAM ↓	ERGAS ↓	SCC ↑	Q ↑	D_λ ↓	D_S ↓	QNR ↑
Brovey	37.7974	0.9026	0.0218	1.3720	0.6446	0.3857	0.0905	0.1443	0.7790
IHS	38.1754	0.9100	0.0243	1.5336	0.6738	0.3682	0.0418	0.1345	0.8301
SFIM	36.9060	0.8882	0.0318	1.7398	0.8128	0.4349	0.0691	0.1312	0.8109
Wavelet	35.7502	0.8213	0.0283	2.0418	0.6515	0.2859	0.1718	0.2424	0.6292
GSA	35.9480	0.8779	0.0368	1.9257	0.8005	0.4235	0.0669	0.1411	0.8035
CNMF	39.3127	0.9299	0.0249	1.3325	0.8343	0.4864	0.0499	0.1143	0.8422
GFPCA	37.9443	0.9204	0.0314	1.5604	0.8032	0.3236	0.0898	0.1815	0.7445
PNN	43.1208	0.9704	0.0172	0.8528	0.9400	0.7390	0.0387	0.1162	0.8494
PANNet	43.0659	0.9685	0.0178	0.8577	0.9402	0.7309	0.0369	0.1219	0.8455
MSDCNN	45.6874	0.9827	0.0135	0.6389	0.9526	0.7759	0.0368	0.1112	0.8560
SRPPNN	47.1998	0.9877	0.0106	0.5586	0.9564	0.7900	0.0364	0.1087	0.8588
GPPNN	44.2145	0.9815	0.0137	0.7361	0.9510	0.7721	0.0360	0.1005	0.8669
Ours	47.4165	0.9892	0.0102	0.5472	0.9601	0.8031	0.0327	0.0999	0.8706

TABLE II

QUANTITATIVE COMPARISON OF NINE METHODS ON THE WORDVIEW II DATASETS. THE BEST, SECOND BEST, AND THIRD BEST RESULTS ARE HIGHLIGHTED BY RED, BLUE, AND UNDERLINE, RESPECTIVELY. ↑ INDICATES THAT THE LARGER THE VALUE, THE BETTER THE PERFORMANCE, AND ↓ INDICATES THAT THE SMALLER THE VALUE, THE BETTER THE PERFORMANCE

Methods	PSNR ↑	SSIM ↑	SAM ↓	ERGAS ↓	SCC ↑	Q ↑	D_λ ↓	D_S ↓	QNR ↑
Brovey	35.8646	0.9216	0.0403	1.8238	0.8913	0.6163	0.0770	0.1360	0.7977
IHS	35.2962	0.9027	0.0461	2.0278	0.8534	0.5704	0.0774	0.1578	0.7770
SFIM	34.1297	0.8975	0.0439	2.3449	0.9079	0.6064	0.0915	0.1277	0.7942
Wavelet	34.9827	0.8806	0.0481	2.0907	0.8752	0.5489	0.1102	0.1701	0.7396
GSA	36.3574	0.9219	0.0397	1.7401	0.9313	0.6506	0.0616	0.1144	0.8320
CNMF	37.0400	0.9374	0.0354	1.5741	0.9383	0.6636	0.0621	0.1137	0.8325
GFPCA	34.5580	0.9038	0.0488	2.1400	0.8924	0.4665	0.1016	0.1656	0.7508
PNN	40.7550	0.9624	0.0259	1.0646	0.9677	0.7426	0.0650	0.1186	0.8250
PANNet	40.8176	0.9626	0.0257	1.0557	0.9680	0.7437	0.0645	0.1189	0.8252
MSDCNN	41.3355	0.9664	0.0242	0.9940	0.9721	0.7577	0.0635	0.1172	0.8276
SRPPNN	41.4538	0.9679	0.0233	0.9899	0.9729	0.7691	0.0637	0.1164	0.8281
GPPNN	41.1622	0.9684	0.0244	1.0315	0.9722	0.7627	0.0642	0.1163	0.8278
Ours	41.8228	0.9694	0.0227	0.9291	0.9750	0.7734	0.0653	0.1151	0.8272

B. Comparison With State-of-the-Art Methods

To prove the effectiveness of our proposed method, we select seven typical traditional algorithms and five advanced models based on deep learning in recent years. The traditional algorithms are the Brovey [21], the IHS, the SFIM, the Wavelet, the GSA, the CNMF, and GFPCA, and the deep-learning-based algorithms are the PNN [12], the PANNet [32], the MSDCNN [33], the SRPPNN [2], and the GPPNN. We conduct several experiments on GF2, WV2, and WV3 datasets between our own model and the selected model to compare the performance differences between each algorithm.

Table I shows the average performance of multiple experiments for each method that we compared on GaoFen2 satellite images, where the top three results are marked in red, blue, and underlined, respectively. As can be seen from Table I, the performance of the deep learning model is generally better than that of traditional algorithms, and our method achieves the best performance in all indicators compared with previous algorithms. The model proposed by us not only achieves the optimal performance but also is far less than the previous best models in terms of the number of parameters and computation.

TABLE III

QUANTITATIVE COMPARISON OF NINE METHODS ON THE WORDVIEW III DATASETS. THE BEST, SECOND BEST, AND THIRD BEST RESULTS ARE HIGHLIGHTED BY RED, BLUE, AND UNDERLINE, RESPECTIVELY. ↑ INDICATES THAT THE LARGER THE VALUE, THE BETTER THE PERFORMANCE, AND ↓ INDICATES THAT THE SMALLER THE VALUE, THE BETTER THE PERFORMANCE

Methods	PSNR ↑	SSIM ↑	SAM ↓	ERGAS ↓	SCC ↑	Q ↑	D_λ ↓	D_S ↓	QNR ↑
Brovey	22.5060	0.5466	0.1159	8.2331	0.7033	0.4394	0.0481	0.2006	0.7603
IHS	22.5579	0.5354	0.1266	8.3616	0.6994	0.4301	0.0356	0.2073	0.7634
SFIM	21.8212	0.5457	0.1208	8.9730	0.6952	0.4531	0.0448	0.1265	0.8347
Wavelet	21.8551	0.5216	0.1368	9.1158	0.6823	0.4356	0.0883	0.1892	0.7401
GSA	21.8845	0.5458	0.1394	9.0781	0.7111	0.4615	0.0460	0.2373	0.7279
CNMF	22.0585	0.5569	0.1194	8.8117	0.7064	0.4534	0.0461	0.1991	0.7640
GFPCA	22.3344	0.4826	0.1294	8.3964	0.6987	0.3115	0.0528	0.1214	0.8325
PNN	29.9418	0.9121	0.0824	3.3206	0.9540	0.8679	0.0460	0.0933	0.8654
PANNet	29.6840	0.9072	0.0851	3.4263	0.9512	0.8631	0.0474	0.0942	0.8634
MSDCNN	30.3038	0.9184	0.0782	3.1884	0.9577	0.8763	0.0432	0.0877	0.8732
SRPPNN	30.4346	0.9202	0.0770	3.1553	0.9581	0.8776	0.0414	0.0909	0.8719
GPPNN	30.1785	0.9175	0.0776	3.2593	0.9568	0.8739	0.0438	0.0936	0.8671
Ours	30.5949	0.9227	0.0755	3.0751	0.9608	0.8834	0.0412	0.0893	0.8739

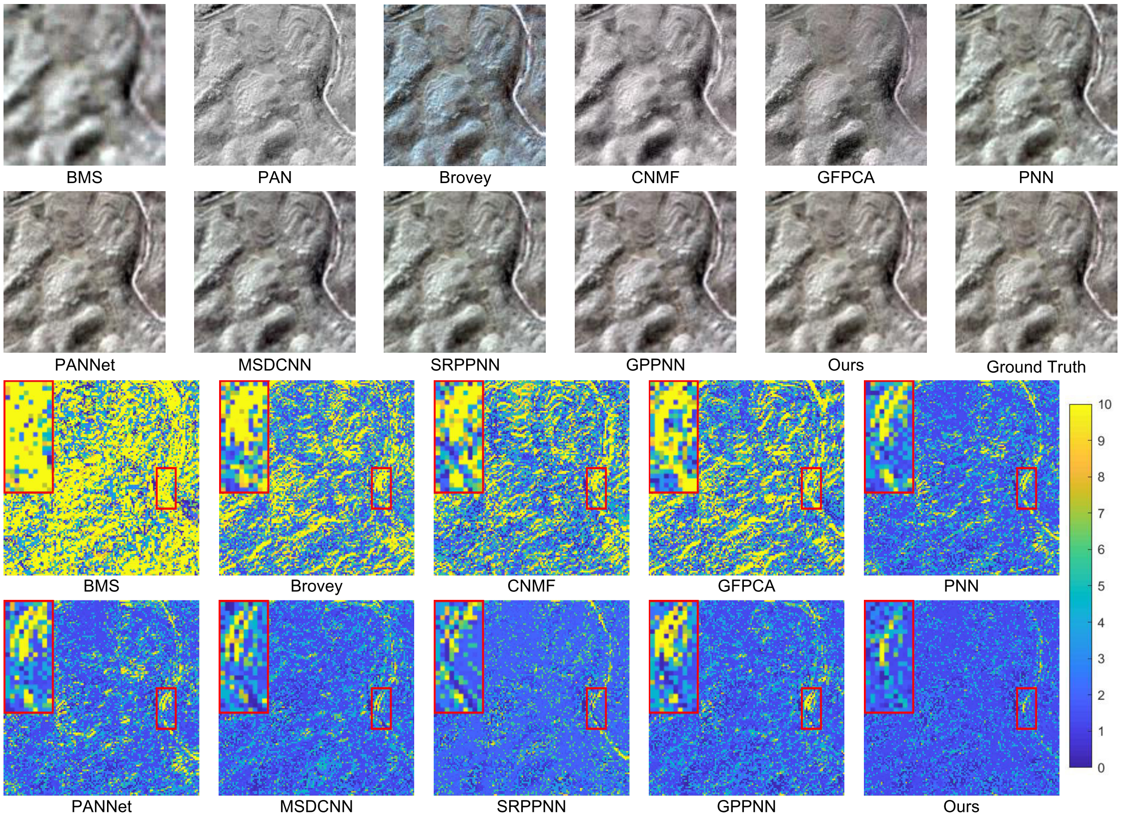

Fig. 6. Qualitative comparison of GCPNet with eight counterparts on a typical satellite image pair from the GF2 dataset. Images in the last two rows visualize the MSE between the pansharpened results and the ground truth. (Please zoom in to see more details.)

As shown in Table IV, our model uses only 0.08 M of memory but has a better performance in all aspects than the SRPPNN model with 1.7 M of memory. In addition, our model also has an advantage in computation, which is nearly fifteen times faster than SRPPNN. Of course, when compared with PANNet and PNN with a similar number of parameters, GCPNet's performance is much better than these models. In order to further verify the generalization ability of the model, we conduct further experiments on two datasets WV2 and WV3. From the comparison results in Tables II and III,

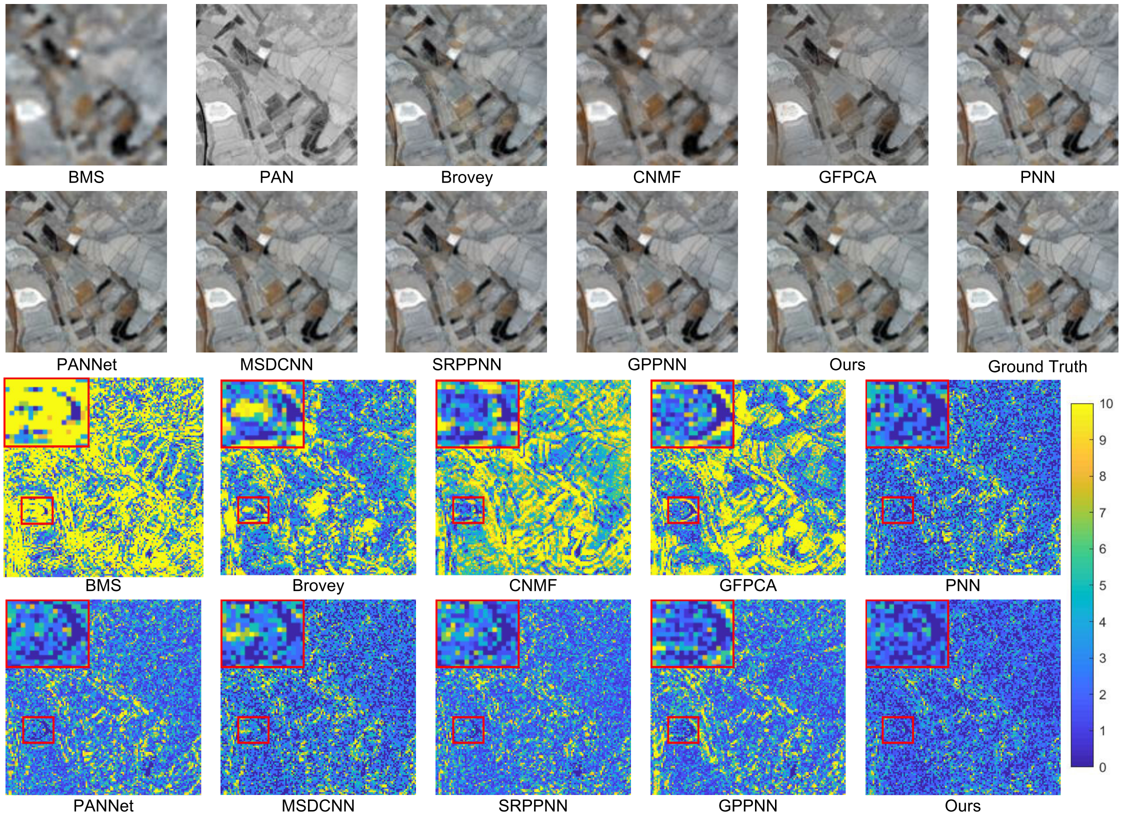

Fig. 7. Qualitative comparison of GCPNet with eight counterparts on a typical satellite image pair from the WV2 dataset. Images in the last two rows visualize the MSE between the pansharpened results and the ground truth. (Please zoom in to see more details.)

TABLE IV
COMPARISON OF PARAMETERS AND FLOATING-POINT OPERATIONS OF DEEP LEARNING MODEL

Moldes	PNN	PANNet	MSDCNN	SRPPNN	GPPNN	**Ours**
Params	0.689	0.688	2.390	17.114	1.198	**0.867**
FLOPs	1.129	1.127	3.916	21.106	1.397	**1.417**

the GCPNet proposed by us is obviously superior to other algorithms compared in all reference evaluation indexes and most reference-free measure indexes, which means that the model can effectively maintain spatial details and avoid spectral distortion.

In order to more intuitively understand the differences between various algorithms, we select one sample from each of the three datasets and display the pansharpening results. In Figs. 6–8, we only show the results of the image composed of three bands for visualization purposes, but all spectral bands are considered quantitatively in our evaluation. To highlight the differences in detail, the last two lines show the MSE pictures computed between pansharpening results and the ground truth, where the closer the color is yellow, the result in this area is quite different from the real situation.

As shown in Fig. 6, most of the GF2 dataset is of large-scale mountains, and the problems after satellite image sharpening are more obvious in the spectrum. Brovey, PNN, PANNet, and GPPNN all have a significant spectral distortion in a large area, but our model GCPNet can keep consistent with the spectrum of ground truth. From the gullies in the MSE picture, we can see that the results of GCPNet show the smallest area of yellow, which means that GCPNet can well restore the original spatial features. Most objects in the WV2 dataset are ponds and fields, while most objects in the WV3 dataset are buildings and vehicles. Therefore, the objects in Figs. 7 and 8 are smaller, and the spectral distortion of each part is not uniform, unlike the large-scale spectral distortion in Fig. 6, but the restoration of details is more prone to error. In general, it can be concluded from the MSE pictures that our model that gives consideration to both global and local features can well process images of objects of different scales and achieves the best performance as its results are the closest to the ground truth.

C. Experiments on Real Full-Resolution Images

In this section, in order to demonstrate the GCPNet's generalization capability in real full-resolution images, we further perform experiments on 200 sets of full-resolution data obtained by GaoFen2 that has not been used during the training stage. Because the ground-truth MS images in the real-world full-resolution scenes are not available, we follow the [64], [65] to adopt D_λ, D_S, and QNR for evaluation.

The quantitative comparison between the representative methods and GCPNet is shown in Table V. As can be seen clearly in Table V, although the SRPPNN and GPPNN

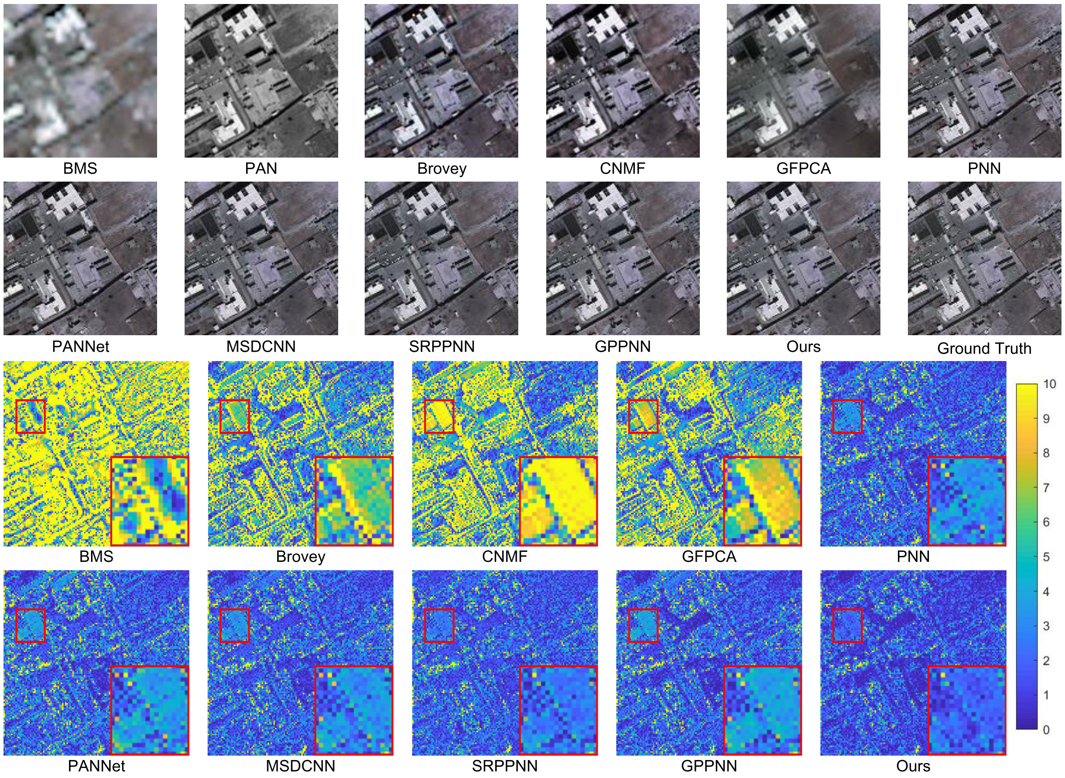

Fig. 8. Qualitative comparison of GCPNet with eight counterparts on a typical satellite image pair from the WV3 dataset. Images in the last two rows visualize the MSE between the pansharpened results and the ground truth. (Please zoom in to see more details.)

TABLE V

NONREFERENCE METRICS ON REAL FULL-RESOLUTION GF2 DATASETS. THE BEST, SECOND BEST, AND THIRD BEST RESULTS ARE HIGHLIGHTED BY RED, BLUE, AND UNDERLINE, RESPECTIVELY. ↑ INDICATES THAT THE LARGER THE VALUE, THE BETTER THE PERFORMANCE, AND ↓ INDICATES THAT THE SMALLER THE VALUE, THE BETTER THE PERFORMANCE

Metric	Brovey	IHS	Wavelet	GSA	GFPCA	PNN	PANNet	MSDCNN	SRPPNN	GPPNN	Ours
$D_\lambda \downarrow$	0.1378	0.0770	0.1782	0.1236	0.0914	0.0746	0.0737	0.0734	0.0767	0.0782	0.0723
$D_S \downarrow$	0.2605	0.2985	0.2027	0.2911	0.1635	0.1164	0.1224	0.1151	0.1162	0.1253	0.1144
QNR ↑	0.6390	0.6485	0.6602	0.6280	0.7615	0.8191	0.8143	0.8215	0.8173	0.8073	0.8265

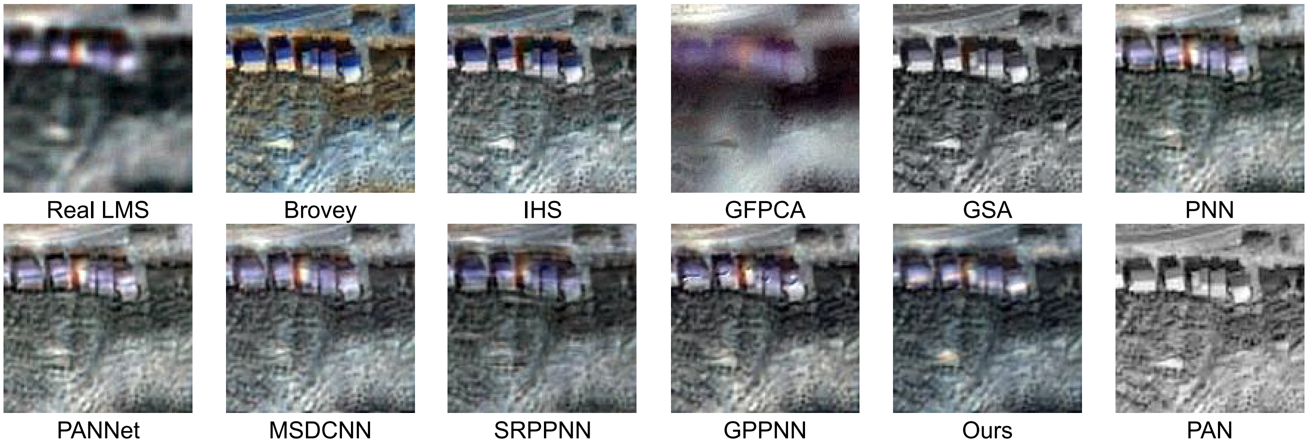

Fig. 9. Qualitative comparison of GCPNet with nine counterparts on a real full-resolution satellite image pair from the GF2. Note that the Real LMS is zoomed in for visualization. (Please zoom in to see more details.)

have a good performance on the simulated datasets, they perform poorly on real full-resolution images. On the contrary, our models can still surpass other competitive pansharpening methods in all the indexes. For qualitative evaluation, the visual results obtained by different methods for the real full-resolution images are depicted in Fig. 9. The images produced

TABLE VI

ABLATION STUDIES. ↑ INDICATES THAT THE LARGER THE VALUE, THE BETTER THE PERFORMANCE, AND ↓ INDICATES THAT THE SMALLER THE VALUE, THE BETTER THE PERFORMANCE

Baseline1	SGCN	BGCN	Asy Dstill	PSNR ↑	SSIM ↑	SAM ↓	ERGAS ↓	SCC ↑	Q ↑
✓	✗	✗	✗	41.5794	0.9682	0.0238	0.9785	0.9732	0.7679
✓	✗	✗	✓	41.6428	0.9686	0.0235	0.9772	0.9736	0.7683
✓	✓	✗	✗	41.6477	0.9690	0.0233	0.9417	0.9745	0.7708
✓	✗	✓	✗	41.5956	0.9693	0.0229	0.9615	0.9745	0.7703
✓	✓	✓	✗	41.6823	0.9689	0.0229	0.9344	0.9747	0.7721
✓	✓	✓	✓	**41.8228**	**0.9694**	**0.0227**	**0.9291**	**0.9750**	**0.7734**

SGCN: Spatial GCN Module
BGCN: Spectral Band GCN Module
Asy Dstill: Asynchronous Knowledge Distillation

by traditional methods have obvious distortion either in the spectrum or in space. As PNN and PANNet do not take the fusion of spectral information into account well, the images they generated have obvious spectral distortion compared with the real LMS. Moreover, the images of SRPPNN and GPPNN have no spectral distortion that can be clearly perceived, but they have artifacts in spatial details compared with the PAN image. The effect of MSDCNN and our model seems to be the best. Through comparison, we can notice that our model is better than MSDCNN in spectral reconstruction.

D. Ablation Studies

In this section, the ablation experiments are performed to verify the effectiveness of the proposed method SGCN, spectral BGCN, ASPM, and the method of asynchronous knowledge distillation. We replace SGCN and BGCN modules in the network structure with resblock, which is used as the basic unit in SRPPNN, as our baseline1 network. Because the resblock that was first presented by He *et al.* [66] has a similar residual connection structure to our module, it is widely used in computer vision tasks because of its good versatility. Table VI lists four variants of the proposed approach and the average evaluation results for each variant. The SGCN and BGCN components and the asynchronous knowledge distillation method will be analyzed in detail in the following.

1) Validation on SGCN: Graph convolution can aggregate information between pixels and has the expression ability of global space. SGCN is used to extract the correlation between pixels and obtain the global receptive field. To confirm the effectiveness of this SGCN module, we compare the model with SCGN to the baseline1 model. As shown in the third line of Table VI, the pansharpening performance of the model is improved when the GCN module is added to baseline1. SCC that estimates the spatial correlation coefficient is significantly improved when SGCN is adopted. These results indicate that SGCN is beneficial to the expression of spatial features.

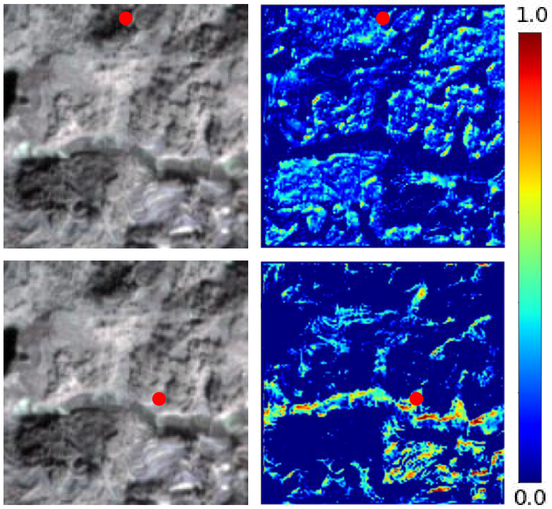

Fig. 10. Visualization of long-range correlation. Given one pixel depicted by the red dot, the SGCN module can attend to all pixels. The pictures in the right column show the depicted weights of the nodes on the whole graph to the selected pixel. (Best viewed in color.)

2) Validation on BGCN: To verify the contribution of the proposed BGCN to the extraction of spectral information, in Table VI, we list the results of the baseline1 method with and without BGCN. We notice that the addition of BGCN reduces the PSNR but can improve the SAM. In addition, when BGCN is added at the same time as SGCN, the value of SAM does not increase, and other indicators also improved. It shows that SGCN and BGCN modules are not mutually exclusive and can be used together in the model to achieve better performance.

3) Effectiveness of Asynchronous Knowledge Distillation: In order to verify the effectiveness of the learning strategy proposed, we not only compare the method on GCPNet but also conduct experiments on our baseline1 model. In Table VI, the addition of asynchronous knowledge distillation improves the performance of both models, indicating that asynchronous knowledge distillation can improve the representational ability of models and, thus, bring better performance.

In Fig. 10, we show a visual result about the long-range correlation captured by the SGCN. Given one pixel from the

TABLE VII

ABLATION STUDIES. ↑ INDICATES THAT THE LARGER THE VALUE, THE BETTER THE PERFORMANCE, AND ↓ INDICATES THAT THE SMALLER THE VALUE, THE BETTER THE PERFORMANCE

Baseline2	PPM	ASPP	ASPM	PSNR ↑	SSIM ↑	SAM ↓	ERGAS ↓	SCC ↑	Q ↑
✓	✗	✗	✗	41.4221	0.9677	0.0239	0.9795	0.9733	0.7661
✓	✓	✗	✗	41.3519	0.9670	0.0240	0.9869	0.9722	0.7582
✓	✗	✓	✗	41.5915	0.9677	0.0236	0.9522	0.9731	0.7665
✓	✗	✗	✓	**41.6823**	**0.9689**	**0.0229**	**0.9344**	**0.9747**	**0.7721**

PPM: Pyramid Pooling Module
ASPP: Atrous Spatial Pyramid Pooling
ASPM: Atrous Spatial Pyramid Module

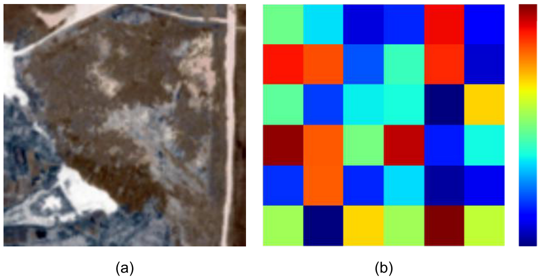

Fig. 11. Visualization of the adjacency matrix in BGCN. (a) Input image. (b) Adjacency matrix. The redder the color, the bigger the value.

input feature, the depicted weights would be the response intensity that is the correlation obtained by calculating the similarity between the feature of the selected pixel and the features of each location at the whole graph in the feature dimension. To be specific, we select one pixel from the output of $\varphi(\mathbf{F}_{\text{in}})\theta(\mathbf{F}_{\text{in}})^{\mathsf{T}}\delta(\mathbf{F}_{\text{in}})$ whose feature dimension is $(N/2)$ and then calculate the similarity matrix to obtain the correlation between nodes. To the best of our knowledge, after the aggregation operation with the adjacency matrix, the more correlated nodes will have more similar features. The correlation between nodes is distinguished by different colors. The redder the color is, the stronger the correlation among the pixels in this region and the selected pixel is. As expected, a single pixel can also aggregate long-range information. Since BGCN's adjacency matrix is smaller, it is more convenient to display, so we visualize the adjacency matrix learned from data to show the long-range correlation between the spectra in Fig. 11. It can be seen from the adjacency matrix that each node can more or less aggregate the information from long-range nodes.

In addition, to verify the advantages of our ASPM over the approach mentioned in Figs. 4, we conduct another separate comparative experiment. We replace the ASPM module in GCPNet with the resblock module as our baseline2 network. It can be seen from Table VII that the model performance decreases when PPM is added, which just validates the negative impact of PPM's downsampling and upsampling operations on image recovery. In the last two rows of Table VII, the performance of ASPM is superior to that of ASPP. We believe that this is because the ASPM combining serial and parallel connections has better nonlinear expression capability than the ASPP that only connects in parallel, which is also mentioned by Simonyan and Zisserman [67].

V. CONCLUSION

In this study, we innovatively propose an encoder–decoder network based on graph convolution for pansharpening. The network structure can further improve the performance of pansharpening combined with the method of asynchronous knowledge distillation. Comparing the experimental results on multiple datasets, it can be concluded that our GCPNet has a very efficient pansharpening ability because it achieves state-of-the-art performance and has faster inference speed and better parameter efficiency than previous models.

The proposed GCPNet includes SGCN, spectral BGCN, and ASPM. The SGCN and BGCN use graph convolution to capture global spatial and spectral context information, respectively. In order to enhance the multiscale capability of model adaptation and avoid the loss of detail caused by downsampling operations, we design the ASPM. In addition to the well-designed network structure, our asynchronous knowledge distillation method enables the teacher network to learn additional information from high-resolution multispectral images that the student network cannot obtain and effectively transfers knowledge representation from the teacher to the student.

After numerous experiments, we believe that the proposed graph convolution method for pansharpening exhibits state-of-the-art performance. At the same time, we creatively provide a new way of using graph convolution for pansharpening, and its excellent performance proves that this approach is worthy of more research. Without more consideration of the effects of parameter number and computation, the potential performance improvement of this method can be further investigated.

REFERENCES

[1] H. Shen, X. Meng, and L. Zhang, "An integrated framework for the spatio–temporal–spectral fusion of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7135–7148, Sep. 2016.

[2] J. Cai and B. Huang, "Super-resolution-guided progressive pansharpening based on a deep convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 6, pp. 5206–5220, Jun. 2021.

[3] D. Hong *et al.*, "More diverse means better: Multimodal deep learning meets remote-sensing imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4340–4354, Aug. 2020.

[4] D. Hong, J. Yao, D. Meng, Z. Xu, and J. Chanussot, "Multimodal GANs: Toward crossmodal hyperspectral–multispectral image segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 6, pp. 5103–5113, Jun. 2021.

[5] P. Shamsolmoali, M. Zareapoor, J. Chanussot, H. Zhou, and J. Yang, "Rotation equivariant feature image pyramid network for object detection in optical remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, 2022.

[6] X. Wu, D. Hong, P. Ghamisi, W. Li, and R. Tao, "LW-ODF: A lightweight object detection framework for optical remote sensing imagery," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2019, pp. 1462–1465.

[7] X. Wu, D. Hong, J. Tian, J. Chanussot, W. Li, and R. Tao, "ORSIm detector: A novel object detection framework in optical remote sensing imagery using spatial-frequency channel features," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 5146–5158, Jul. 2019.

[8] X. Lv, D. Ming, Y. Chen, and M. Wang, "Very high resolution remote sensing image classification with SEEDS-CNN and scale effect analysis for superpixel CNN classification," *Int. J. Remote Sens.*, vol. 40, no. 2, pp. 506–531, 2018.

[9] Z. Shao, H. Fu, D. Li, O. Altan, and T. Cheng, "Remote sensing monitoring of multi-scale watersheds impermeability for urban hydrological evaluation," *Remote Sens. Environ.*, vol. 232, Oct. 2019, Art. no. 111338.

[10] F. Bovolo, L. Bruzzone, L. Capobianco, A. Garzelli, S. Marchesi, and F. Nencini, "Analysis of the effects of pansharpening in change detection on VHR images," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 1, pp. 53–57, Jan. 2010.

[11] K. Zhang, F. Zhang, and S. Yang, "Fusion of multispectral and panchromatic images via spatial weighted neighbor embedding," *Remote Sens.*, vol. 11, no. 5, p. 557, Mar. 2019.

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

[13] Y. Qu, R. K. Baghbaderani, H. Qi, and C. Kwan, "Unsupervised pansharpening based on self-attention mechanism," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 4, pp. 3192–3208, Apr. 2021.

[14] M. Ehlers, "Multisensor image fusion techniques in remote sensing," *ISPRS J. Photogramm. Remote Sens.*, vol. 46, no. 1, pp. 19–30, 1991.

[15] G. Piella, "Image fusion for enhanced visualization: A variational approach," *Int. J. Comput. Vis.*, vol. 83, no. 1, pp. 1–11, Jun. 2009.

[16] X. X. Zhu and R. Bamler, "A sparse image fusion algorithm with application to pan-sharpening," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 5, pp. 2827–2836, May 2013.

[17] W. J. Carper, T. M. Lillesand, and R. W. Kiefer, "The use of intensity-hue-saturation transformations for merging SPOT panchromatic and multispectral image data," *Photogramm. Eng. Remote Sens.*, vol. 56, no. 4, pp. 459–467, Apr. 1990.

[18] P. Kwarteng and A. Chavez, "Extracting spectral contrast in Landsat thematic mapper image data using selective principal component analysis," *Photogramm. Eng. remote Sens.*, vol. 55, nos. 339–348, p. 1, 1989.

[19] V. P. Shah, N. H. Younan, and R. L. King, "An efficient pan-sharpening method via a combined adaptive PCA approach and contourlets," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 5, pp. 1323–1335, May 2008.

[20] C. A. Laben and B. V. Brower, "Process for enhancing the spatial resolution of multispectral imagery using pan-sharpening," U.S. Patent 6011875, Jan. 4, 2000.

[21] A. R. Gillespie, A. B. Kahle, and R. E. Walker, "Color enhancement of highly correlated images. II. Channel ratio and 'chromaticity' transformation techniques," *Remote Sens. Environ.*, vol. 22, no. 3, pp. 343–365, 1987.

[22] F. DadrasJavan, F. Samadzadegan, and F. Fathollahi, "Spectral and spatial quality assessment of IHS and wavelet based pan-sharpening techniques for high resolution satellite imagery," *Eur. J. Appl. Sci.*, vol. 6, no. 2, p. 1, 2018.

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

[24] A. L. da Cunha, J. Zhou, and M. N. Do, "The nonsubsampled contourlet transform: Theory, design, and applications," *IEEE Trans. Image Process.*, vol. 15, no. 10, pp. 3089–3101, Oct. 2006.

[25] J. Ma and G. Plonka, "Computing with curvelets: From image processing to turbulent flows," *Computing Sci. Eng.*, vol. 11, no. 2, pp. 72–80, Mar. 2009.

[26] T. Uezato, D. Hong, N. Yokoya, and W. He, "Guided deep decoder: Unsupervised image pair fusion," in *Proc. Eur. Conf. Comput. Vis.*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Cham, Switzerland: Springer, 2020, pp. 87–102.

[27] D. Hong, J. Chanussot, and X. X. Zhu, "An overview of multimodal remote sensing data fusion: From image to feature, from shallow to deep," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. IGARSS*, Jul. 2021, pp. 1245–1248.

[28] M. Zhou, X. Fu, J. Huang, F. Zhao, A. Liu, and R. Wang, "Effective pan-sharpening with transformer and invertible neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–15, 2022.

[29] X. Liu, C. Deng, J. Chanussot, D. Hong, and B. Zhao, "StfNet: A two-stream convolutional neural network for spatiotemporal image fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6552–6564, Sep. 2019.

[30] J. Zhong, B. Yang, G. Huang, F. Zhong, and Z. Chen, "Remote sensing image fusion with convolutional neural network," *Sens. Imag.*, vol. 17, no. 1, p. 10, 2016.

[31] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, Feb. 2015.

[32] 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.*, Oct. 2017, pp. 5449–5457.

[33] Q. Yuan, Y. Wei, X. Meng, H. Shen, and L. Zhang, "A multiscale and multidepth convolutional neural network for remote sensing imagery pan-sharpening," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 3, pp. 978–989, Mar. 2018.

[34] S. Xu, J. Zhang, Z. Zhao, K. Sun, J. Liu, and C. Zhang, "Deep gradient projection networks for pan-sharpening," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 1366–1375.

[35] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61–80, Jan. 2009.

[36] J. Bruna, W. Zaremba, A. Szlam, and Y. Lecun, "Spectral networks and locally connected networks on graphs," in *Proc. Int. Conf. Learn. Represent.*, Apr. 2014, pp. 183–196.

[37] L. Zhao, X. Peng, Y. Tian, M. Kapadia, and D. N. Metaxas, "Semantic graph convolutional networks for 3D human pose regression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3420–3430.

[38] X. Li, Y. Yang, Q. Zhao, T. Shen, Z. Lin, and H. Liu, "Spatial pyramid based graph reasoning for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8947–8956.

[39] X. Qi, R. Liao, J. Jia, S. Fidler, and R. Urtasun, "3D graph neural networks for RGBD semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5209–5218.

[40] G. Wadhwa, A. Dhall, S. Murala, and U. Tariq, "Hyperrealistic image inpainting with hypergraphs," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 3912–3921.

[41] X. Fu, Q. Qi, Z. Zha, Y. Zhu, and X. Ding, "Rain streak removal via dual graph convolutional network," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 1352–1360.

[42] D. Valsesia, G. Fracastoro, and E. Magli, "Image denoising with graph-convolutional neural networks," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 2399–2403.

[43] L. Zhang, X. Li, A. Arnab, K. Yang, Y. Tong, and P. H. S. Torr, "Dual graph convolutional network for semantic segmentation," 2019, *arXiv:1909.06121*.

[44] W. Gedara Chaminda Bandara, J. Maria Jose Valanarasu, and V. M. Patel, "SPIN road mapper: Extracting roads from aerial images via spatial and interaction space graph reasoning for autonomous driving," 2021, *arXiv:2109.07701*.

[45] A. Qin, Z. Shang, J. Tian, Y. Wang, T. Zhang, and Y. Y. Tang, "Spectral-spatial graph convolutional networks for semisupervised hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 2, pp. 241–245, Sep. 2019.

[46] S. Wan, C. Gong, P. Zhong, B. Du, L. Zhang, and J. Yang, "Multiscale dynamic graph convolutional network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3162–3177, May 2020.

[47] D. Hong, L. Gao, J. Yao, B. Zhang, A. Plaza, and J. Chanussot, "Graph convolutional networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 5966–5978, Jul. 2021.

[48] D. Hong, L. Gao, X. Wu, J. Yao, and B. Zhang, "Revisiting graph convolutional networks with mini-batch sampling for hyperspectral image classification," in *Proc. 11th Workshop Hyperspectral Imag. Signal Process., Evol. Remote Sens. (WHISPERS)*, Mar. 2021, pp. 1–5.

[49] J. Kang, R. Fernandez-Beltran, D. Hong, J. Chanussot, and A. Plaza, "Graph relation network: Modeling relations between scenes for multilabel remote-sensing image classification and retrieval," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4355–4369, May 2021.

[50] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015, *arXiv:1503.02531*.

[51] N. C. Garcia, P. Morerio, and V. Murino, "Modality distillation with multiple stream networks for action recognition," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 106–121.

[52] W. Lee, J. Lee, D. Kim, and B. Ham, "Learning with privileged information for efficient image super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, vol. 12369, 2020, pp. 465–482.

[53] H. Wu, J. Liu, Y. Xie, Y. Qu, and L. Ma, "Knowledge transfer dehazing network for NonHomogeneous dehazing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 1975–1983.

[54] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1163–1176.

[55] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7794–7803.

[56] Y. Chen, Y. Kalantidis, J. Li, S. Yan, and J. Feng, "A^2-Nets: Double attention networks," in *Proc. Conf. Neural Inf. Process. Syst.*, 2018, pp. 350–359.

[57] Y. Chen, M. Rohrbach, Z. Yan, Y. Shuicheng, J. Feng, and Y. Kalantidis, "Graph-based global reasoning networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 433–442.

[58] H. Zhang, Z.-J. Zha, Y. Yang, S. Yan, and T.-S. Chua, "Robust (semi) nonnegative graph embedding," *IEEE Trans. Image Process.*, vol. 23, no. 7, pp. 2996–3012, Jul. 2014.

[59] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 6230–6239.

[60] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2017.

[61] J. Hoffman, S. Gupta, and T. Darrell, "Learning with side information through modality hallucination," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 826–834.

[62] J. Lee, S. Seo, and M. Kim, "SIPSA-net: Shift-invariant pan sharpening with moving object alignment for satellite imagery," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 10161–10169.

[63] 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. JPL, Summaries 3rd Annu. JPL Airborne Geosci. Workshop*, 1992, pp. 147–149.

[64] Y. Zheng, J. Li, Y. Li, G. Jie, X. Wu, and J. Chanussot, "Hyperspectral pansharpening using deep prior and dual attention residual network," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 11, pp. 8059–8076, Nov. 2020.

[65] W. G. C. Bandara, J. M. J. Valanarasu, and V. M. Patel, "Hyperspectral pansharpening based on improved deep image prior and residual reconstruction," 2021, *arXiv:2107.02630*.

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

[67] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–14.

Keyu Yan received the B.E. degree in electronic and information engineering from the Dalian University of Technology, Dalian, China, in 2021.

He is pursuing the M.S. degree with the University of Science and Technology of China, Hefei, China. His research field includes embedded systems and applications, signal processing, and computer vision.

Man Zhou is pursuing the Ph.D. degree with the University of Science and Technology of China, Hefei, China.

His research interests include image/video processing and computer vision.

Liu Liu received the B.S. degree from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2015, the M.S. degree from the University of Manchester, Manchester, U.K., in 2016, and the Ph.D. degree from the University of Science and Technology of China, Hefei, China, in 2020.

He is a Post-Doctoral Researcher with Shanghai Jiao Tong University, Shanghai, China. His research interests include computer vision, deep learning, and embodied AI.

Chengjun Xie received the M.S. degree in software engineering from the Hefei University of Technology, Hefei, China, in 2008, and the Ph.D. degree from the Hefei University of Technology, Anhui, China, in 2014.

He is working with the Institute of Intelligent Machinery, Chinese Academy of Sciences, Beijing, as an Associate Researcher. His research interests include image processing, machine learning, and pattern recognition.

Danfeng Hong received the M.Sc. degree *(summa cum laude)* in computer vision from the College of Information Engineering, Qingdao University, Qingdao, China, in 2015, and the Dr.-Ing. degree *(summa cum laude)* from the Signal Processing in Earth Observation (SiPEO), Technical University of Munich (TUM), Munich, Germany, in 2019.

He is a Professor with the Key Laboratory of Computational Optical Imaging Technology, Aerospace Information Research Institute, Chinese Academy of Sciences (CAS). Before joining CAS, he has been a Research Scientist and led a Spectral Vision Working Group at the Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Oberpfaffenhofen, Germany. He was also an Adjunct Scientist at GIPSA-lab, Grenoble INP, CNRS, Universite Grenoble Alpes, Grenoble, France. His research interests include signal/image processing, hyperspectral remote sensing, machine/deep learning, artificial intelligence, and their applications in Earth Vision.

Dr. Hong is a Topical Associate Editor of the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING (TGRS), an Editorial Board Member of *Remote Sensing*, and an Editorial Advisory Board Member of the *ISPRS Journal of Photogrammetry and Remote Sensing*. He was a recipient of the Best Reviewer Award of the IEEE TGRS in 2021 and 2022, the Best Reviewer Award of the IEEE JSTARS in 2022, the Jose Bioucas Dias Award for recognizing the Outstanding Paper at WHISPERS in 2021, the Remote Sensing Young Investigator Award in 2022, and the IEEE GRSS Early Career Award in 2022.