Pansharpening via Detail Guided and Global Scale Convolution

Weisheng Li, *Member, IEEE*, Xudong Zhi, Yidong Peng, and Yijian Hu

***Abstract*—Pansharpening is a crucial step in various remote sensing tasks, aimed at generating high-resolution multispectral images from panchromatic and low-resolution multispectral images. While deep learning has shown promising results in improving the accuracy of pansharpening, previous models often enhanced accuracy by stacking a large number of trainable parameters, making model training and application challenging. In this article, we propose a pansharpening network based on detail guided and global scale convolution, which can balance the parameter quantity of the model and its accuracy. Specifically, our model utilizes the global convolutional neural network (GCNN) module, which has favorable time complexity and, to some extent, alleviates issues such as insufficient receptive fields and excessive compression of long-distance information found in traditional convolutional neural networks. GCNN enables our model to capture global information effectively. In addition, we introduce a detail guided residual learning module that uses high-resolution image information to enhance details and compensate for the loss of high-frequency information during forward propagation. Furthermore, we design a lightweight convolutional module named channel aggregation learning that utilizes partial convolution for efficient interaction of interchannel feature information. Moreover, we introduce fast Fourier transform loss in the loss function to capture frequency domain information loss, further improving model performance. Extensive experiments on multiple datasets demonstrate the effectiveness of our proposed method.**

***Index Terms*—Cross scale convolution, detail guided, image fusion, pansharpening, remote sensing.**

I. Introduction

NOWADAYS, pansharpening is widely utilized in numerous remote sensing tasks because it enables the provision of high-resolution multispectral (HR-MS) images, allowing image information to encompass both spatial and spectral information. Consequently, these tasks employ pansharpening to obtain HR-MS images, which are subsequently used for corresponding remote sensing tasks, such as target detection [1], [2], [3] and semantic segmentation [4], [5] as well as practical applications such as digital mapping and agriculture [6]. However, due to technological and physical constraints, remote sensing satellites are unable to directly acquire HR-MS images. Hence, we need to employ deep learning (DL) to strike a balance between spatial and spectral details, enabling the model to fuse panchromatic (PAN) and low-resolution multispectral (LR-MS) images to generate HR-MS images, a process known as pansharpening.

In the past several decades, many effective studies and models have been dedicated to better integrating the information of LR-MS images and PAN images to achieve the desired HR-MS images for applications. According to different fusion methods, they can be roughly categorized into four main types: component substitution (CS), multiresolution analysis (MRA), variational optimization (VO), and DL [7], [8], [9]. The following is a brief overview of the four methods.

Based on CS algorithms, the upsampled LR-MS image is typically transformed into a feature space to separate its spatial components, which are then replaced by the PAN image to fill in the missing spatial details. Well-known CS algorithms include methods utilizing the intensity-hue-saturation [10] transform, principal component analysis [11], and band-dependent spatial detail [12].

MRA-based methods typically decompose the LR-MS image into multiple resolutions, extract spatial detail information using multiscale analysis, and replace it with the PAN image rich in high-frequency detail information. Representative MRA methods include wavelet transform [13], [14], generalized Laplacian pyramid [15], and intensity modulation based on smooth filtering [16].

VO-based methods construct an appropriate mathematical model with suitable regularization terms based on prior knowledge or assumptions. The advantage of VO methods lies in their ability to flexibly establish models according to task requirements. Typical VO methods include Bayesian-based fusion methods [17], establishing probabilistic models, and utilizing sparse representation theory to inject PAN image detail information into LR-MS images [18].

The above three methods are traditional learning approaches. Among them, methods based on MRA are prone to spatial distortion, leading to the loss of detailed texture information. Methods based on CS have poor spectral fidelity and are susceptible to spectral distortion. Methods based on VO have relatively high computational costs and low processing speeds.

DL techniques have also been widely applied in the field of remote sensing. By learning features and patterns from a large

Manuscript received 1 May 2024; revised 25 June 2024 and 28 July 2024; accepted 8 August 2024. Date of publication 16 August 2024; date of current version 5 September 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62331008, Grant 62027827, Grant 62221005, and Grant 62276040, in part by the Natural Science Foundation of Chongqing under Grant 2023NSCQ-LZX0045 and Grant CSTB2022NSCQ-MSX0436, and in part by the Special Funding for Postdoctoral Research Projects of Chongqing under Grant 2022CQBSHTB3103. *(Corresponding authors: Weisheng Li; Yidong Peng.)*

The authors are with the Chongqing Key Laboratory of Image Cognition, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: liws@cqupt.edu.cn; S230201159@stu.cqupt.edu.cn; pengyd@cqupt.edu.cn; S220231037@stu.cqupt.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2024.3444003

amount of remote sensing image data, they can better extract useful information, thus achieving significant results in various remote sensing applications. In the task of pansharpening, inspired by image super-resolution methods, Masi et al. [19] designed pansharpening by convolutional neural networks (PNN) with a three-layer convolutional structure and achieved satisfactory results in the field of pansharpening. PanNet, proposed by Yang et al. [20], directly upsamples LR-MS images and adds them to mixed feature images extracted from the high-pass filtering domain to achieve spectral fidelity and spatial detail injection. Yuan et al. [21] used multiscale convolution to extract receptive fields to cope with different sizes of remote sensing targets. They used multiscale residual blocks to extract deep-level features and shallow-level features for summation (MSDCNN), achieving superior performance. Jin et al. [22] proposed a convolutional neural network that combines local and global contextual information (LAGConv) for remote sensing and sharpening tasks. By considering the interaction of local and global information, this model achieves more accurate remote sensing and sharpening effects. Fan et al. [36] designed a DL model based on transformer for remote sensing and sharpening tasks, achieving fusion of PAN and LR-MS images. Based on observations from three following aspects:

1) The loss of detailed information during the propagation process, possibly due to the continuous aggregation of forward propagation information.
2) The interaction of complex channel information leads to a vast number of parameters and computational overhead.
3) Convolutional neural network (CNN)-based methods often face limitations due to their fixed receptive fields and information compression. In addition, they tend to focus more on local information, thereby losing the aggregation of long-distance information.

To address these issues, we propose a pansharpening network based on detail guided and global scale convolution (DGGSC) that integrates both global and local information. When modeling the problem of super-resolution, Zhou et al. [31] utilized high-resolution objects in images to guide the low-resolution objects, but for pansharpening tasks, we have access to high-resolution PAN images. Therefore, we can utilize a shared encoder to extract features from the PAN images, which naturally possess high resolution and clear texture details. We use the PAN image features to refine the feature information during the forward propagation process and increase feature differences using power functions, enabling the model to more easily capture differential information to correct for detail loss during forward propagation. The shared encoder reduces the number of parameters and enhances the perception of feature information at different levels for the model, making the model more generalized. In addition, to address the limitations of CNNs in shallow networks due to fixed receptive fields, information compression, and the inability to obtain long-range information, we designed a lighter global convolutional neural network (GCNN) module. This module ensures effective aggregation of long-range information, enabling better integration of global information. Furthermore, the GCNN effectively addresses issues with information compression during information propagation in CNNs. In addition, it enlarges the receptive field of the model and accelerates the transfer of feature information, which is very beneficial for the transfer of features when the number of model layers is small.

In addition, to enable better interaction of information in the channel dimension, we utilized partial convolution [24] (PConv) and pointwise convolution to design an efficient and lightweight channel aggregation learning (CAL) module. In the channel fusion module, efficient integration and interaction of channel information have been achieved to improve the accuracy of the model.

In summary, our contributions are as follows.

1) We designed a parameter-lightweight pansharpening model that can effectively reduce the parameter quantity of the pansharpening model while ensuring accuracy.
2) We introduced a GCNN module to expand the receptive field and alleviate the overcompression of local information. By incorporating a detail guided residual learning (DRL) module, we compensate for the high-frequency detail loss that may occur due to information aggregation during forward propagation.
3) An efficient CAL module has been designed to facilitate effective interaction of information between channels, ensuring model performance.

The rest of this article is organized as follows. Section II reviews related works. Section III provides a detailed explanation of the proposed method. Section IV presents the experimental results and analysis. Section V conducted ablation studies on the model. Finally, Section VI concludes this article.

II. Related Work

A. Residual-Based Injection DL Methods

Residual injection networks are widely used in DL, which is an improvement on traditional aggregation-type networks (PNN) [19]. They are roughly divided into two categories.

The first type concatenates the PAN image with the upsampled LR-MS image along the channel dimension, extracts the fused feature information through DL, and adds it to the upsampled LR-MS image. He et al. [25] proposed an end-to-end DL architecture for pansharpening based on image detail injection. The process of concatenation-type models is as follows:

$$H_x = \hat{L_x} + V_x \cdot \text{Concat}(P,\ L^{\uparrow}) \tag{1}$$

where H_x represents the xth band of the HR-MS image. $\hat{L} \in \mathbb{R}^{H\times W\times B}$ denotes the upsampled LR-MS image at the PAN scale, where H and W denote the height and width of the PAN image, respectively. $\hat{L_x}$ represents the xth band of the upsampled LR-MS image. V_x denotes the injection gain matrix. $P \in \mathbb{R}^{H\times W\times 1}$ is the PAN image. $L \in \mathbb{R}^{h\times w\times B}$ denotes the LR-MS image, where h and w are the height and width of the LR-MS image, respectively. For pansharpening, $H = h \times 4$ and $W = w \times 4$. B is the number of LR-MS bands. $L^{\uparrow}$ represents the upsampled LR-MS image, with the same upsampling method as $\hat{L}$.

The second type of model involves first obtaining the difference information between the PAN image and the LR-MS

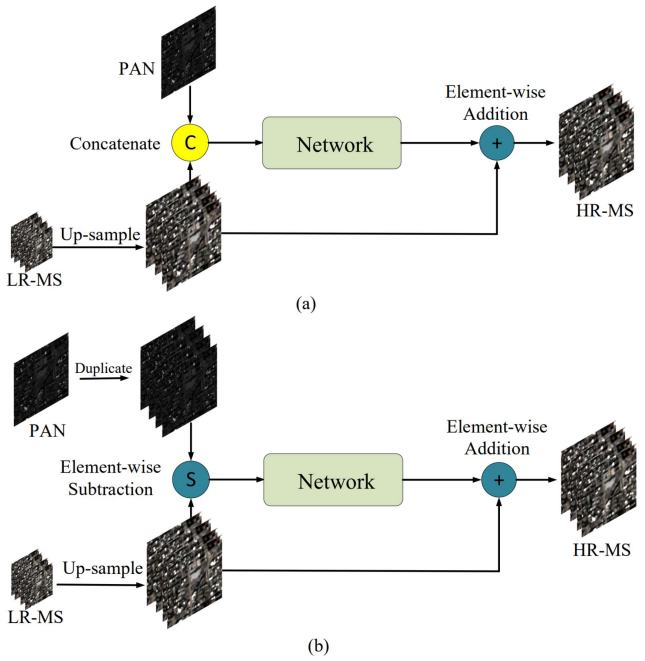


Fig. 1. Two residual network architectures differ in their approach to merging PAN and LR-MS images. (a) Network architecture based on concatenation. (b) Network architecture based on difference.

image through differencing. This difference information is then subjected to feature extraction and injected into the upsampled LR-MS image. Because it focuses solely on the different information, it often achieves better results compared to directly concatenating the PAN image and the LR-MS image. Deng et al. [26] proposed a novel pansharpening fusion network architecture (FusionNet), which effectively preserves the spatial information and potential spectral information of the image by directly subtracting the original upsampled LR-MS image from the PAN image to extract details. Jin et al. [22] proposed a convolutional module called LAGConv that adapts to local content and dynamically generates convolutional kernels using difference feature maps for modeling. The resulting model can quickly and efficiently fit the training data. The architecture of differencing-type model is roughly as follows:

$$H_x = \hat{L_x} + G_x \cdot (\hat{P} - L^{\uparrow}) \tag{2}$$

where G_x represents the nonlinear mapping of the model and $\hat{P}$ denotes the duplication of PAN image in the band channel. As shown in Fig. 1, the two aforementioned residual structures are demonstrated. G_x and $L^{\uparrow}$ can be obtained through different DL methods or traditional learning methods. Therefore, the choice of calculation method is particularly important.

B. Long Range Dependency Methods

Vision transformer (ViT) [27] is the first model to introduce transformer into images, which segments the image into patches and embeds them into a sequence of vectors. While ensuring the extraction of global information, it greatly reduces the time complexity. At the same time, it allows images to undergo attention mechanisms by splitting patches. With the success of ViT in the field of vision, many efforts have been made to apply ViT to high-resolution image processing. Meng et al. [28] applied ViT to handle pansharpening tasks, where they concatenated the upsampled LR-MS image with the PAN image, then extracted feature information through ViT to reconstruct patches, and finally concatenated all patches together to obtain the HR-MS image. Yin et al. [29], based on the transformer structure, proposed a local and nonlocal feature interaction network that continuously interacts with information during the forward propagation process to improve performance. They also utilized transformer to extract global feature information and employed multiscale convolutional modules to enhance the model's perception of remote sensing objects of different sizes, achieving superior performance. Li et al. [30] proposed a dual-branch multiscale network for pansharpening. They utilized transformer structures to model spatial and spectral information separately, then concatenated their features and input them into an image reconstruction module to generate HR-MS images. Fan et al. [36] proposed a multiscale embedding and dual attention transformers (MDPNet) that embeds the multiscale information of the image into vectors, thereby making more efficient use of the multiscale information. They introduced the additive hybrid attention transformer fusion module and the channel self-attention transformer detail generation module, which improved the fusion ability of the model.

C. Super Resolution Methods

An image is worth graph of nodes (VisionGNN) [23] is the first application of a graph convolution network in visual images. It divides the image into patches and extracts features through the embed operation, then converts them into feature vectors. It then constructs a graph network based on the similarity between vectors. Inspired by VisionGNN, cross-scale internal graph neural network [31] models the super-resolution problem by guiding low-resolution pixels with high-resolution pixels, which allows for better detail guidance and, consequently, achieves superior super-resolution results. Spatially-adaptive feature modulation for efficient image super-resolution [33] enhances the performance of super-resolution models by employing various downsampling rates. The use of different sampling rates has two advantages: On one hand, it enables the model to capture feature information at different scales, and on the other hand, it allows the model to obtain a larger receptive field without increasing computational load.

III. Proposed Methods

A. Overall Network Architecture

The overall architecture of the proposed DGGSC is depicted in Fig. 2, which consists of three main layers. The layers, namely, DRL, GCNN, and multiscale and multichannel learning (MSM), begin by taking the difference between the PAN image and the upsampled LR-MS image. Subsequently, they extract shallow features using a feature encoder. The process is as follows:

$$f_0 = \text{Convs}(\hat{P} - \hat{L}) \tag{3}$$

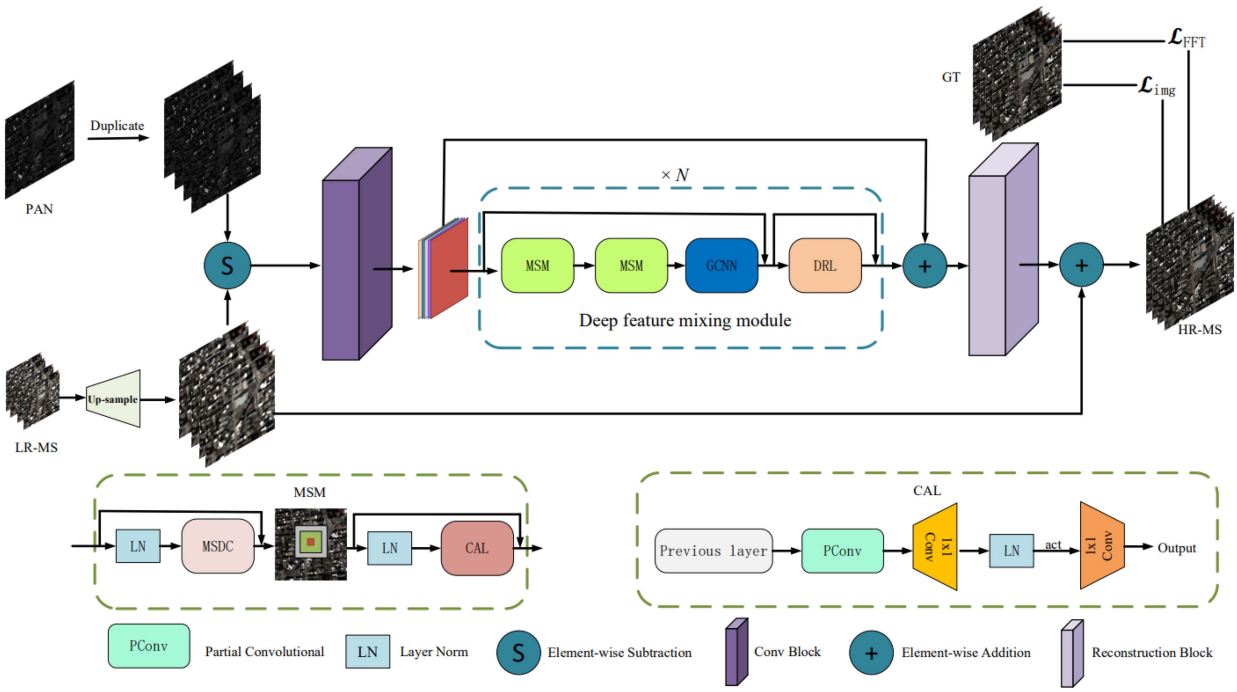


Fig. 2. Overview of the proposed pansharpening framework DGGSC, which consists of several fundamental building blocks. Each block includes four core designs: multiscale feature detection and capture layer (MSDC), CAL, GCNN, and DRL.

where $\hat{P}$ denotes the duplication of PAN image in the band channel. Convs$(\cdot)$ denotes the shallow feature encoder. Afterward, the model passes through three important modules DRL, GCNN, and MSM. Among them, MSM downsamples the image to obtain a wider receptive field and distinguishes different scales of remote sensing ground objects through different downsampling sizes. Subsequently, the model interacts with channel dimension features through a lightweight channel aggregation module. Then, the model obtains long-range scale information through GCNN, which alleviates the problem of overcompression of distant information to some extent. Finally, DRL is used to compensate for the loss of feature information due to forward propagation. It corrects feature information at each layer to ensure the proper propagation of features. The forward process of the model can be represented as follows:

$$\hat{f}_i = \text{GCNN}(\text{MSM}(\text{MSM}(f_{i-1}))) + f_{i-1} \tag{4}$$

$$f_i = \text{DRL}(\hat{f}_i) + \hat{f}_i \tag{5}$$

where f_i represents the output of the ith layer of the model. After deep feature extraction, it undergoes residual processing with the initial features and image reconstruction. The reconstructed image is then directly added to the upsampled LR-MS image. The process is shown as follows:

$$H_m = \text{Restruct}(f_n + f_0) + \hat{L} \tag{6}$$

where f_n represents the output features of the last layer network, which have the same shape as f_0, and Restruct$(\cdot)$ represents the reconstruction module, composed of several stacked convolutions. H_m represents the HR-MS images.

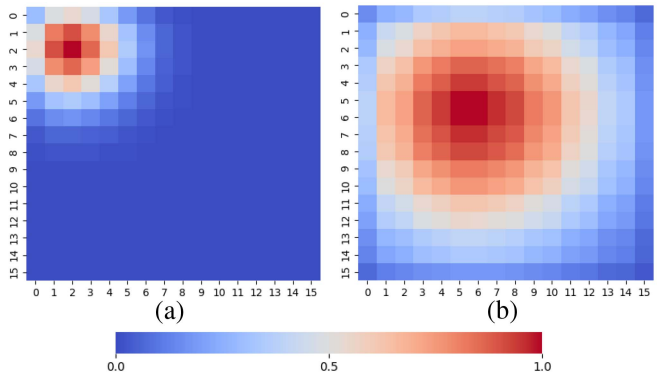


Fig. 3. Comparison of feature propagation in (a) traditional convolution and (b) GCNN. The more red the color, the greater the degree of information aggregation, whereas the more blue the color, the smaller the degree of information aggregation.

B. GCNN Module

In traditional CNN-based pansharpening networks, there are two main issues. First, using conventional convolutional kernels of size 3 can not achieve a sufficiently large receptive field. Solutions to this problem include replacing traditional convolutions with dilated convolutions, downsampling feature maps, or increasing the size of convolutional kernels. Second, traditional convolutions are prone to overcompressing surrounding pixels, which limits the transmission performance of long-range feature information.

As shown in Fig. 3, first, traditional convolution exhibits a rapid decline in perceptual capability as pixels move away from the convolution center. Second, feature information used for

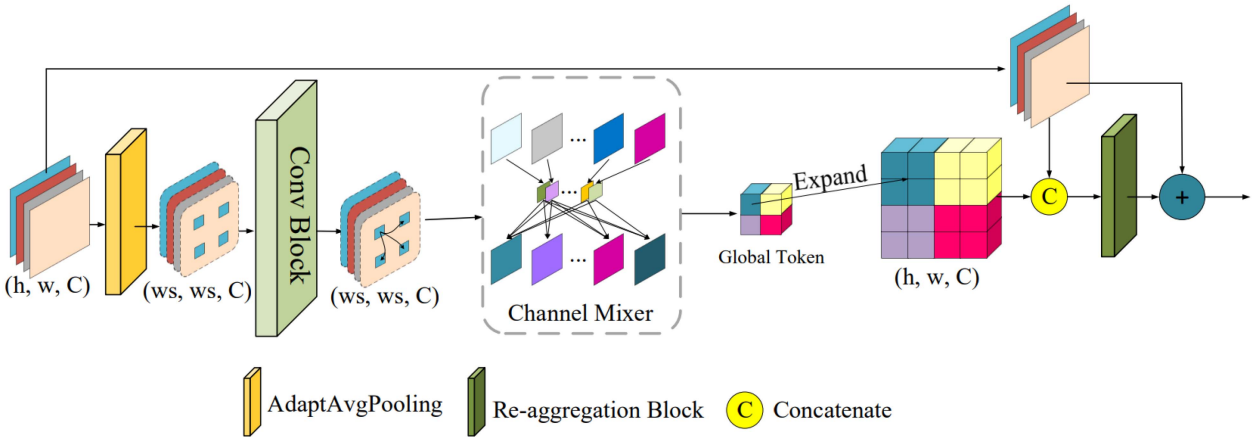


Fig. 4. Illustration of the GCNN module.

aggregation often tends to favor surrounding pixels, and this restricts the transmission of long-distance information. To address these two issues, we designed a GCNN module. Compared with the traditional CNN, this module can effectively increase the receptive field of the CNN, and the feature information can break through the limitation centered on itself and aggregate information to farther regions, alleviating the problem of excessive information compression caused by the CNN to a certain extent. As illustrated in Fig. 3, when GCNN and CNN perform feature aggregation at (2, 2), it can be observed that GCNN has a larger receptive field compared to CNN. Meanwhile, CNN tends to aggregate features based on its immediate surroundings, whereas GCNN can aggregate information not centered on itself but from more distant areas, which is beneficial to the transmission of long-range information.

The specific structure of GCNN is shown in Fig. 4. It can be seen that GCNN first partitions the feature information into windows, then features interact in space and channels to obtain global biased information, and finally reaggregates the obtained biased information with the original feature information.

GCNN first performs average pooling on the input feature map to transform it to the window size (ws). This operation can be described by the following:

$$X_{\text{pool}} = \text{AvgPool}(V_c) \tag{7}$$

where V_c represents the input feature map, and $\text{AvgPool}(\cdot)$ is the AdaptAvgPooling operation. The feature maps after pooling will pass through the feature fusion module. First, the features will undergo convolution with a large kernel size, during which the convolution will aggregate global bias information. Then, the output feature map will undergo convolution with a small kernel size, which will mix feature information in the channel dimension of the feature map. The above process can be described by the following:

$$X_f = \text{CMixer}(\text{GC}_{5\times5}(X_{\text{pool}})) \tag{8}$$

where $\text{GC}_{5\times5}(\cdot)$ represents group convolution with a kernel size of 5, and $\text{CMixer}(\cdot)$ refers to a channel mixer, representing a convolution operation with a kernel size of 1. X_f represents the output feature map.

Finally, we expand X_f in the spatial dimension to the same size as the original feature map, obtaining global bias information. We concatenate the biased information with the original information and aggregate it through convolution. The process can be represented by the following:

$$\hat{X} = \text{RAG}(\text{Concat}(V_c, \Uparrow_{\frac{H}{\text{ws}}}(X_f))) + V_c \tag{9}$$

where $\text{RAG}(\cdot)$ refers to a reaggregation block, representing a convolution with a kernel size of 1, ws represents the window size, $\Uparrow_{\frac{H}{\text{ws}}}(\cdot)$ denotes the feature map expanded by a factor of $\frac{H}{\text{ws}}$, $\text{Concat}(\cdot)$ represents feature map concatenation along the channel dimension and $\hat{X}$ represents the output feature map.

For efficient modeling, we proposed a new forward propagation module of the CNN. This module enables the model to capture global information by dividing windows, which can expand the receptive field of the model to a certain extent and alleviate the problem of overcompression of information caused by the aggregation of surrounding features. And the perception degree of global biased information can be controlled by controlling the size of ws. Let M be the size of ws, W_f be the width and height of the input feature map of GCNN, and dim be the channel dimension of the input feature map of GCNN. Initially, the feature maps pass through a group convolution module with a kernel size of K_1, and its time complexity is $2K_1^2\text{dim}M^2$. Subsequently, the output feature maps are processed by a channel mixer, which has a time complexity of 2dim^2M^2. Finally, the original feature maps are concatenated with the feature maps output by the channel mixer and passed through a reaggregation module to obtain the final output results, as its time complexity is $2\text{dim}^2W_f^2$. So the time complexity of the GCNN module and the traditional CNN with a convolution kernel size of 3 on the feature map is as follows:

$$\Omega(\text{CNN}) = 9\text{dim}^2W_f^2 \tag{10}$$

$$\Omega(\text{GCNN}) = 2\left(\left(\frac{W_f}{M}\right)^2 + \frac{K_1^2}{\text{dim}} + 1\right)\text{dim}^2M^2 \tag{11}$$

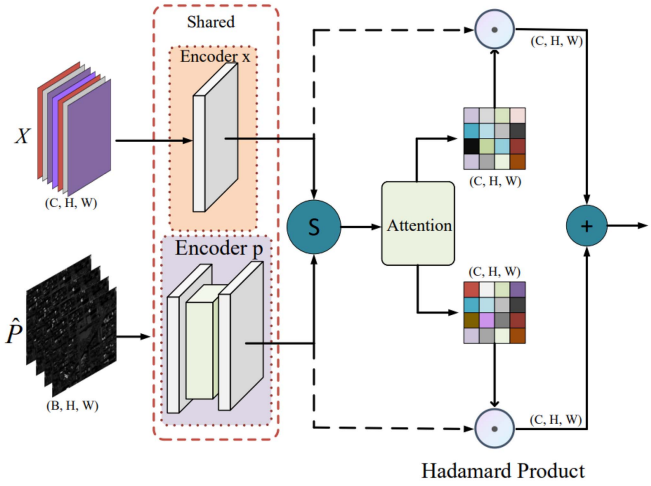


Fig. 5. Structure of DRL.

where K_1 represents the size of the large convolution kernel. Typically, $\frac{K_1^2}{\dim} \ll (\frac{W_f}{M})^2$, so the complexity of the GCNN module is approximately as follows:

$$\Omega(\text{GCNN}) \approx 2\dim^2 W_f^2. \tag{12}$$

C. DRL Module

1) Overall DRL Module Architecture: The main architecture of the module is shown in Fig. 5. We pass the feature vectors from the intermediate layers of the model and the PAN image through corresponding encoders, mapping them to a differential feature space. The feature encoder and PAN image encoder are designed as shared modules, and the parameters of these two components being involved in feature extraction across different layers. Difference operations are then employed to extract different information. This different information is used to derive attention weights through a focusing module. Finally, the feature information is reaggregated based on these attention weights.

2) Shared Encoder: The application of the graph convolution network proposed by Zhou et al. [31] in super-resolution aims to complete detailed texture features by guiding low-resolution pixels with high-resolution pixels. By comparing downsampling with the original image, the objects corresponding to the downsampling in the original image are the high-detail images.

The method proposed by Zhou et al. [31] has demonstrated excellent results in super-resolution. Therefore, we are considering applying their approach, which guides low-resolution pixels with high-resolution ones, to pansharpening tasks. In pansharpening, we utilize the feature information of high-resolution images to compensate for the important features lost by the model during forward propagation in order to improve the performance of the model. In the pansharpening task, we have high-resolution PAN images containing abundant detail and texture information. Therefore, we encode the PAN images through a feature encoder and similarly encode the intermediate layer features of the model through another feature encoder, mapping them into a feature differential space, so that we can compute their differences to obtain differential information. For these feature encoders, we designed them as a shared module. On the one hand, it can reduce the parameters of the model. On the other hand, it enables the encoders to capture information at different levels and improve its generalization ability. Its forward process can be expressed by the following:

$$X_p = \text{Conv}_{3\times3}(Act(\text{Conv}_{3\times3}(\hat{P}))) \tag{13}$$

$$X_e = \text{Conv}_{3\times3}(X) \tag{14}$$

where $Act(\cdot)$ represents the GELU activation function, X represents the input feature map to the DRL, the outputs X_p and X_e have the same shape, and $\text{Conv}_{3\times3}(\cdot)$ represents a convolution operation with a kernel size of 3.

3) Spatial Detail Attention: Han et al. [32] made the differences between features more obvious by applying the power operation to the feature vectors. Inspired by this, we introduce the power operation to enable the model to focus on the feature regions with a severe loss of detailed textures each time and reaggregate the features of these feature regions. This approach enables the model to increase the perception degree of feature differences and compensate for feature information more evenly. The module first computes the difference between X_e and X_p to obtain a feature map. It then amplifies the differences using a power function, allowing the model to better focus on differential information. Finally, an activation function is applied to obtain the attention matrix. The PAN image features and the features in the forward propagation process of the model are reaggregated to obtain the corrected feature map. The forward process can be described as follows:

$$X_{\text{delta}} = X_e - X_p \tag{15}$$

$$\hat{X}_{\text{delta}} = \frac{X_{\text{delta}}^q}{\text{Norm}(X_{\text{delta}}^q)}\text{Norm}(X_{\text{delta}}) \tag{16}$$

$$\text{attn} = \text{Sig}(\hat{X}_{\text{delta}}) \tag{17}$$

$$\bar{X} = \text{attn} \odot X_p + (I - \text{attn}) \odot X_e \tag{18}$$

where $\text{Norm}(\cdot)$ denotes the L_1 norm of the features. $\bar{X}$ represents the output feature map. $\text{Sig}(\cdot)$ represents the sigmoid activation function, $\odot$ denotes elementwise multiplication, I is a matrix of the same shape as attn with all elements equal to 1, and q is a model parameter, set to 3 in this article. $(\cdot)^q$ represents the operation of raising a vector to the power of q.

D. MSM Module

Compared to traditional large-kernel convolutional attention mechanisms, the method proposed by Sun et al. [33] is more lightweight and possesses multiscale feature representation capabilities. We apply it to pansharpening tasks and further introduce residual connections to enhance performance. The structure is depicted in Fig. 6. This module enables the model to capture multiscale feature information without increasing the number of parameters in the model. Specifically, the module first evenly divides the feature map along the channel dimension, then applies different downsampling rates to the divided feature

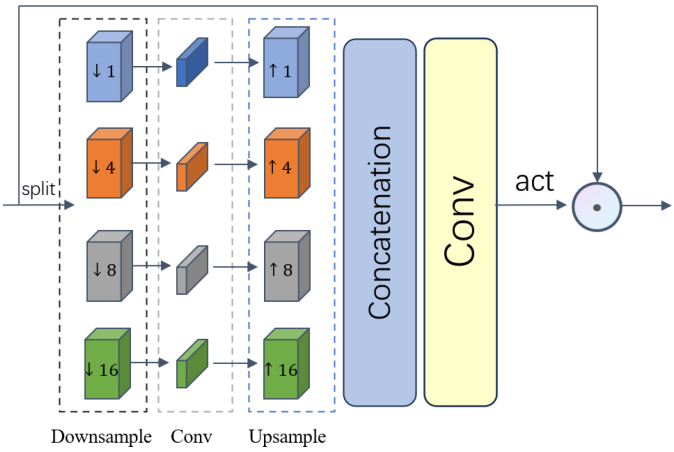


Fig. 6. Structure of MSDC.

maps to capture different scales of receptive fields, which is particularly beneficial for capturing objects of various sizes in remote sensing imagery. Then, the features are extracted at various scales through depthwise convolution (DWConv), and the output feature maps are obtained through upsampling. Given the input feature map V_m, the forward process of the module can be defined as follows:

$$[x_0, x_1, x_2, x_3] = \text{split}(V_m) \tag{19}$$

$$\hat{x}_0 = \text{DWConv}_{3\times3}(x_0) \tag{20}$$

$$\hat{x}_{i-1} = \uparrow_{2^i} (\text{DWConv}_{3\times3}(\downarrow_{2^i} (x_i))), 2 \le i \le 4 \tag{21}$$

where $\text{split}(\cdot)$ represents the operation of splitting along the channel dimension, $\text{DWConv}_{3\times3}(\cdot)$ denotes a depthwise convolution operation with a kernel size of 3, $\uparrow_{2^i}(\cdot)$ indicates upsampling the feature map by a factor of 2^i using nearest interpolation, and $\downarrow_{2^i}(\cdot)$ represents downsampling the feature map by a factor of 2^i.

Subsequently, the obtained multiscale feature maps are concatenated along the channel dimension, and information aggregation is performed on the concatenated feature maps through convolutional operations with a kernel size of 1. It can be described as follows:

$$\hat{Y} = \text{Conv}_{1\times1}(\text{Concat}([\hat{x}_0, \hat{x}_1, \hat{x}_2, \hat{x}_3])) \tag{22}$$

where $\text{Conv}_{1\times1}(\cdot)$ denotes convolutional operations with a kernel size of 1, $\text{Concat}(\cdot)$ represents the concatenation of feature maps along the channel dimension, and $\hat{Y}$ represents the output feature map.

Finally, the feature map $\hat{Y}$ is passed through an activation function to obtain a spatial attention matrix. This attention matrix is then multiplied elementwise with the input feature map V_m. The forward process can be represented by the following:

$$O = \Phi(\hat{Y}) \odot V_m \tag{23}$$

where $\Phi(\cdot)$ represents the GELU activation function, which converts the feature maps into a spatial attention matrix and $\odot$ denotes elementwise multiplication.

Only by performing spatial attention while ignoring the information interaction between channels may limit the performance of the model. To enable efficient interchannel information interaction, we designed a CAL module. This module has fewer parameters and requires less computation compared to regular inverted convolution blocks, while efficiently facilitating feature information interaction across channels.

The structure of CAL is illustrated in Fig. 2. The input feature map first undergoes preliminary feature extraction via PConv [24]. Because PConv only interacts with partial feature channels, its performance surpasses that of depthwise separable convolution. Subsequently, the output feature map passes through the channel aggregation block. At the core of the channel aggregation block is an inverted convolutional block with a kernel size of 1. The forward process of CAL, given the input feature V_l, is as follows:

$$X_o = \text{Conv}_{1\times1}(\text{Act}(\text{LN}(\text{Conv}_{1\times1}(\text{PConv}(V_l))))) \tag{24}$$

where $\text{LN}(\cdot)$ represents layer normalization, and $\text{Act}(\cdot)$ represents the GELU activation function. $\text{Conv}_{1\times1}(\cdot)$ denotes the convolutional operation with a kernel size of 1, and $\text{PConv}(\cdot)$ represents the partial convolution.

E. Loss Function

As shown in Fig. 2, we use two types of losses, including pixel reconstruction loss and frequency domain reconstruction loss, which are computed as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{img}} + \alpha\mathcal{L}_{\text{FFT}} \tag{25}$$

where α represents a hyperparameter used to balance the frequency domain loss and pixelwise loss. The fast Fourier transform (FFT) loss maps the predicted image and the ground truth (GT) image to the frequency domain and calculates the loss. We choose the L_1 loss in the frequency domain to reconstruct the frequency domain difference. The formula is as follows:

$$\mathcal{L}_{\text{img}} = ||H_m - \text{GT}||_1 \tag{26}$$

$$\mathcal{L}_{\text{FFT}} = ||\text{FFT}(H_m) - \text{FFT}(\text{GT})||_1 \tag{27}$$

where GT represents the GT labels and $\text{FFT}(\cdot)$ represents the Fourier transform of the image. H_m represents the HR-MS images.

IV. Experimental Results

A. Datasets

Our performance metrics on the DGGSC across multiple datasets, including WorldView3 (WV3), GaoFen2 (GF2), and QuickBird (QB), which contain remote sensing images with varying band sizes, test the generalization ability of the model. The spatial resolution of PAN images is four times higher than that of multispectral (MS) images. Table I details the information about the datasets. "SSI" in the table denotes the abbreviation of "spatial sampling interval."

The process of generating the training set typically consists of two steps. First, T patches of size $256 \times 256 \times 1$ and $64 \times 64 \times B$ are cropped from the original PAN and MS images obtained.

TABLE I
DETAILS OF THE DATASETS INCLUDED IN OUR EXPERIMENTS

Dataset	QB	GF2	WV3
PAN SSI	0.61	1	0.3
LR-MS SSI	2.44	4	1.2
Radiometric	11	10	11
Training data	17139	19809	9714
Validation data	1905	2201	1080

Then, utilizing the Wald protocol, the cropped image patches are filtered with the modulation transfer function specific to each satellite and subsequently downsampled using interpolation to obtain low-resolution scales of both PAN and MS images. The sizes of the low-resolution PAN and MS images are $64 \times 64 \times 1$ and $16 \times 16 \times B$, respectively. The MS image patches before downsampling are used as label data, with a size of $64 \times 64 \times B$. The generation process for the test set is identical to the training set. The half-resolution PAN and MS images are of sizes $256 \times 256 \times 1$ and $64 \times 64 \times B$, respectively, whereas the full-resolution PAN and MS images are of sizes $512 \times 512 \times 1$ and $256 \times 256 \times B$.

The QB dataset comprises LR-MS images with 4 spectral bands, the GF2 dataset has the same number of spectral bands in its LR-MS images as the QB dataset, and the WV3 dataset contains LR-MS images with 8 spectral bands.

B. Compared Methods and Quantitative Metrics

PNN [19], DiCNN [25], FusionNet [26], MSDCNN [21], LAGConv [22], ADKNet [34], BiMPan [35], and MDPNet [36] represent the eight DL models selected by us for comparison with the proposed method in order to validate the effectiveness of the proposed model. In addition, we also compare the proposed framework with three traditional algorithms: GSA [37], Wavelet [38], and Brovey [39]. For fair comparison, all DL-based models were retrained on a Windows system equipped with an NVIDIA RTX A6000 GPU, and all hyperparameters for DL are set to be the same.

We followed the fundamental research standards of pansharpening and selected the spectral angle mapper (SAM) [40], erreur relative globale adimensionnelle de synthèse (ERGAS), spatial correlation coefficient (SCC) [41], the structural similarity index measure (SSIM), and the peak signal-to-noise ratio (PSNR) as indicators for evaluating the quality of low-resolution images. For full-resolution assessment, we employed three no-reference indicators, including the quality with no-reference (QNR) [42] index, the spectral distortion D_λ index, and the spatial distortion D_s index.

C. Training Details

We implemented our DGGSC model using Python 4.9 and PyTorch 1.7 on the Windows operating system with an NVIDIA RTX A6000 GPU. We set the initial learning rate to 0.001, which is halved halfway through training. The model runs for 200 000 iterations with betas set to [0.9, 0.99] for the Adam optimizer. We use a minibatch size of 16.

In the GCNN module, our ws is set to 8. The kernel size for the large convolutional layer is set to 5, whereas for the small convolutional layer, it is set to 1. The loss function includes both L_1 loss and FFT loss between the fused image and the reference image. The balance factors for the FFT loss function and the L_1 loss function are set to 0.05 and 1, respectively. The size of the dimension is 32 in the QB dataset and 36 in the GF2 and WV3 datasets.

The number of modules N is set to 4, and the exponent of the power q is set to 3.

D. Performance on the Reduced Resolution Dataset

In this section, we will compare the performance of the traditional learning and DL models selected on the three datasets we have chosen. Each dataset contains 20 MS images and their corresponding PAN images, where the spatial size of PAN images is four times that of MS images and the channel size of MS images is band times that of PAN images. The band varies according to different datasets, it is 4 for the QB dataset and the GF2 dataset, and 8 for the WV3 dataset. The original PAN images and MS images will be divided into smaller images for training, and the original images will be used as GT images to validate the performance metrics of the models.

1) Performance on the QB Dataset: We evaluate the performance of all methods on the QB dataset. Table II gives the quantitative evaluation results of various methods on the QB dataset. In the table, the average value is used to summarize the model performance, and the best results are marked in bold. It can be seen from the observation that the DL method shows excellent average indicators compared with the traditional method. The performance of our model is better than that of other DL models in all indicators, highlighting our superiority in preserving spectral information as well as spatial information. The superior performance metrics of DL compared to traditional learning methods demonstrate the superiority of DL.

In order to visually compare the performance of different models on the QB dataset, we sampled a pair of MS and PAN images from the test set and fed them into both traditional and DL models. As shown in Fig. 7, the first three images are the fusion results from traditional learning, followed by nine fusion results from DL models, with the last image being the GT label. It can be observed from the images that the fusion results from traditional learning exhibit noticeable color and detail discrepancies compared to the reference image, whereas the DL methods show less severe physical distortions. This suggests that DL performs better on the QB dataset. In addition, it can be observed that the method we proposed achieves better spatial resolution and spectral information.

Furthermore, in order to make the fusion results of the model more intuitive, we generated residual images between the fused images and the GT images, as illustrated in Fig. 8. Ideally, residual images should tend towards zero, with the richness of residual image content corresponding to the degree of blurring in the fused image. In DL methods, our residual images contain less content compared to other methods, indicating that our model preserves both texture and spectral information of the fused images well.

TABLE II
MODEL PERFORMANCE ON THE REDUCED RESOLUTION QB TEST SET

Method	SAM	PSNR	ERGAS	SCC	SSIM
GSA	8.6210	31.2454	8.6010	0.9194	0.8317
Wavelet	9.7863	30.1107	9.6276	0.8698	0.7896
Brovey	8.9705	29.7741	10.2067	0.8837	0.7901
FusionNet	4.9245	37.4325	4.1001	0.9819	0.9543
PNN	5.6613	36.2081	4.7712	0.9741	0.9395
MSDCNN	4.9188	37.5443	4.0894	0.9822	0.9546
ADKNet	4.9746	37.7931	3.9649	0.9804	0.9564
DiCNN	5.4581	35.7412	5.0140	0.9749	0.9367
LAGConv	4.5992	38.3040	3.7319	0.9854	0.9618
BiMPan	4.9372	37.3411	4.2366	0.9844	0.9568
MDPNet	4.9082	37.7149	3.9662	0.9814	0.9560
Ours	**4.5445**	**38.4731**	**3.6485**	**0.9864**	**0.9631**
Ideal Value	0	∞	0	1	1

The bold values represent the best evaluation indicators of the comparison methods.

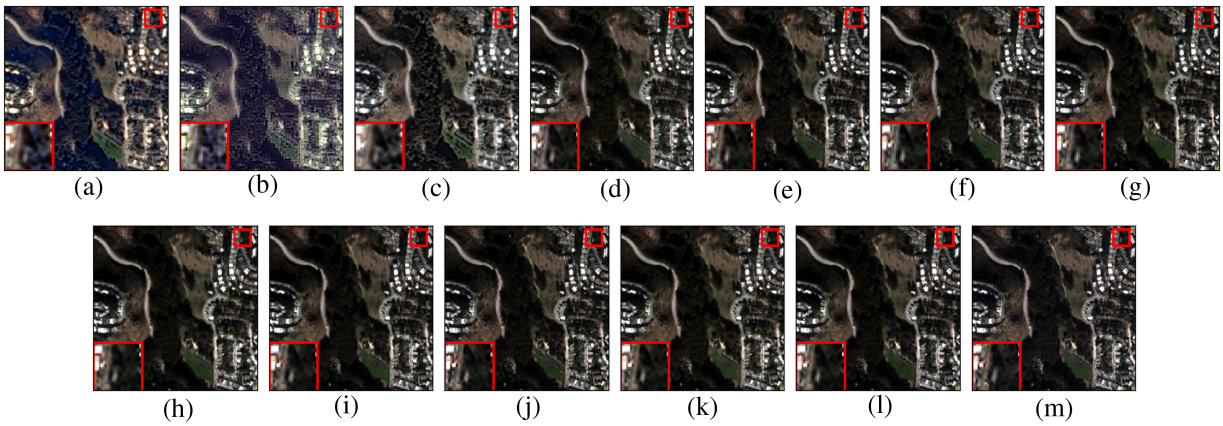


Fig. 7. Visual representation of the HR-MS image corresponding to the model on the QB dataset is generated using GT images as reference. (a) Brovey. (b) Wavelet. (c) GSA. (d) PNN. (e) FusionNet. (f) MSDCNN. (g) ADKNet. (h) DiCNN. (i) LAGConv. (j) BiMPan. (k) MDPNet. (l) Ours. (m) GT.

TABLE III
MODEL PERFORMANCE ON THE REDUCED RESOLUTION GF2 TEST SET

Method	SAM	PSNR	ERGAS	SCC	SSIM
GSA	1.8986	34.2154	1.8420	0.9553	0.8874
Wavelet	7.1080	29.5765	6.1915	0.8960	0.8068
Brovey	2.0615	31.3477	2.4126	0.8844	0.8683
FusionNet	0.9772	39.7755	0.9713	0.9901	0.9655
PNN	1.2292	37.4261	1.2686	0.9827	0.9454
MSDCNN	0.9752	40.1415	0.9345	0.9899	0.9668
ADKNet	0.9127	41.4166	0.8001	0.9914	0.9744
DiCNN	1.0679	38.9561	1.0700	0.9874	0.9598
LAGConv	0.8120	42.4446	0.7154	0.9934	0.9799
BiMPan	1.0330	40.4616	0.8901	0.9892	0.9693
MDPNet	0.8863	41.3579	0.8014	0.9910	0.9749
Ours	**0.7668**	**42.8902**	**0.6715**	**0.9935**	**0.9818**
Ideal Value	0	∞	0	1	1

2) Performance on the GF2 Dataset: This section presents the performance of all selected methods and the DGGSC model on the GF2 reduced resolution dataset. The GF2 dataset contains 20 PAN images and the corresponding 20 LR-MS images.

The evaluation results of the selected traditional learning methods and DL models on the test set are shown in Table III, and the best results are marked in bold. All DL-based methods show better results than traditional learning. It is worth noting that the quantitative results of the method DGGSC proposed in this article on the GF2 reduced resolution test set are superior to other models, and it shows performance beyond other models in SAM, PSNR, ERGAS, SCC, and SSIM. This indicates the superiority of the proposed model.

In addition, to provide a more intuitive visual validation of the performance, Fig. 9 displays the enlarged details of the HR-MS images within the red rectangles. It can be observed

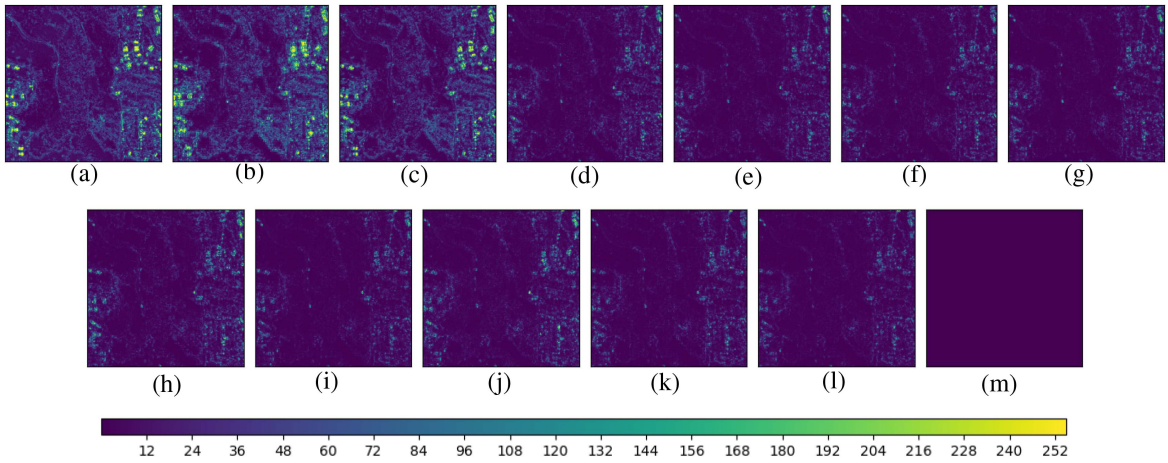


Fig. 8. Visual representation of the corresponding residual image of the model on the QB dataset is generated using GT images as reference. (a) Brovey. (b) Wavelet. (c) GSA. (d) PNN. (e) FusionNet. (f) MSDCNN. (g) ADKNet. (h) DiCNN. (i) LAGConv. (j) BiMPan. (k) MDPNet. (l) Ours. (m) GT.

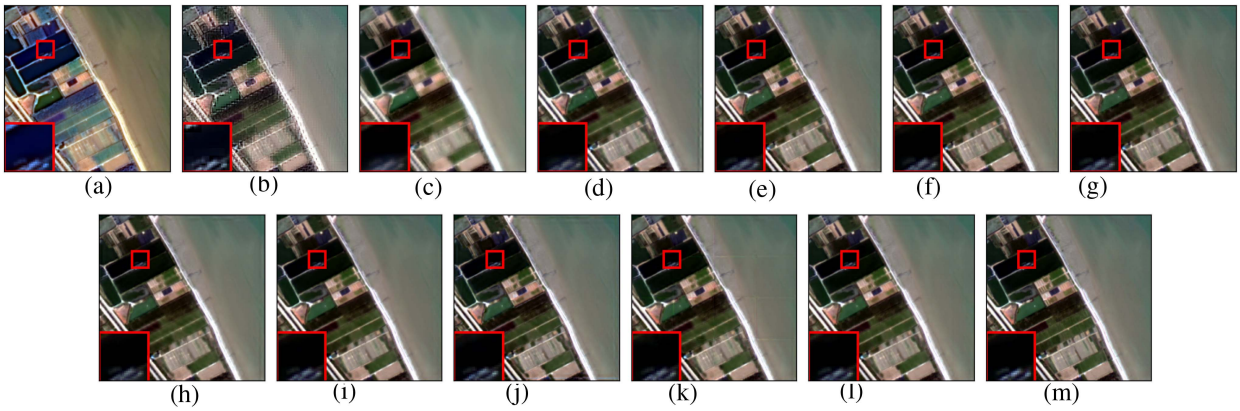


Fig. 9. Visual representation of the HR-MS image corresponding to the model on the GF2 dataset is generated using GT images as reference. (a) Brovey. (b) Wavelet. (c) GSA. (d) PNN. (e) FusionNet. (f) MSDCNN. (g) ADKNet. (h) DiCNN. (i) LAGConv. (j) BiMPan. (k) MDPNet. (l) Ours. (m) GT.

that our model exhibits superior perceptual quality. Furthermore, to better visualize the detailed differences between the fused images and the label images, we present the residual images between the fused and label images. As shown in Fig. 10, we calculate the average differences across channels and display the difference values through images. It is worth noting that in the residual images, our method reconstructs clearer texture details, thereby validating the effectiveness and superiority of our approach.

3) Performance on the WV3 Dataset: In this section, we will present the performance of all methods on the WV3 dataset, as shown in Table IV. This sentence explains what the bold text in Table IV represents. Similar to the performance on the QB and GF2 datasets, all DL-based methods outperform traditional learning methods in terms of quantitative results. In addition, our model achieves superior performance on all five metrics with reference.

For intuitive comparison, we visualized the fusion results of traditional learning and DL. The fusion results of the selected method are shown in Fig. 11. It can be observed through the magnified images of some areas of the fused image that the model we proposed can have excellent spatial texture detail information and spectral information. We also visualized the residual images between the fusion results of the model on the WV3 dataset and the GT images, as shown in Fig. 12. In the residual images of the fused image and the label image, it can be seen that the model proposed in this article has residual images with less information compared to other models. This also indicates that the method we proposed has better performance and visual performance compared to other DL methods and traditional learning methods in this article.

E. Performance of the Model on the Full Resolution Dataset

We also evaluated the performance of the selected traditional learning methods and DL methods on the full resolution datasets, including QB, GF2, and WV3. In the model performance evaluation tables of the three datasets, we bold the optimal index data.

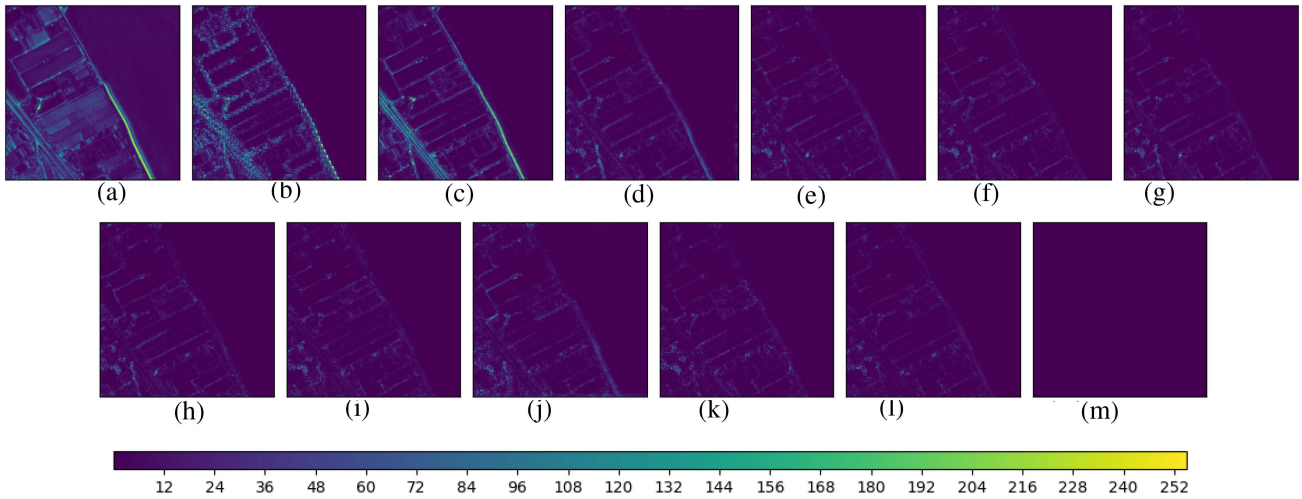


Fig. 10. Visual representation of the corresponding residual image of the model on the GF2 dataset is generated using GT images as reference. (a) Brovey. (b) Wavelet. (c) GSA. (d) PNN. (e) FusionNet. (f) MSDCNN. (g) ADKNet. (h) DiCNN. (i) LAGConv. (j) BiMPan. (k) MDPNet. (l) Ours. (m) GT.

TABLE IV
MODEL PERFORMANCE ON THE REDUCED RESOLUTION WV3 TEST SET

Method	SAM	PSNR	ERGAS	SCC	SSIM
GSA	5.7632	31.7432	5.2572	0.9415	0.8633
Wavelet	7.1080	29.5765	6.1915	0.8960	0.8068
Brovey	6.0762	30.3629	5.9197	0.9167	0.8376
FusionNet	3.2268	38.1622	2.3786	0.9854	0.9700
PNN	4.3967	36.0716	3.0940	0.9663	0.9544
MSDCNN	3.5937	37.4300	2.6068	0.9785	0.9650
ADKNet	3.1888	38.3742	2.3364	0.9856	0.9707
DiCNN	3.6122	37.2762	2.6841	0.9785	0.9636
LAGConv	3.0498	38.7257	2.2354	0.9873	0.9735
BiMPan	3.1189	38.5672	2.2762	0.9877	0.9728
MDPNet	3.2170	38.1774	2.3745	0.9845	0.9700
Ours	**2.9464**	**39.0238**	**2.1480**	**0.9887**	**0.9753**
Ideal Value	0	∞	0	1	1

TABLE V
PERFORMANCE OF THE MODEL ON THE FULL RESOLUTION QB TEST SET

Method	D_λ	D_s	QNR
GSA	0.0545	0.1785	0.7772
Wavelet	0.0567	0.1081	0.8420
Brovey	0.0259	0.1211	0.8562
FusionNet	0.0416	0.0268	0.9331
PNN	0.0277	0.1115	0.8643
MSDCNN	0.0401	0.0367	0.9245
ADKNet	0.0337	**0.0200**	0.9472
DiCNN	0.0261	0.1052	0.8719
LAGConv	0.0379	0.0410	0.9228
BiMPan	0.0287	0.0310	0.9415
MDPNet	0.0293	0.0463	0.9261
Ours	**0.0197**	0.0332	**0.9477**
Ideal Value	0	0	1

The three datasets each contain 20 pairs of LR-MS and PAN images. The shape of LR-MS in the QB dataset is $128 \times 128 \times 4$, the shape of LR-MS in the GF2 dataset is $128 \times 128 \times 4$, and the shape of LR-MS in the WV3 dataset is $128 \times 128 \times 8$. The shape of PAN images in the three datasets is $512 \times 512 \times 1$.

As shown in Table V, in the full-resolution testing on the QB dataset, the D_λ performance index of Brovey is better than some DL methods, whereas the D_s performance index of Wavelet is superior to that of PNN. However, regarding the QNR evaluation metric, DL methods are superior to traditional learning methods. Despite the fact that traditional learning methods may excel in certain metrics, DL approaches demonstrate greater scalability and can adapt to diverse datasets, which suggests that DL has an advantage in maintaining the fidelity of spectral and spatial information. It is noteworthy that our model exhibits excellent performance compared to other DL methods in both D_λ and QNR evaluation metrics.

As shown in Table VI, in the full-resolution testing on the GF2 dataset, the performance metrics of DL outperform those of the three traditional methods. It is noteworthy that our model achieves the best results compared to other methods in the D_s and QNR metrics.

As shown in Table VII, in the full-resolution testing on the WV3 dataset, the D_s index of the Wavelet method is superior to PNN and DiCNN in DL. However, our proposed model achieves the best results in all three no-reference metrics for the WV3 dataset, demonstrating the effectiveness of our proposed method.

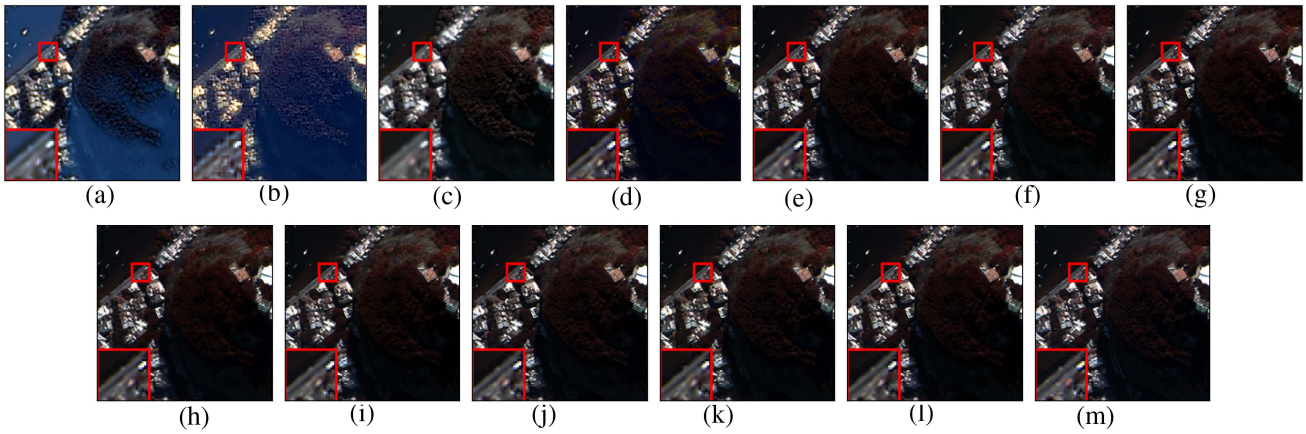


Fig. 11. Visual representation of the HR-MS image corresponding to the model on the WV3 dataset is generated using GT images as reference. (a) Brovey. (b) Wavelet. (c) GSA. (d) PNN. (e) FusionNet. (f) MSDCNN. (g) ADKNet. (h) DiCNN. (i) LAGConv. (j) BiMPan. (k) MDPNet. (l) Ours. (m) GT.

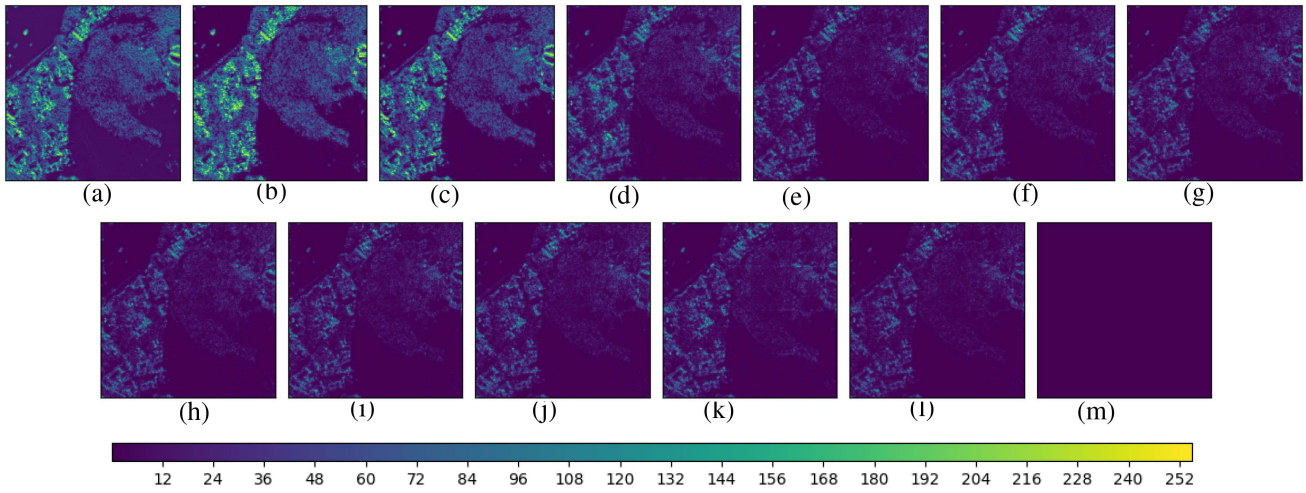


Fig. 12. Visual representation of the corresponding residual image of the model on the WV3 dataset is generated using GT images as reference. (a) Brovey. (b) Wavelet. (c) GSA. (d) PNN. (e) FusionNet. (f) MSDCNN. (g) ADKNet. (h) DiCNN. (i) LAGConv. (j) BiMPan. (k) MDPNet. (l) Ours. (m) GT.

TABLE VI
PERFORMANCE OF THE MODEL ON THE FULL RESOLUTION GF2 TEST SET

Method	D_λ	D_s	QNR
GSA	0.0288	0.0652	0.9082
Wavelet	0.0602	0.0751	0.8701
Brovey	0.0249	0.0786	0.8988
FusionNet	0.0132	0.0441	0.9433
PNN	0.0160	0.0490	0.9359
MSDCNN	0.0112	0.0388	0.9506
ADKNet	**0.0064**	0.0266	0.9672
DiCNN	0.0115	0.0452	0.9439
LAGConv	0.0114	0.0381	0.9510
BiMPan	0.0071	0.0309	0.9623
MDPNet	0.0065	0.0316	0.9621
Ours	0.0073	**0.0206**	**0.9723**
Ideal Value	0	0	1

TABLE VII
PERFORMANCE OF THE MODEL ON THE FULL RESOLUTION WV3 TEST SET

Method	D_λ	D_s	QNR
GSA	0.0231	0.0888	0.8908
Wavelet	0.0602	0.0751	0.8701
Brovey	0.0131	0.0846	0.9035
FusionNet	0.0206	0.0644	0.9165
PNN	0.0278	0.0787	0.8959
MSDCNN	0.0210	0.0692	0.9115
ADKNet	0.0190	0.0497	0.9323
DiCNN	0.0166	0.0757	0.9092
LAGConv	0.0173	0.0629	0.9211
BiMPan	0.0201	0.0575	0.9236
MDPNet	0.0242	0.0603	0.9171
Ours	**0.0110**	**0.0339**	**0.9556**
Ideal Value	0	0	1

In addition, we also conducted visualizations on the full-resolution datasets. As shown in Fig. 13, we visualized the full-resolution images on the QB dataset. It can be observed that, compared to other DL and traditional learning methods, our proposed method is capable of capturing finer textural details and more abundant spectral information. This reflects

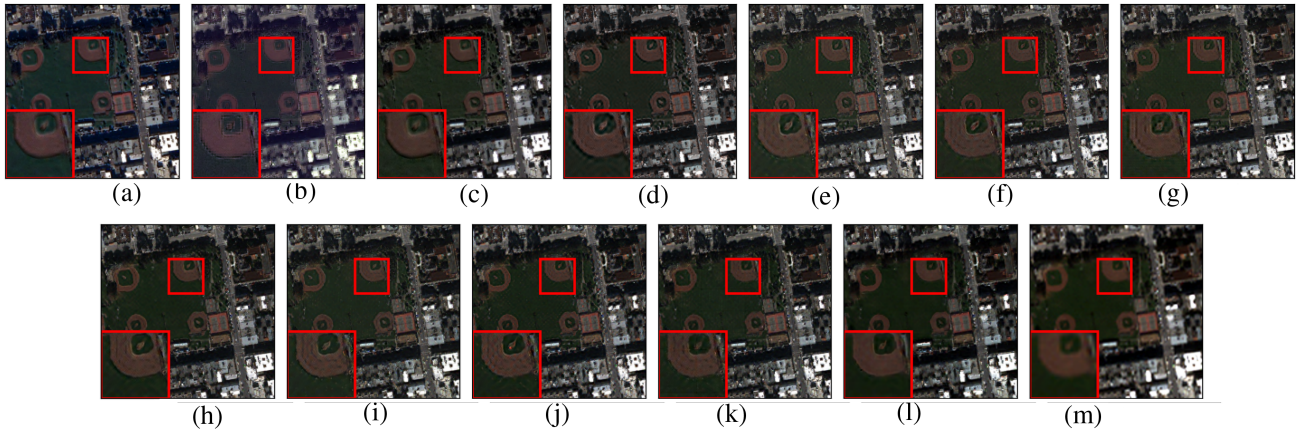


Fig. 13. Visual performance of the method on the QB dataset is assessed. (a) Brovey. (b) Wavelet. (c) GSA. (d) PNN. (e) FusionNet. (f) MSDCNN. (g) ADKNet. (h) DiCNN. (i) LAGConv. (j) BiMPan. (k) MDPNet. (l) Ours. (m) MS.

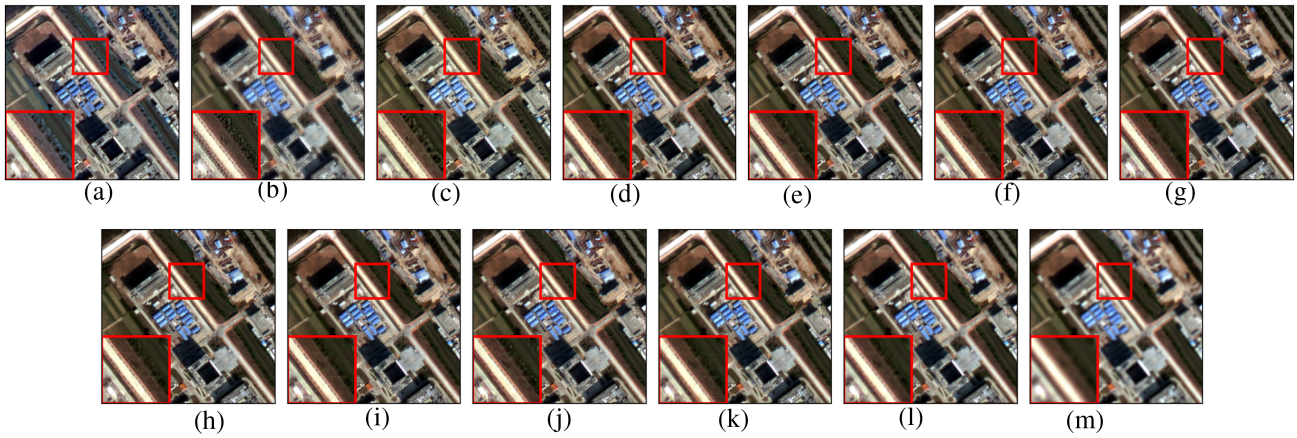


Fig. 14. Visual performance of the method on the GF2 dataset is assessed. (a) Brovey. (b) Wavelet. (c) GSA. (d) PNN. (e) FusionNet. (f) MSDCNN. (g) ADKNet. (h) DiCNN. (i) LAGConv. (j) BiMPan. (k) MDPNet. (l) Ours. (m) MS.

the efficiency of our model. As shown in Fig. 14, it can be seen that the fusion results of our model can display better spectral and spatial information on the GF2 dataset at full resolution, which demonstrates the superiority of our model. As shown in Fig. 15, our model demonstrates superior performance on the WV3 dataset compared to other traditional learning methods, capturing better spectral and detailed information. Compared with DL, our fusion results do not exhibit spectral distortion, which illustrates the superiority of our model.

V. Discussion

In this section, in order to prove the effectiveness of the module we proposed and to obtain the optimal results of the module, we conducted a series of ablation experiments, including verifying the effectiveness of the four main modules in the DGGSC model, exploring the influence of the parameter N and the loss function on the performance of the model, and for all ablation experiments, the best results are marked in bold.

Our base module includes the DRL module, the GCNN module, the MSDC module, and the CAL module. In addition, in order to evaluate the role of the module more objectively, we used the reduced-resolution QB test set to test the effectiveness of the module. The experiments show that the module and loss function proposed by us play an important role in improving the performance of the model.

A. Analysis of the Model Efficiency

As shown in Fig. 16, the relationship among the PSNR index, the number of parameters, and floating point operations (FLOPs) of the model is presented. Circle sizes indicate the size of FLOPs. It can be seen that our model has achieved a good balance between the parameters, the FLOPs, and the PSNR index. Under the condition of improving the performance of the model, out model can significantly reduce the trainable parameters of the model. As shown in Table VIII, we compared the average training time and average testing time of DL models. Testing time under one second is underlined in the table. It can be observed that the training time of our model is shorter than that of BiMPan and ADKNet. Regarding the testing time, the testing time of our model is under 1 s and is superior to the testing time of ADKNet, LAGConv, and BiMPan.

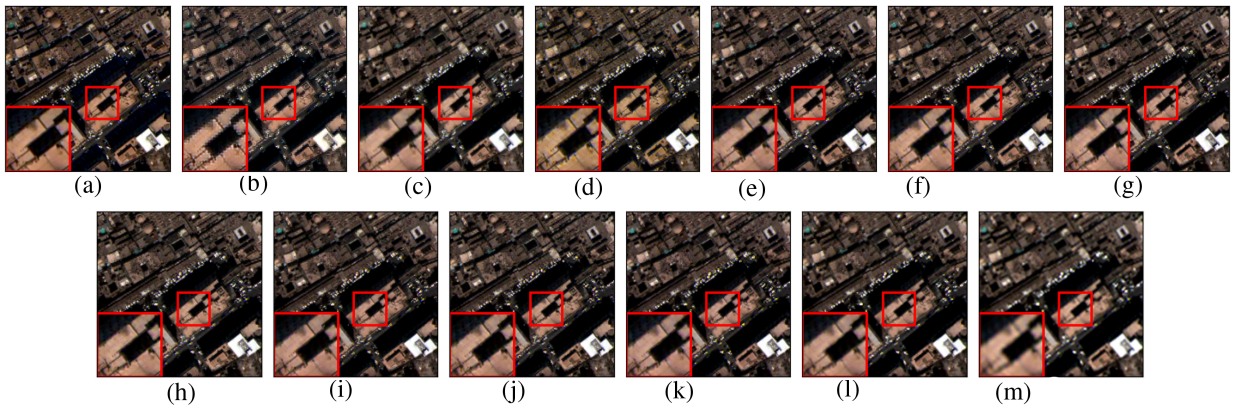


Fig. 15. Visual performance of the method on the WV3 dataset is assessed. (a) Brovey. (b) Wavelet. (c) GSA. (d) PNN. (e) FusionNet. (f) MSDCNN. (g) ADKNet. (h) DiCNN. (i) LAGConv. (j) BiMPan. (k) MDPNet. (l) Ours. (m) MS.

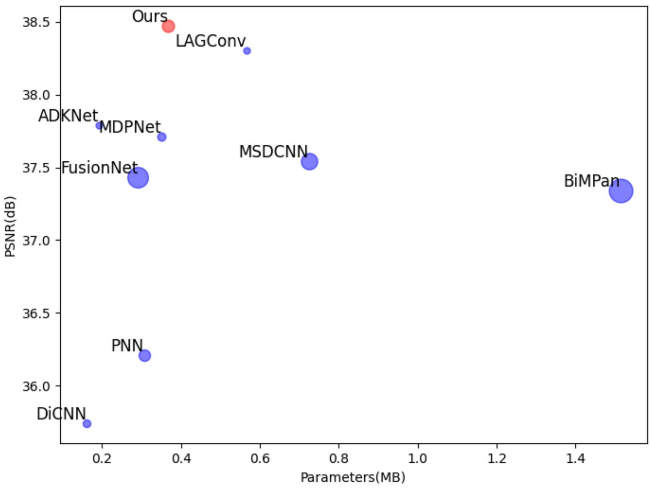


Fig. 16. Parameter versus PSNR versus FLOPs.

TABLE VIII
TRAINING TIME AND TESTING TIME OF DIFFERENT METHODS ON THE REDUCED RESOLUTION QB TEST SET

Method	Training time (h)	Testing time (s)
FusionNet	0.6	0.0728
PNN	0.33	0.0301
MSDCNN	0.75	0.1163
ADKNet	3.68	1.0923
DiCNN	0.33	0.0308
LAGConv	3.35	1.1528
BiMPan	23.25	2.9935
MDPNet	1.6	0.2610
Ours	3.43	0.9663

B. Analysis of Deep Feature Mixing Module

As shown in Table XIII, we explored the parameter N in the model. Experiments indicate that when the number of deep feature mixing modules is set to 4, the performance of the model reaches its optimum.

C. Effectiveness of GCNN Module

As shown in Table IX, GCNN plays a significant role in the model. When GCNN is added, the performance of the model has improved significantly. We also investigated the influence of the value of ws in the GCNN module on the model's performance. As indicated in Table XI, when ws is set to 8, the model performance reaches its optimum. When the GCNN module is replaced by the CNN module, the performance declines, which indicates the efficiency of our GCNN model.

D. Effectiveness of DRL Module

As shown in Table IX, the effectiveness of the DRL module is demonstrated. We further explored the parameter q in DRL. As shown in Table XIV, when q is set to 3, the performances of PSNR, ERGAS, and SSIM of the model reach the optimum. When q is set to 4, the SAM performance of the model reaches its optimum, whereas the SCC performance has a small difference from that when q is set to 3. Therefore, in this article, we choose to set q to 3.

E. Effectiveness of CAL Module

We explored the validity of the CAL module. As shown in Table IX, compared with the model with the CAL module, the performance of the model without the CAL module decreased significantly, which indicates the validity of the CAL module.

We also explored the influence of LN in the channel attention learning module on performance and the selection of activation functions. We compared the performance metrics of using LN and not using LN on the reduced-resolution test set images, as shown in Table XV. Compared with not using LN, the performance was improved to a certain extent when using LN.

In addition, we compared the influence of the traditional ReLU activation function and the GELU activation function adopted in this article on the model performance, as shown in Table XV. The performance of the GELU activation function is superior to that of the ReLU performance metric, and the influence of the activation function on performance is greater than that of LN.

TABLE IX
IMPACT OF MODULE COMBINATION ON PERFORMANCE FOR THE REDUCED RESOLUTION QB TEST SET

DRL	CAL	GCNN	MSDC	SAM	PSNR	ERGAS	SCC	SSIM
✓	✓	✓	×	4.6671	38.1967	3.7696	0.9844	0.9609
✓	✓	✓	✓	**4.6072**	**38.3909**	**3.6739**	**0.9861**	**0.9624**
×	✓	✓	✓	4.6517	38.2272	3.7415	0.9851	0.9613
×	×	✓	✓	4.9495	37.7065	3.9840	0.9800	0.9562
×	×	×	✓	5.2531	37.2947	4.2384	0.9780	0.9518

TABLE X
PERFORMANCE OF THE LOSS FUNCTION SETUP ON THE REDUCED RESOLUTION QB TEST SET

MAE	FFT	MSE	SAM	PSNR	ERGAS	SCC	SSIM
×	✓	✓	4.6072	38.3909	3.6739	0.9861	0.9624
✓	✓	×	4.5445	**38.4731**	**3.6485**	**0.9864**	**0.9631**
✓	×	×	**4.5412**	38.3845	3.6857	0.9861	0.9627
×	×	✓	4.9057	37.8961	3.8939	0.9815	0.9575

TABLE XI
PERFORMANCE OF THE WS PARAMETER IN GCNN ON THE REDUCED RESOLUTION QB TEST SET

WS	SAM	PSNR	ERGAS	SCC	SSIM
2	4.5457	38.3989	3.6745	0.9863	0.9629
4	4.5669	38.4124	3.6766	0.9862	0.9628
8	**4.5445**	**38.4731**	**3.6485**	**0.9864**	**0.9631**
16	5.5327	36.5898	4.5485	0.9702	0.9454
CNN	4.5491	38.4020	3.6758	**0.9864**	0.9624

TABLE XII
PERFORMANCE OF THE α PARAMETER IN FFT LOSS FUNCTION ON THE REDUCED RESOLUTION QB TEST SET

α	SAM	PSNR	ERGAS	SCC	SSIM
0	4.5412	38.3845	3.6857	0.9861	0.9627
0.05	**4.5445**	**38.4731**	**3.6485**	**0.9864**	**0.9631**
0.1	4.5830	38.3342	3.6863	0.9860	0.9625
0.2	4.5795	38.4181	3.6827	0.9862	0.9628
0.5	4.5623	38.3969	3.6670	0.9863	0.9628

TABLE XIII
PERFORMANCE OF THE N PARAMETER IN DGGSC ON THE REDUCED RESOLUTION QB TEST SET

N	SAM	PSNR	ERGAS	SCC	SSIM
2	4.6558	38.2687	3.7384	0.9849	0.9612
3	4.6061	38.3103	3.7168	0.9857	0.9619
4	**4.5445**	**38.4731**	**3.6485**	**0.9864**	**0.9631**
5	4.5546	38.3177	3.7137	0.9860	0.9623

F. Effectiveness of MSDC Module

We conducted an evaluation of the effectiveness of the MSDC module, as illustrated in Table IX. The results indicate that the performance metrics, including SAM, PSNR, ERGAS, SCC, and SSIM, exhibit a decline when the model operates without the MSDC module. This underscores the efficacy of the MSDC module.

G. Effectiveness of Our Model Loss Function

We explored the changes in the model performance caused by different combinations of loss functions, as shown in Table X.

TABLE XIV
PERFORMANCE OF THE q PARAMETER IN DRL ON THE REDUCED RESOLUTION QB TEST SET

q	SAM	PSNR	ERGAS	SCC	SSIM
2	4.5609	38.3427	3.7025	0.9860	0.9625
3	4.5445	**38.4731**	**3.6485**	0.9864	**0.9631**
4	**4.5370**	38.4149	3.6637	**0.9865**	0.9629
5	4.5436	38.4072	3.6589	0.9864	0.9628
8	4.5512	38.4189	3.6704	0.9862	0.9628

TABLE XV
PERFORMANCE EVALUATION OF CAL ON THE REDUCED RESOLUTION QB TEST SET

	SAM	PSNR	ERGAS	SCC	SSIM
w/ LN	**4.5445**	**38.4731**	**3.6485**	**0.9864**	**0.9631**
w/o LN	4.5490	38.4128	3.6760	**0.9864**	0.9629
act $\rightarrow$ ReLU	4.5505	38.3946	3.6750	**0.9864**	0.9629

The performance of the model using FFT loss is significantly better than that of the model without using FFT loss. Through comparison, we finally selected the L_1 loss function as our image reconstruction loss function.

We further explored the influence of the coefficient of FFT loss on the model performance, as shown in Table XII. We selected different coefficients to evaluate the model. When α takes 0.05, the performance of the model reaches its optimum. Therefore, in this article, we set the parameter α to 0.05.

VI. CONCLUSION

In this article, we propose an efficient remote sensing PAN sharpening network named DGGSC. The model uses residual structure, the GCNN module, the DRL module, and the MSM module to increase the expressive ability of the model and improve its performance. The model utilizes downsampling to achieve multiscale information perception without increasing the time and parameter quantities of the model. Through our analysis, DGGSC is capable of generating visually clearer and more realistic HR-MS images, including better texture details.

The DGGSC model can generate excellent HR-MS images on multiple satellite image datasets with different spectral resolutions and spatial resolutions. However, it still has some limitations. For instance, the ability of the model to consider context perception needs to be further improved. In the future, we will further enhance the performance indicators of the model to make it more convenient and efficient.

REFERENCES

[1] C. Bai, X. Bai, and K. Wu, "A review: Remote sensing image object detection algorithm based on deep learning," *Electronics*, vol. 12, no. 24, Dec. 2023, Art. no. 4902.

[2] S. N. Shivappriya, M. J. P. Priyadarsini, A. Stateczny, C. Puttamadappa, and B. D. Parameshachari, "Cascade object detection and remote sensing object detection method based on trainable activation function," *Remote Sens.*, vol. 13, no. 2, Jan. 2021, Art. no. 200.

[3] Y. Hu, J. Chen, D. Pan, and Z. Hao, "Edge-guided image object detection in multiscale segmentation for high-resolution remotely sensed imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4702–4711, Aug. 2016.

[4] S. Ren and Q. Liu, "Small target augmentation for urban remote sensing image real-time segmentation," *IEEE Trans. Intell. Transport. Syst.*, vol. 25, no. 2, pp. 2076–2088, Feb. 2024.

[5] A. Ma, J. Wang, Y. Zhong, and Z. Zheng, "FactSeg: Foreground activation-driven small object semantic segmentation in large-scale remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5606216.

[6] J. K. Gilbertson, J. Kemp, and A. V. Niekerk, "Effect of pan-sharpening multi-temporal Landsat 8 imagery for crop type differentiation using different classification techniques," *Comput. Electron. Agriculture*, vol. 134, pp. 151–159, Mar. 2017.

[7] H. Hallabia and H. Hamam, "An enhanced pansharpening approach based on second-order polynomial regression," in *Proc. Int. Wireless Commun. Mobile Comput.*, Harbin City, China, 2021, pp. 1489–1493.

[8] X. Feng, J. Hu, W. Wu, and S. Fan, "Dynamic large-small kernel convolutional neural network for pansharpening," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, 2023, Art. no. 5002305.

[9] S.-S. Xiao, C. Jin, T.-J. Zhang, R. Ran, and L.-J. Deng, "Progressive band-separated convolutional neural network for multispectral pansharpening," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Brussels, Belgium, 2021, pp. 4464–4467.

[10] M. Choi, "A new intensity-hue-saturation fusion approach to image fusion with a tradeoff parameter," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 6, pp. 1672–1682, Jun. 2006.

[11] 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.

[12] A. Garzelli, F. Nencini, and L. Capobianco, "Optimal MMSE pan sharpening of very high resolution multispectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 1, pp. 228–236, Jan. 2008.

[13] X. He, C. Zhou, J. Zhang, and X. Yuan, "Using wavelet transforms to fuse nighttime light data and POI Big Data to extract urban built-up areas," *Remote Sens.*, vol. 12, no. 23, Nov. 2020, Art. no. 3887.

[14] M. M. Khan, J. Chanussot, L. Condat, and A. Montanvert, "Indusion: Fusion of multispectral and panchromatic images using the induction scaling technique," *IEEE Geosci, Remote Sens. Lett.*, vol. 5, no. 1, pp. 98–102, Jan. 2008.

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

[16] M. Wang, G. Xie, Z. Zhang, Y. Wang, S. Xiang, and Y. Pi, "Smoothing filter-based panchromatic spectral decomposition for multispectral and hyperspectral image pansharpening," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 3612–3625, 2022.

[17] D. Fasbender, J. Radoux, and P. Bogaert, "Bayesian data fusion for adaptable image pansharpening," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 6, pp. 1847–1857, Jun. 2008.

[18] M. R. Vicinanza, R. Restaino, G. Vivone, M. D. Mura, and J. Chanussot, "A pansharpening method based on the sparse representation of injected details," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 1, pp. 180–184, Jan. 2015.

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

[20] 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.*, Venice, Italy: IEEE, 2017, pp. 1753–1761.

[21] 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. Top. Appl. Earth Observ. Remote Sens.*, vol. 11, no. 3, pp. 978–989, Mar. 2018.

[22] Z.-R. Jin, T.-J. Zhang, T.-X. Jiang, G. Vivone, and L.-J. Deng, "LAGConv: Local-context adaptive convolution kernels with global harmonic bias for pansharpening," in *Proc. AAAI Conf. Artif. Intell.*, 2022, pp. 1113–1121.

[23] K. Han, Y. Wang, J. Guo, Y. Tang, and E. Wu, "Vision GNN: An image is worth graph of nodes," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 8291–8303.

[24] J. Chen et al., "Run, don't walk: Chasing higher FLOPS for faster neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 12 021–12 031.

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

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

[27] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Representations*, 2020, pp. 1–12.

[28] X. Meng, N. Wang, F. Shao, and S. Li, "Vision transformer for pansharpening," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5409011.

[29] J. Yin, J. Qu, L. Sun, W. Huang, and Q. Chen, "A local and nonlocal feature interaction network for pansharpening," *Remote Sens.*, vol. 14, no. 15, Aug. 2022, Art. no. 3743.

[30] Z. Li, J. Li, L. Ren, and Z. Chen, "Transformer-based dual-branch multi-scale fusion network for pan-sharpening remote sensing images," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 614–632, 2024.

[31] S. Zhou, J. Zhang, W. Zuo, and C. C. Loy, "Cross-scale internal graph neural network for image super-resolution," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 3499–3509.

[32] D. Han, X. Pan, Y. Han, S. Song, and G. Huang, "FLatten Transformer: Vision transformer using focused linear attention," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 5961–5971.

[33] L. Sun, J. Dong, J. Tang, and J. Pan, "Spatially-adaptive feature modulation for efficient image super-resolution," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*2023, pp. 13 190–13 199.

[34] S. Peng, L.-J. Deng, J.-F. Hu, and Y. Zhuo, "Source-adaptive discriminative kernels based network for remote sensing pansharpening," in *Proc. 31st Int. Joint Conf. Artif. Intell.*, Vienna, Austria, 2022, pp. 1283–1289.

[35] J. Hou, Q. Cao, R. Ran, C. Liu, J. Li, and L. Deng, "Bidomain modeling paradigm for pansharpening," in *Proc. 31st ACM Int. Conf. Multimedia*, Ottawa ON Canada: ACM, 2023, pp. 347–357.

[36] W. Fan, F. Liu, and J. Li, "Pansharpening via multiscale embedding and dual attention transformers," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 2705–2717, 2024, doi: 10.1109/JSTARS.2023.3344215.

[37] B. Aiazzi, S. Baronti, and M. Selva, "Improving component substitution pansharpening through multivariate regression of MS +Pan data," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 10, pp. 3230–3239, Oct. 2007.

[38] R. L. King and J. Wang, "A wavelet based algorithm for pan sharpening landsat 7 imagery, in IGARSS 2001. scanning the present and resolving the future," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Sydney, NSW, Australia: IEEE, 2001, pp. 849–851.

[39] 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, Aug. 1987.

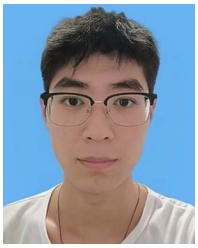
[40] R. H. Yuhas, A. F. Goetz, and J. W. Boardman, "Discrimination among semi-arid landscape endmembers using the spectral angle mapper (SAM) algorithm," in *Proc. 3rd Annu. JPL Airborne Geosci. Workshop*, 1992, pp. 147–149.

[41] X. Liu, Q. Liu, and Y. Wang, "Remote sensing image fusion based on two-stream fusion network," *Inf. Fusion*, vol. 55, pp. 1–15, Mar. 2020.

[42] A. Arienzo, G. Vivone, A. Garzelli, L. Alparone, and J. Chanussot, "Full-resolution quality assessment of pansharpening: Theoretical and hands-on approaches," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 3, pp. 168–201, Sep. 2022.

Weisheng Li (Member, IEEE) received the B.S. degree in industrial automation and the M.S. degree in mechanical manufacturing and automation from the School of Electronics and Mechanical Engineering, Xidian University, Xi'an, China, in 1997 and 2000, respectively, and the Ph.D. degree in computer science and technology from the School of Computer Science and Technology, Xidian University, in 2004.

He is currently a Professor with the Chongqing University of Posts and Telecommunications, Chongqing, China. His current research interests include intelligent information processing and pattern recognition.



Xudong Zhi was born in Hebei, China, in 2001. He received the bachelor's degree in computer science and technology in 2023 from the Chongqing University of Science and Technology of China, Chongqing China, where he is currently working toward the master's degree in computer science and technology with the Key Laboratory of Computer Science and Technology.

His research interests include deep learning and remote sensing image processing.

Yidong Peng was born in Chongqing, China, in 1988. He received the M.S. and Ph.D. degrees in computer science and technology from the Institute of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, China, in 2017 and 2021, respectively.

He is currently a Lecturer with the Chongqing University of Posts and Telecommunications. His research interests include urban thermal infrared remote sensing, remote sensing image processing, and ecological environment monitoring and evaluation.

Yijian Hu was born in Ningbo, China, in 2000. He received the bachelor's degree in 2022 from the Chongqing University of Posts and Telecommunications of China, Chongqing, China, where he is currently working toward the masters degree with the Key Laboratory of Computer Science and Technology.

His research interests include deep learning and remote sensing image processing.