A Spectral-Spatial Feature Extraction Method With Polydirectional CNN for Multispectral Image Compression

Fanqiang Kong, Kedi Hu, Yunsong Li, Dan Li, Xin Liu, *Senior Member, IEEE*, and Tariq S. Durrani

***Abstract*—Convolutional neural networks (CNN) has been widely used in the research of multispectral image compression, but they still face the challenge of extracting spectral feature effectively while preserving spatial feature with integrity. In this article, a novel spectral-spatial feature extraction method is proposed with polydirectional CNN (SSPC) for multispectral image compression. First, the feature extraction network is divided into three parallel modules. The spectral module is employed to obtain spectral features along the spectral direction independently, and simultaneously, with two spatial modules extracting spatial features along two different spatial directions. Then all the features are fused together, followed by downsampling to reduce the size of the feature maps. To control the tradeoff between the rate loss and the distortion, the rate-distortion optimizer is added to the network. In addition, quantization and entropy encoding are applied in turn, converting the data into bit stream. The decoder is structurally symmetric to the encoder, which is convenient for structuring the framework to recover the image. For comparison, SSPC is tested along with JPEG2000 and three-dimensional (3-D) SPIHT on the multispectral datasets of Landsat-8 and WorldView-3 satellites. Besides, to further validate the effectiveness of polydirectional CNN, SSPC is also compared with a normal CNN-based algorithm. The experimental results show that SSPC outperforms other methods at the same bit rates, which demonstrates the validity of the spectral-spatial feature extraction method with polydirectional CNN.**

***Index Terms*—Compression algorithms, convolutional neural network (CNN), feature extraction, multispectral image, rate-distortion optimizer.**

I. Introduction

WITH the rapid development of multispectral imaging technology, rich spectral-spatial features are encoded in several narrow and contiguous spectral bands [1] to generate multispectral images which can reflect more characteristics of the same scene rather than RGB images. With the abundance of spectral and spatial information, multispectral images can provide subtle geometric features to analyze targets [2]. Hence, they have been exploited in various areas, such as environment monitoring, crop condition assessment, military reconnaissance, target surveillance, and so on [3]–[6]. However, due to the complex characteristics, the data volume of multispectral image increases dramatically. The huge amount of data brings great pressure to the transmission, storage and application of the images; thus it is necessary to compress the data effectively, especially when channel capability is limited.

For RGB image compression, only the spatial correlation needs to be considered in most cases, hence various useful traditional compression methods have emerged, such as JPEG and JPEG2000. These algorithms usually make transformations in spatial dimension to get rid of spatial redundancy, which leads to compressed images with a higher compression ratio. It also obtains pretty good effects when directly applying these traditional methods to multispectral image compression, though, in view of the characteristics of multispectral images, it represents a research trend into a more customized algorithm focusing on spectral-spatial feature extraction.

With decades of unremitting efforts, multispectral image compression technology has achieved good results so far. Multispectral image compression methods may be roughly summarized as vector quantization (VQ) coding [7], predictive coding [8] and transform coding [9], [10]. The theoretical basis of VQ coding is Shannon's rate distortion theory. Here the input vector is replaced by the index of the codeword that matches the input vector in the codebook for data transmission and storage, and when decoding, only simple search for the codebook is enough. Since the performance of this algorithm is closely related to the design of codebook, in order to improve the efficiency of the algorithm and reduce the complexity of the operation as well, Motta *et al.* [11] used piecewise VQ to reduce the size of the codebook and speed up the searching of the codebook, so as to achieve a good compression performance. Due to technologies such as codebook designing and codeword searching, the computation complexity of VQ coding is usually very high, which has hindered the development of high-performance algorithm. Predictive coding has been widely used in lossless

Manuscript received November 16, 2021; revised February 14, 2022; accepted March 5, 2022. Date of publication March 10, 2022; date of current version April 15, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61801214, and in part by the National Key Laboratory Foundation under Grant 6142411192112. *(Corresponding authors: Kedi Hu; Xin Liu.)*

Fanqiang Kong, Kedi Hu, and Dan Li are with the College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China (e-mail: kongfq@nuaa.edu.cn; kedi_hu@nuaa.edu.cn; danli@nuaa.edu.cn).

Yunsong Li is with the State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China (e-mail: ysli@mail.xidian.edu.cn).

Xin Liu is with the School of Information and Communication Engineering, Dalian University of Technology, Dalian 116024, China (e-mail: liuxinstar1984@dlut.edu.cn).

Tariq S. Durrani is with the Department of Electronic and Electrical Engineering, University of Strathclyde, G1 1XQ Glasgow, U.K. (e-mail: t.durrani@strath.ac.uk).

Digital Object Identifier 10.1109/JSTARS.2022.3158281

compression. The main idea is to utilize the spatial (or spectral) correlations to predict the pixel of the current position based on its adjacent pixels, and then encode the residual between the predicted value and the actual value. Differential pulse code modulation (DPCM) [12] is one of the most basic predictive coding algorithms. In view of the characteristic of multispectral image compression, Mielikainen *et al.* [13] proposed an algorithm based on clustering and DPCM to cluster the spectra and build the prediction model, calculating the predictors in order to remove the spectral correlations. Although predictive coding is easy to compute and implement, it also has limitation in that the compression ratio is relatively low, which imposes restrictions on its applied range. Transform coding algorithm removes the spatial or spectral correlations via mapping the data representation from spatial domain to frequency (or others) domain. It prefers bearing a certain amount of information loss in exchange for a bigger compression ratio, so it is more popular in lossy compression. For better performance, the spectral and spatial redundancies of the image are usually removed by combinations of different transform coding schemes. Typical coding schemes include Karhunen–Loève transform (KLT) [14], discrete cosine transform [15], discrete wavelet transform [16], and so on. Extending to three-dimension (3-D), both 3-D-SPIHT [17] and 3-D-SPECK [18] have been proved to have a superior comprehensive property.

As mentioned above, the development of traditional compression algorithms has achieved great results up to now. With the deeper insight into multispectral images, it is obvious that the traditional methods can no longer meet the rising standard of its diversified application requirements. To seek breakthroughs, many researchers have paid attention to deep learning technology, which has been flourishing in recent years. Deep learning combines several linear and nonlinear layers together to extract features and improve the expression ability of the model. Dong *et al.* [19] proposed a CNN-based method for single image super-resolution, leading to a new era of solving the pixel-level problems of images with deep learning technology. In comparison with recurrent neural network (RNN) and generative adversarial network (GAN), the procedure of CNN processing of the input information is very similar to that of visual system, making this a reliable approach in the image processing field. GAN, however, it can generate fraudulent image with clear texture and high resolution, which may be quite different from the original image. As for RNN, it applies to sequence images, such as predicting the next frame of a video, etc. Because of its distinctive framework, RNNs generally have high algorithmic complexity. On the other hand, the simple structure of CNN makes it easier to build functional modules, to adapt to different specific requirements. To sum up, we decided to choose CNN to construct the multispectral image compression framework. The history of CNN-based methods has matured with the work on LeNet-5 [20], which consists of two convolution layers, two pooling layers and three full-connected layers. AlexNet [21], proposed in 2012, followed this idea but added extra convolution layers between every two pooling layers as an improvement, to take features of different scales into account. Later, excellent frameworks like VGG [22] and GoogLeNet [23] emerged, which constantly optimize the architecture and improve the performance of the network. Another milestone in the development of CNN is ResNet [24], where the idea of shortcut and residual learning was introduced, helping deal with the problem of vanishing gradient when the depth of the network is increased. Motivated by this, a novel compression framework based on optimized residual unit [25] is presented by Liu and his group. The experimental results show that the proposed learning framework is superior to BPG and JPEG2000 both objectively and subjectively. To obtain a high-quality recovered image at low bit rate, Jiang *et al.* [26] integrated two CNNs seamlessly into one compression framework for joint training. The CNN in the encoder is for feature extraction and the other in the decoder is for image reconstruction. The collaboration of two CNNs successfully reduces the block effects and improves the quality of the restored images. Since the algorithms mentioned above are all designed for RGB image, and as the RGB image also has three channels, it is naturally to apply these compression methods to multispectral images, which may be seen as special images with more channels. As a consequence, Kong *et al.* [27] improved the framework and put forward an end-to-end learning framework based on CNN specifically for multispectral image compression. This algorithm surpasses JPEG2000 on peak signal to noise ratio (PSNR) by about 2 dB, however, it still has problem that the strong spectral correlation is ignored. The reason lies in the dominance of spatial correlation in visible image compression, but for multispectral image, spectral features also need to be taken seriously or else this causes unnecessary information loss.

In response to the problems of the compression framework mentioned above, it is important to work on the spectral feature extraction methods. Recently, there is several effective feature extraction algorithms have been proposed based on CNN in the field of hyperspectral image (HSI) classification. Patel and Upla [28] combined the autoencoder and CNN together, and used the autoencoder to enhance the image in the first place, after which the features can be easily obtained by the shallow CNN architecture. Similar to Patel, Sellami and Tabbone also employed a simple autoenccoder to preprocess the HSI which can highly reduce the high dimensionality of the data [29]. After that, a multi-view deep autoencoder is proposed to combine both spectral and spatial features, followed by a CNN to integrate graph topology and improve the performance by preserving the spectral-spatial features. Zhong *et al.* [30] designed a spectral-spatial residual network with 3-D convolution. The residual blocks connect every convolutional layer to facilitate the gradient propagation. Roy *et al.* [31] proposed a hybrid spectral CNN, which consists of a 3-D-CNN and a 2-D-CNN. First, the 3-D-CNN is used to extract joint spectral-spatial features, and then the 2-D-CNN is used to learn more abstract-level spatial representation. Different from the aforementioned frameworks, the two-branch architecture is also popular, such as [32] and [33], to extract the joint spectral spatial features from HSI. Generally, the spatial branch adopts normal 2-D convolution to extract hierarchical features from spatial domain, and the spectral branch is composed of 1-D convolution, dramatically reduces the computational load. Inspired by these excellent ideas, we propose a spectral-spatial feature extraction method

with polydirectional CNN for multispectral image compression in this article.

With a view to the rich spectral-spatial features, the spectral-spatial feature extraction method with polydirectional CNN (SSPC) is used to extract two parts of features from different directions. The whole framework is composed of an encoder and a decoder. In the encoder, there are three parallel feature extraction branches, of which one is for spectral and the other two are for spatial features. It is noteworthy that the input image tensor needs to be permuted additionally before extracting spatial features. After this, three parts of features are first fused separately, and then concatenated together for downsampling and quantization. Lossless entropy encoding is then employed to obtain a compressed binary bit stream. When reconstructing the image, the bit stream goes through the entropy decoder, inverse quantization and upsampling in turn, to restore the size of the feature maps. At the end of the decoder, the data is sent to the SSPC network to recover the spectral and spatial features respectively and then fuse them together. With the joint spectral-spatial feature, the multispectral image can be recovered with high quality. The experiment results validate the efficacy of our proposed network, and our approach exceeds JPEG2000 and 3-D-SPIHT on both criteria of PSNR and spectral similarity (SS).

The key contributions of this article can be listed as follows.

1) The proposed framework performs end-to-end training in allusion to the characteristics of multispectral images, without any complicated image preprocessing operations such as registration, correction, etc. After obtaining the optimal model, the uncalibrated image to be tested is directly input into the framework for quick compression.
2) The proposed method separately extracts the spectral and spatial features of the multispectral image from different directions, which not only ensures the integrity of spatial feature, but also preserves the independence of the spectral information. Comparing with the existing CNN-based image compression method, our framework concentrates on the rich spectral features, which always tend to be ignored in most cases.
3) Different from the normal 3-D convolution ($k \times k \times c$), this article uses point-wise convolution ($1 \times 1 \times c$) and depth-wise convolution ($k \times k \times 1$) to extract the spectral and spatial features, respectively, effectively reducing the parameter numbers when the number of convolution kernels is the same.
4) Batch processing is used in the proposed method. Faced with massive amounts of data, the spectral-spatial features of each image can be individually extracted, and the corresponding compressed bit stream is generated, as well as the reconstructed image, which improves the compression efficiency.

The rest of article is organized as follows. Section II demonstrates the proposed network and some main components of the framework, Section III represents the implementation details of the experiment and equipment, and the training process as well. The experimental results are reported in Section IV. Finally, Section V concludes this article.

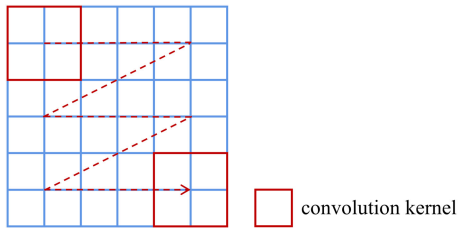

Fig. 1. Moving direction of convolution kernel.

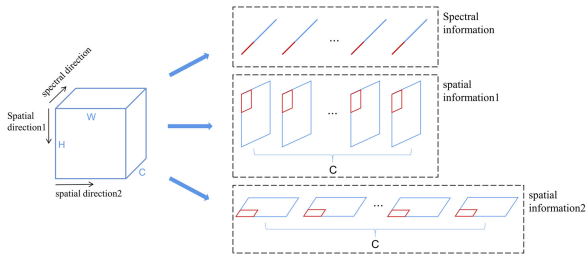

Fig. 2. Decomposition diagram of polydirectional CNN.

II. Proposed Method

In this section, we develop the proposed SSPC and demonstrate the training flow diagram. Besides, the spectral-spatial feature module is explained in detail, as well as the rate-distortion optimizer.

A. Polydirectional CNN

Normal convolution, whether 1-D, 2-D, or 3-D, calculates the image tensor with a sliding window in the same direction—from left to right and from top to bottom, as shown in Fig. 1.

The sliding window can cover all pixels on the spatial dimension and lower the computational complexity. Compared with traditional compression methods, CNN has been proven to be effective offering greater potential in several fields of image vision and processing, such as image classification, object identification, target detection, etc. For RGB images, CNN is sufficient to extract the needed spatial features since spectral information is not that important for them. When it comes to multispectral image compression, however, simple CNN might ignore pretty much spectral information, which is crucial for multispectral data.

In view of the problem mentioned above, we put forward polydirectional CNN as the homologous counterplan, which enables the convolution kernel to extract spectral features along spectral direction and spatial features along the spatial direction, respectively. The characteristics of sliding window make it possible to obtain spectral-spatial features with integrality, as long as the moving direction could be changed.

Since the arrangement of the image tensor and the moving direction of convolution kernel are relative, we adopt the method of transposing the image tensor as a substitute, which is much easier to operate. The decomposition of polydirectional CNN is shown as Fig. 2, the size of the input image is $H \times W \times C$, and convolution kernels are marked in red.

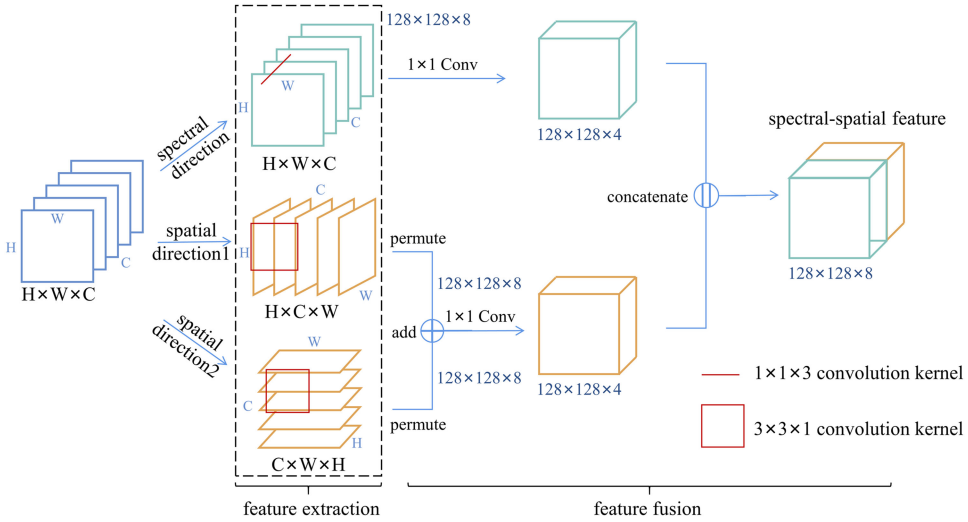

Fig. 3. Spectral-spatial module based on polydirectional CNN.

B. SSPC

Fig. 3 shows the structure of the spectral-spatial feature extraction module based on polydirectional CNN. As we can tell, the module is divided into two parts: feature extraction and feature fusion. The multispectral images are fed into three parallel circuits first, one for spectral and other two for spatial. After feature extraction, fusing spectral and spatial features together for further processing, downsampling, for example.

1) Feature Extraction: To adapt to multispectral image, which is 3-D, 3-D convolution is employed in the network. This operation can be formulated as

$$v_{ij}^{xyz} = f\left(\sum_{k=1}^{K_{i-1}}\sum_{p=0}^{P_i-1}\sum_{q=0}^{Q_i-1}\sum_{r=0}^{R_i-1} w_{ijk}^{pqr} v_{(i-1)k}^{(x+p)(y+q)(z+r)} + b_{ij}\right) \tag{1}$$

where v_{ij}^{xyz} is the output value at position (x, y, z) of the jth feature map in the ith layer, w_{ijk}^{pqr} denotes the weight of the convolution kernel at position (p, q, r) connected to the kth feature map, and k indexes over the set of feature maps in the $(i-1)$th layer which connected to the current feature map, b_{ij} represents the bias of the jth feature map in the ith layer, $f(\cdot)$ denotes the activation function, K_{i-1} is the number of feature maps in the ith layer, P_i, Q_i and R_i indicate the height, width and depth of the convolution kernel, respectively.

a) Spectral Feature Extraction: As the spectral feature should be extracted independently, point-wise convolution is used and the size of the convolution kernel is set to $1 \times 1 \times 3$, which avoids spatial information getting mixed into it. Therefore, P_i and Q_i are set to 1, (1) can be written as follows:

$$v_{ij}^{xyz} = f\left(\sum_{k=1}^{K_{i-1}}\sum_{r=0}^{R_i-1} w_{ijk}^{pqr} v_{(i-1)k}^{(x+p)(y+q)(z+r)} + b_{ij}\right). \tag{2}$$

With regard to the activation function, we adopt rectified linear units (ReLUs) [21] to regulate the liveness of neurons in the network, contributing to efficient gradient descent and back propagation, as it ameliorates the gradient explosion and gradient vanishing problem of deep CNN training. Furthermore, using ReLU ensures the sparsity of the network comparing with Sigmoid function, which not only has lower computation cost, but also alleviates the overfitting problem. The ReLU can be formulated as

$$f(X) = \max(0, X). \tag{3}$$

Suppose that the input image tensor is Γ, then it can be represented by

$$\Gamma = (x, y, z) \tag{4}$$

and the tensor of spectral direction is the same as the original one

$$\Gamma_{spe} = \Gamma = (x, y, z). \tag{5}$$

Combining (2) and (5), we can express the spectral feature extraction as

$$v_e = f\left(\left[\Gamma_{spe} \otimes w_1\right](x, y) + b_1\right) \tag{6}$$

where $\otimes$ represents the convolution operation.

b) Spatial feature extraction: As seen from Fig. 3, the image tensor need to be permuted when extracting spatial features

$$\Gamma_{\text{spa1}} = (x, z, y) \tag{7}$$

$$\Gamma_{\text{spa2}} = (z, y, x) \tag{8}$$

where Γ_{spa1} and Γ_{spa2} are transposed tensors of spatial direction 1 and spatial direction 2, respectively. As a result, the size of Γ_{spa1} is $H \times C \times W$, and that of Γ_{spa2} is $C \times W \times H$.

When extracting spatial features, the correlation between pixels in both horizontal and vertical directions should be taken into account. On the other hand, to avoid the data volume becoming too large, and to reduce the number of parameters as well, we use depth-wise convolution with the kernel size of $3 \times 3 \times 1$ in both spatial directions. As the kernel size is $3 \times 3 \times 1$, by extension, P_i and Q_i in (1) are set to 3, R_i is set to 1, we obtain

$$v_{ij}^{xyz} = f\left(\sum_{k=1}^{K_{i-1}}\sum_{p=0}^{P_i-1}\sum_{q=0}^{Q_i-1} w_{ijk}^{pqr} v_{(i-1)k}^{(x+p)(y+q)(z+r)} + b_{ij}\right). \tag{9}$$

Similarly, plugging (7) and (8) into (9), the following results are obtained

$$v_{a1} = f\left(\left[\Gamma_{\text{spa1}} \otimes w_2\right](x, z) + b_2\right) \tag{10}$$

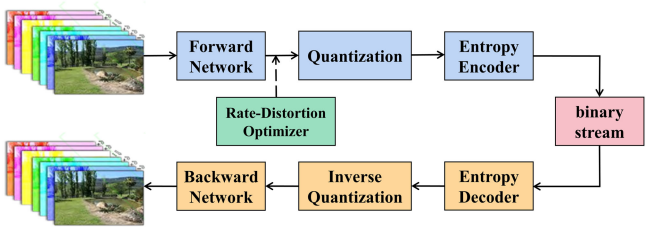

Fig. 4. Diagram of the compression network.

$$v_{a2} = f\left(\left[\Gamma_{\text{spa2}} \otimes w_3\right](z, y) + b_3\right) \tag{11}$$

where v_{a1} is the feature extracted from spatial direction 1, and v_{a2} is of spatial direction 2. That spatial features are extracted from two differrent directions, makes full use of the correlations between rows and between columns of each pixel. It's worth noting that the size of these two tensors is both $H \times W \times C$ after a reverse permute operation at the end of the extraction module.

2) Feature Fusion: After feature extraction, the tensor v_e contains spectral information, v_{a1} and v_{a2} contain spatial information. As mentioned above, they are of the same size of $H \times W \times C$. To ensure the independence and integrity between spectral and spatial features, concatenation is applied as the feature fusion method rather than simple addition. First, v_{a1} and v_{a2} are added together for the preliminary fusion on the spatial dimension, followed by a 1×1 convolution which is operated for dimensionality reduction

$$v_a = v_{a1} + v_{a2} \tag{12}$$

$$u_a = g\left(v_a\right) \tag{13}$$

where $g(\cdot)$ indicates the 1×1 convolution, u_a is the fused spatial feature. Meanwhile, v_e is also processed with a 1×1 convolution so as to lessen the tensor from $H \times W \times C$ to $H \times W \times \frac{C}{2}$, the same as spatial feature

$$u_e = g\left(v_e\right) \tag{14}$$

where u_e denotes the fused spectral feature. In addition, using 1×1 convolution can reduce the parameters while maintaining good performance of the network. Afterward, spectral and spatial features are concatenated, convenient for further processing like downsampling and quantization. Thus, we obtain the joint spectral-spatial feature u_{ss} as

$$u_{ss} = concat\left(u_a; u_e\right). \tag{15}$$

C. Framework of the Proposed Network

The framework of our proposed SSPC network for multispectral image compression is illustrated in Fig. 4. The multispectral images are first fed into the forward network, which consists of feature extraction module and feature fusion module, as mentioned in Section II-B. After obtaining the joint spectral-spatial features, the data are compressed and encoded through quantization and the entropy encoder, to convert to the binary stream for ease of transmission. To recover the image, the bit stream goes through the entropy decoder, inverse quantization, and backward network in succession. The structure of the decoder is symmetrical to the encoder, as well as the backward network is with the forward network, the detailed architecture of which will be demonstrated in the following section.

1) Forward Network and Backward Network: As is stated above, the forward network and the backward network are symmetrical, and these are shown in Figs. 5 and 6, respectively. Further, the architecture of spectral block and spatial block is shown in Fig. 7. This is derived from ResNet, which is conducive to solving the degradation problem of deep network. With shortcut and identity mapping simulation, the input information can be retained and transmitted to the output end with integrity. It not only ensures the normal propagation of the gradient, but also simplifies the learning difficulty, which accelerates the network convergence.

The forward network including SSPC and downsampling. First of all, the multispectral images are simultaneously fed into three parallel feature extraction branches. In spectral direction, the image tensor is directly operated via several spectral blocks to extract independent spectral features. After that, fuse the feature with 1×1 convolution. And it also plays the role of dimensionality reduction, which is convenient for the subsequent data concatenation. As for spatial direction, take direction 1 for example, the image tensor needs to be permuted from $H \times W \times C$ to $H \times C \times W$, and then using corresponding spatial blocks to extract spatial features. Similarly, the image tensor in the direction 2 is firstly transposed to $C \times W \times H$ and then processed with the same amount of spatial blocks to gain spatial features. Before these two parts of data are added together, a reverse permute operation should be performed to ensure that their sizes match. Then, spectral and spatial features are concatenated together after dimensionality reduction, followed by downsampling to reduce the size of the joint spectral-spatial feature maps. At the end of the forward network, the output is normalized and limited to values between 0 and 1 using the sigmoid function. Additionally, activation functions like ReLU and sigmoid can bring nonlinear factors into the network, which enhances the generalization ability of neural networks.

When recovering the image as expressed in Fig. 6, the procedure is the inverse of the forward network. The backward network consists of three upsampling layers, several convolution layers, and SSPC network, among which the upsampling is realized by PixelShuffle [34]. To be precise, PixelShuffle is the equivalent of subpixel convolution that turns input low-resolution images into high-resolution output with periodic shuffling.

2) Quantization and Entropy Coding: Quantization is an indispensable procedure in lossy compression, since it is a many-to-one mapping and is irreversible. The intermediate feature data are transformed into a series of discrete integers by the quantizer. Also, as the rounding function is not differentiable [35], the gradient propagation could be impeded, resulting in the parameters of the network not being normally updated. To deal with the problem, a relaxation operation is used on the rounding function, which can be calculated as

$$u_Q = round\left[\left(2^Q - 1\right) \times u_S\right] \tag{16}$$

where Q denotes the quantization level, u_S is the output of the forward network, and its value is between 0 to 1 after application of the sigmoid function. To balance the trade-off between the bit

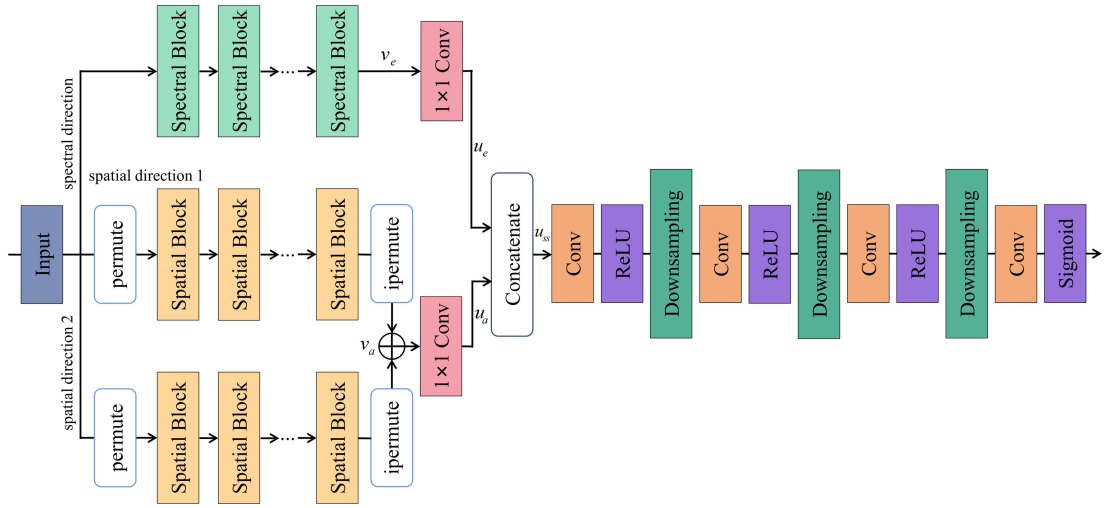

Fig. 5. Forward network.

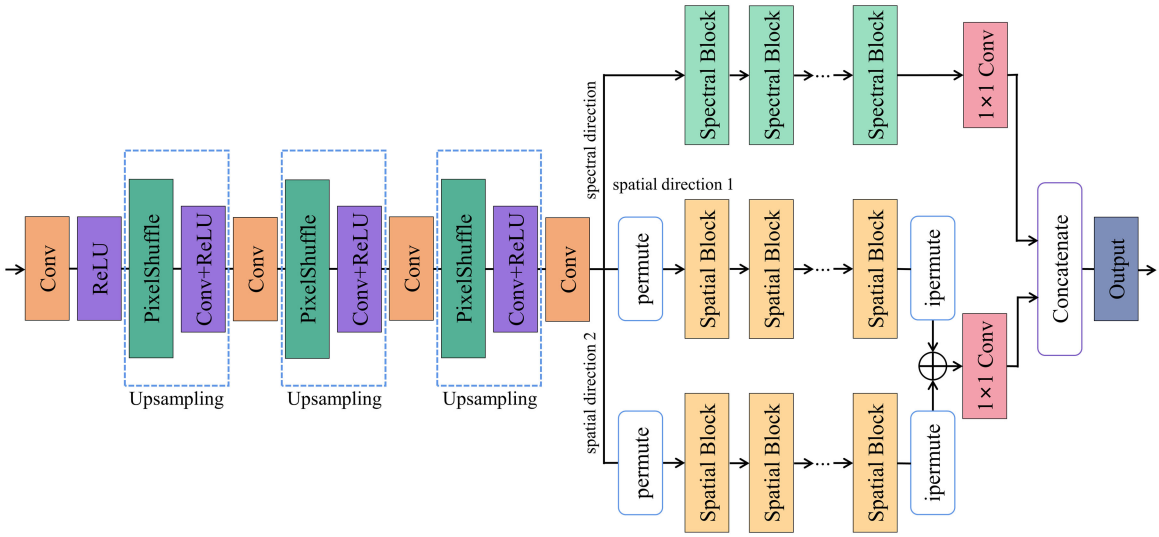

Fig. 6. Backward network.

rate and the information loss, a proper Q needs to be selected. Usually, a bigger quantization level can reduce the information loss during quantization, and increase the bit rate accordingly under the same conditions. As the entropy encoder used in this article is suitable for 256 bit image, after careful deliberation, we choose $Q = 8$ as the quantization level.

This function rounds the data after forward network, but is skipped during backward propagation to pass the gradient directly to the previous layer, which simplifies the complexity of gradient calculation. And the quantization step is fixed at 0.5 for uniform quantization of the data.

Next, the entropy encoder is adopted to convert u_Q to a binary bit stream, where we use ZPAQ [36] as the entropy coding standard and choose "method-6" as the compression pattern. Correspondingly, when recovering the image, the bit stream successively goes through ZPAQ entropy decoder and dequantization to obtain the intermediate data $u_Q/(2^Q - 1)$ which is prepared to be fed into the backward network.

D. Rate-Distortion Optimizer

The purpose of rate-distortion optimization is to select the optimal parameters on the basis of a certain strategy in order to achieve the optimal coding performance. When it comes to image compression, it is most important to strike the balance between bit rate and distortion loss because, in general, the bit rate is inversely proportional to the loss. Hence, the rate-distortion optimizer is introduced as

$$L = L_D + \lambda L_R \tag{17}$$

where L_D represents the distortion loss, the rate loss L_R can controlled by adjusting the penalty λ to obtain different bit rates, and L is the loss function that needs to be minimized through training.

The commonly used distortion loss function includes mean square error (MSE) and mean absolute error (MAE). Although, compared with the former, MAE is more robust to outliers, it also has the fatal disadvantage that the updated gradient is always

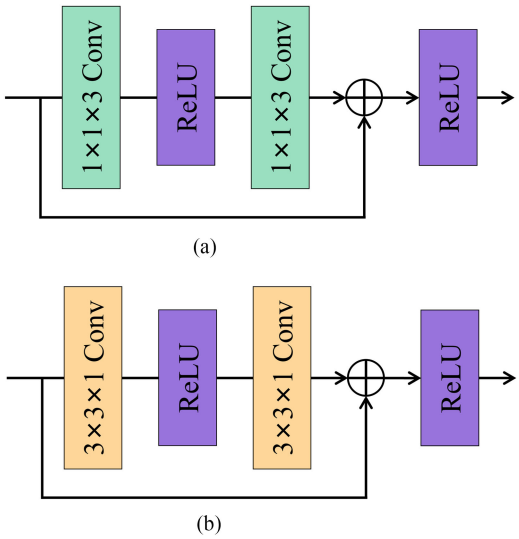

Fig. 7. (a) Spectral block. (b) Spatial block.

the same. That is to say, the gradient remains large even for very small differences, and it does not contribute to learning the model. As a consequence, we adopt MSE to represent L_D

$$L_D = \frac{1}{H \times W \times C} \sum_{x=1}^{H} \sum_{y=1}^{W} \sum_{z=1}^{C} \left\| \Gamma(x,y,z) - \tilde{\Gamma}(x,y,z) \right\|^2 \tag{18}$$

where H, W, and C are height, width and spectral channel number of the multispectral image, respectively, $\Gamma(x,y,z)$ denotes the original image, and $\tilde{\Gamma}(x,y,z)$ denotes the reconstructed image.

As for L_R, we calculate the entropy of the data before quantization as the estimation of the rate loss [37], which can be expressed as

$$L_R = -E\left[\log_2 P(X)\right] \tag{19}$$

where $P(X)$ indicates the discrete probability of the corresponding pixel, E is the expectation. However, as the derivatives of the rounding function are almost zero everywhere, the entropy calculation is also nondifferentiable, and the entropy is discrete. To make it differentiable, a piecewise linear approximation is applied. First, scalar quantization is used at integral point, and then sampling is operated to make the entropy continuous with interpolation. So, (19) can be written as

$$L_R = -E\left[\log_2 P_q\right] \tag{20}$$

$$P_q = \int_{x-\frac{1}{2}}^{x+\frac{1}{2}} P_d(x)\, dx \tag{21}$$

where $P_d(\cdot)$ is the probability density function of the original data. During training, the data distribution becomes compact, further improving the performance of the network.

III. Experimental Settings and Training

A. Datasets

The datasets we used include two types of multispectral images with different number of bands, one of which is seven-band and the other is eight-band. Among them, the seven-band dataset comes from Landsat8 satellite that contains rich texture features of ground objects, including different seasons and different terrains, to make sure the diversity of the data. We select the first seven bands with spatial resolution of 30 m and wavelengths ranging from 0.433 to 2.300 μm, and then synthesize them into one multispectral image. To facilitate training, the image is cropped into blocks with the size of 128×128, and the blocks with clouds and black areas need to be abandoned. The training set contains more than 30 000 images, ensuring that there are enough features for learning to avoid the network from overfitting. Besides, 17 representative images distinguished from the training set are randomly selected from the original data set to verify the network performance as the test set, with a size of 512×512.

Likewise, the eight-band dataset from WorldView-3 are also divided into training set and test set, with image size of 128×128 and 512×512 , respectively. In addition, the training set has about 17 000 images and test set has 14 images. Although the eight-band dataset is relatively small compared with the seven-band dataset due to the rare public data of the satellite, this still guarantees that the training set contains various terrains under different weather conditions to ensure the diversity of the feature. Also, there are no identical images in the training and test datasets.

TABLE I
Parameter Settings

Parameter	Value
Batch size	32
Epoch	1000; 80 (with rate-distortion optimizer)
Learning rate	1e-4, 1e-5
Intermediate data size	16×16×48
λ	1e-5, 3e-5, 5e-5, 7e-5, 9e-5, 2e-4, 3e-4

B. Equipment and Parameter Settings

We adopt the Adam optimizer to train the model and update the parameter of the network. To accelerate the network convergence, the learning rate is initially set to 0.0001. According to [38], the step-down LR method, fix the learning rate at 0.0001 until the loss function drops to a certain level, then adjust it to 0.00001. The parameter settings of the training network are given in Table I where epoch is an estimate. For random initialization, the epoch is around 1000 until the network finally converges. And after adding rate-distortion optimizer, it only needs about 80 epochs because the optimal model of the first step can be directly substituted into the network.

Moreover, the equipment information we used to implement the experiment is given in Table II.

C. Training Process

To make the encoder and the decoder effectively collaborate, an end-to-end learning algorithm is introduced in this article. First of all, the weights of the network are randomly initialized,

TABLE II
EQUIPMENT INFORMATION

DELL computer	Details
Operating system	64-bit win7
RAM	32.0 GB
CPU	Intel(R) Xeon(R) CPU E5-2620 v3 @ 2.40 GHz
GPU	NVIDIA GeForce RTX 2080 Ti

with Adam optimizer, the parameters of the network can be continuously updated as iterations increase. The training process is split into two stages. In the first stage, only distortion loss is considered as the loss function, namely MSE. The goal of the optimization is to minimize the loss function, which can be expressed as

$$\left(\tilde{\theta}_1, \tilde{\theta}_2, \tilde{\theta}_3\right) = \arg\min_{\theta_1,\theta_2,\theta_3} \left\| Re\left(Qu\left(\begin{array}{l} Se\left(\theta_1, x\right) + \\ Sa_1\left(\theta_2, x\right) + \\ Sa_2\left(\theta_3, x\right) \end{array} \right) \right) - x \right\|^2 \quad (22)$$

where x is the input original image, θ_1, θ_2, and θ_3 are the network parameters of three directions, respectively. That is, $Se(\cdot)$ denotes the feature extraction network in spectral direction, $Sa_1(\cdot)$ and $Sa_2(\cdot)$ are of spatial directions 1 and 2, respectively. $Qu(\cdot)$ represents the quantizer, and $Re(\cdot)$ represents the decoder. To simplify the expression, an auxiliary variable x_m is introduced

$$x_m\left(\theta\right) = Se\left(\theta_1, x\right) + Sa_1\left(\theta_2, x\right) + Sa_2\left(\theta_3, x\right) \quad (23)$$

$$\theta = \left(\theta_1, \theta_2, \theta_3\right). \quad (24)$$

Accordingly

$$\tilde{\theta} = \arg\min_{\theta} \left\| Re\left(Qu\left(x_m\left(\theta\right)\right)\right) - x \right\|^2. \quad (25)$$

During the backward propagation, the quantizer can be skipped. Therefore, (25) can be written as

$$\tilde{\theta} = \arg\min_{\theta} \left\| Re\left(x_m\left(\theta\right)\right) - x \right\|^2. \quad (26)$$

After finishing the first stage of training, the rate loss is also brought into the loss function. As a result, combine (19) and (25) to obtain the final optimization procedure

$$\tilde{\theta} = \arg\min_{\theta} \left\{ \begin{array}{l} \left\| Re\left(Qu\left(x_m\left(\theta\right)\right)\right) - x \right\|^2 - \\ \lambda E\left[\log_2\left(\int_{x-\frac{1}{2}}^{x+\frac{1}{2}} P_d\left(\theta, x\right) dx\right)\right] \end{array} \right\}. \quad (27)$$

When the loss function no longer declines, the optimization reaches the optimal solution. Furthermore, different value of λ is successively set to get multiple bit rates, which are given in Table I.

IV. RESULTS AND DISCUSSION

In this section, we display the experimental results comparing with JPEG2000 and 3-D-SPIHT, including the PSNR curves at different bit rates, and the SS that represents the degree of spectral information recovery. For further validate the superiority of our proposed framework, another CNN-based compression algorithm proposed in [27] is also added to the experiment, which can be seen as one-directional CNN corresponding to our polydirectional CNN.

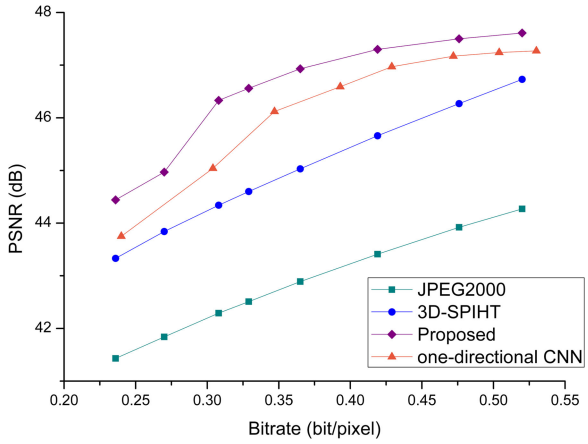

Fig. 8. Average PSNR curve of seven-band test set.

TABLE III
AVERAGE PSNR OF SEVEN-BAND TEST SET

Bitrate	JPEG2000	3D-SPIHT	One-directional CNN	Proposed
0.236	41.43	43.33	43.75	**44.44**
0.270	41.84	43.84	44.37	**44.97**
0.308	42.29	44.34	45.04	**46.33**
0.329	42.51	44.60	45.59	**46.56**
0.365	42.89	45.03	46.32	**46.93**
0.419	43.41	45.66	46.77	**47.30**
0.476	43.92	46.27	47.17	**47.50**
0.520	44.27	46.73	47.27	**47.61**

A. Spatial Information Recovery

Fig. 8 shows the comparison result of four algorithms tested on the seven-band test set. As seen below, the proposed method and the CNN-based algorithm are both obviously superior to JPEG2000 and 3-D-SPIHT, which indicates that the application of deep learning in image compression is highly effective. When the bitrate ranges from 0.3 to 0.4 bit/pixel, the proposed SSPC algorithm obtains the most prominent performance and has about 2 dB better than 3-D-SPIHT and 4 dB better than JPEG2000. Moreover, the image compression algorithm based on one-directional CNN also obtains excellent results, but still pales in comparison with our proposed method. This result shows that the polydirectional CNN is indeed valid and can enhance the performance of the network compared with the normal CNN.

To show it more precisely, the bitrates (bit/pixel) and corresponding PSNR (dB) are all given in Table III.

To ensure the universality of the network, we also carried out experiments on the eight-band test set. As shown in Fig. 9 and Table IV, the one-directional-CNN-based algorithm surpasses 3-D-SPIHT and JPEG2000 by nearly 2.5–4 and 5–7.5 dB, respectively, but it is still inferior to our proposed algorithm by about 0.5–1.4 dB.

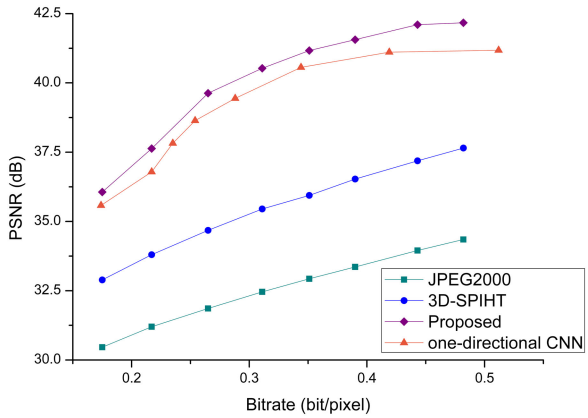

Fig. 9. Average PSNR curve of eight-band test set.

TABLE IV
AVERAGE PSNR OF EIGHT-BAND TEST SET

Bitrate	JPEG2000	3D-SPIHT	One-directional CNN	Proposed
0.175	30.46	32.89	35.88	**36.06**
0.217	31.20	33.80	36.29	**37.63**
0.265	31.86	34.68	38.99	**39.63**
0.311	32.46	35.45	39.88	**40.53**
0.351	32.93	35.94	40.66	**41.17**
0.390	33.36	36.53	40.86	**41.56**
0.443	33.95	37.19	41.13	**42.10**
0.482	34.35	37.65	41.16	**42.17**

Combining the results of Figs. 8 and 9, it can be found that the more bands the image has, the more obvious the advantage of the CNN-based compression method will be.

B. Spectral Information Recovery

In order to comprehensively evaluate the performance of the proposed scheme, the measurement criteria should give consideration to both spatial information recovery and spectral information recovery. Therefore, in addition to the general PSNR, SS is introduced as an extra evaluation criterion.

Traditional methods for spectral information measurement, such as Euclidean distance and spectral angle (SA), only calculate the magnitude (luminance) or distance between two spectra. As for SS in [39], it combines RMSE and Pearson correlation coefficient together, which can be formulated as

$$\mathrm{SS} = \sqrt{\mathrm{RMSE}_{X,Y}^2 + \left(1 - \mathrm{corr}_{X,Y}^2\right)^2} \tag{28}$$

where

$$\mathrm{RMSE}_{X,Y} = \sqrt{\frac{1}{n_z}\sum_{z}\left[\Gamma(x,y,z) - \tilde{\Gamma}(x,y,z)\right]^2} \tag{29}$$

$$\mathrm{corr}_{X,Y} = \frac{\sum_{z}\left(I(x,y,z)\right)\left(\tilde{I}(x,y,z)\right)}{(n_z-1)\,\sigma_{\Gamma(x,z,\cdot)}\sigma_{\tilde{\Gamma}(x,z,\cdot)}} \tag{30}$$

$$I(x,y,z) = \Gamma(x,y,z) - \mu_{\Gamma(x,y,\cdot)} \tag{31}$$

$$\tilde{I}(x,y,z) = \tilde{\Gamma}(x,y,z) - \mu_{\tilde{\Gamma}(x,y,\cdot)} \tag{32}$$

TABLE V
AVERAGE SS OF SEVEN-BAND TEST SET

Bitrate (bit/pixel)	JPEG2000	3D-SPIHT	One-directional CNN	Proposed
0.236	572.50	467.12	454.81	**343.36**
0.270	546.13	440.37	409.85	**327.36**
0.308	518.68	415.70	364.89	**300.95**
0.329	505.61	403.32	453.98	**294.58**
0.365	483.65	383.68	331.84	**285.51**
0.419	455.19	357.47	313.92	**276.51**
0.476	429.59	333.95	301.53	**271.30**
0.520	412.17	316.53	298.01	**270.01**

TABLE VI
AVERAGE SS OF EIGHT-BAND TEST SET

Bitrate (bit/pixel)	JPEG2000	3D-SPIHT	One-directional CNN	Proposed
0.175	1982.08	1497.84	1052.29	**997.93**
0.217	1821.40	1350.03	1003.73	**835.00**
0.265	1687.31	1221.60	733.06	**698.02**
0.311	1577.09	1116.96	662.17	**633.57**
0.351	1493.97	1056.69	608.11	**593.54**
0.390	1423.10	987.87	603.85	**592.86**
0.443	1331.32	917.49	574.43	**556.24**
0.482	1272.29	870.90	574.65	**551.26**

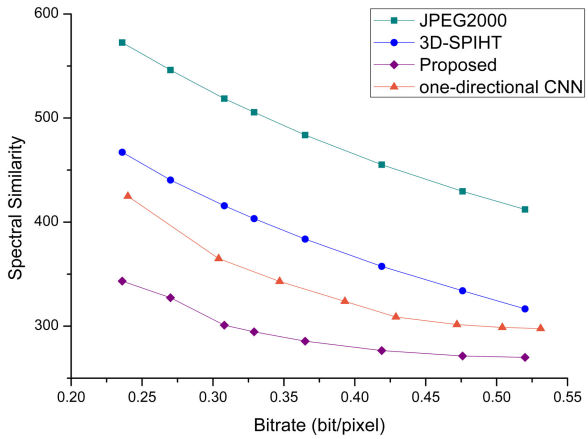

Fig. 10. Average SS curve of seven-band test set.

where $\Gamma(x,y,z)$ and $\tilde{\Gamma}(x,y,z)$ denote the pixel value at spatial position (x,y,z) of the original image and the recovered image, respectively. Besides, $\sigma_{\Gamma(x,y,\cdot)}$ is the standard deviation at (x,y) of all band pixels, and similarly, $\mu_{\Gamma(x,y,\cdot)}$ is the mean value at (x,y) of all band pixels of the image. From (28), we can conclude that the smaller the SS is, the more similar two spectra are.

Similar to the comparison experiment of PSNR, we first conduct test on the seven-band dataset, and the results are given in Table V. It turns out that applying deep learning to multispectral image compression not only contributes to the spatial information recovery, but also has a remarkable effect on the spectral information retention. Furthermore, our polydirectional CNN extracts the feature of multispectral images through three directions, which can better retain the spectral information with integrity, and this is reflected in Fig. 10 where the SS value of our

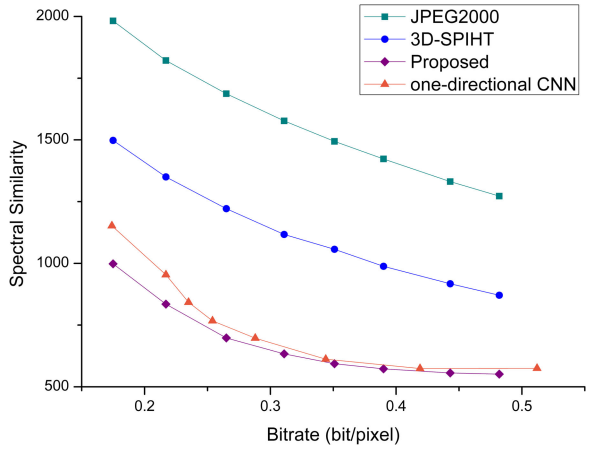

Fig. 11. Average SS curve of eight-band test set.

TABLE VII
AVERAGE SA OF SEVEN-BAND TEST SET

Bitrate (bit/pixel)	JPEG2000	3D-SPIHT	One-directional CNN	Proposed
0.236	0.0346	0.0315	0.0269	**0.0239**
0.270	0.0331	0.0298	0.0247	**0.0225**
0.308	0.0316	0.0284	0.0225	**0.0210**
0.329	0.0309	0.0276	0.0223	**0.0205**
0.365	0.0298	0.0263	0.0206	**0.0190**
0.419	0.0282	0.0249	0.0195	**0.0183**
0.476	0.0269	0.0237	0.0187	**0.0180**
0.520	0.0260	0.0227	0.0186	**0.0177**

proposed algorithm is always lower than that of one-directional CNN.

Similarly, SS curve is then calculated on the sight-band test set, as shown in Fig. 11. And the corresponding data are listed in Table VI. As the number of bands increases, the value of SS also increases significantly, which indicates that there is a strong correlation between different spectra of the multispectral image. This correlation is almost ignored when using traditional compression methods. Compared with one-directional CNN, our polydirectional CNN has a more obvious superiority as the bitrate decreases.

In addition to the SS, SA [40] is also chosen as another metric to weigh the spectral information recovery. Considering two spectra as two vectors, and SA is the angle between these two vectors, to measure the similarity of these spectra. The smaller the absolute value of SA, the more similar the two spectra are. The formula of SA can be written as (33), and this ranges between -1 and 1

$$SA\left(\Gamma,\tilde{\Gamma}\right)=\cos^{-1}\left(\frac{\sum_{\lambda}\Gamma\left((x,y,\lambda)\cdot\tilde{\Gamma}\left(x,y,\lambda\right)\right)}{\sqrt{\sum_{\lambda}\Gamma^{2}\left(x,y,\lambda\right)\sum_{\lambda}\tilde{\Gamma}^{2}\left(x,y,\lambda\right)}}\right). \tag{33}$$

After further testing, SA values of the recovered images of seven-band and eight-band test sets are both obtained and listed in Tables VII and VIII, respectively. Obviously, our method achieves the best performance at all bit rates

TABLE VIII
AVERAGE SA OF EIGHT-BAND TEST SET

Bitrate (bit/pixel)	JPEG2000	3D-SPIHT	One-directional CNN	Proposed
0.175	0.0603	0.0815	0.0405	**0.0393**
0.217	0.0581	0.0724	0.0379	**0.0367**
0.265	0.0551	0.0652	0.0345	**0.0331**
0.311	0.0527	0.0625	0.0326	**0.0315**
0.351	0.0511	0.0616	0.0308	**0.0296**
0.390	0.0495	0.0591	0.0306	**0.0294**
0.443	0.0474	0.0550	0.0305	**0.0291**
0.482	0.0460	0.0528	0.0304	**0.0290**

C. Comparison of Visual Effects

Here, we use a more intuitive visual effect comparison to distinguish the advantages and disadvantages of several compression methods. To be specific, we select four representative images each from two test sets and display the grayscale image of the third band of these images to show the differences more clearly. For better visual effect, the images with the bitrate of around 0.4 bit/pixel are selected.

As shown in Fig. 12, it is plainly visible that there are dense texture details in the original images. However, with JPEG2000 and 3-D-SPIHT, these details are blurred seriously, and obvious block effects come into being when using JPEG2000. Two CNN-based algorithms, on the other hand, both reconstruct the image well with the texture details. Nonetheless, according to Figs. 8 and 10, the performance of the one-directional CNN algorithm degrades rapidly as the bitrate reduces to 0.35 bit/pixel and lower. On the contrary, our proposed method performs more stably at all bitrates. For easy observation, we enlarge the details of two test images in Fig. 12, as shown in Figs. 13 and 14.

When it comes to eight-band test set, due to its richer texture and margin information, the differences between these four algorithms become more obvious. As seen from Fig. 15, the texture and margin information of the roads and buildings is extremely blurred in the recovered image of JPEG2000 and 3-D-SPIHT, and even one-directional CNN algorithm also tends to lose its edge in comparison with our proposed method. With the combined result of seven-band test set together, the eight-band experiment proves that these algorithms will be extremely inferior when dealing with multispectral images with more bands, if they do not pay enough attention to spectral correlation. This also proves the correctness and effectiveness of our SSPC algorithm. Also, partial enlarged details of two of the test images are shown in Figs. 16 and 17.

V. CONCLUSION

In this article, a novel multispectral image compression framework using spectral-spatial feature extraction method with polydirectional CNN is proposed. The innovation of this algorithm lies in the approach that the features of the image are extracted from three different directions, which not only preserves spatial features with integrity, but also ensures the independence of spectral features. As the multispectral images are volumetric

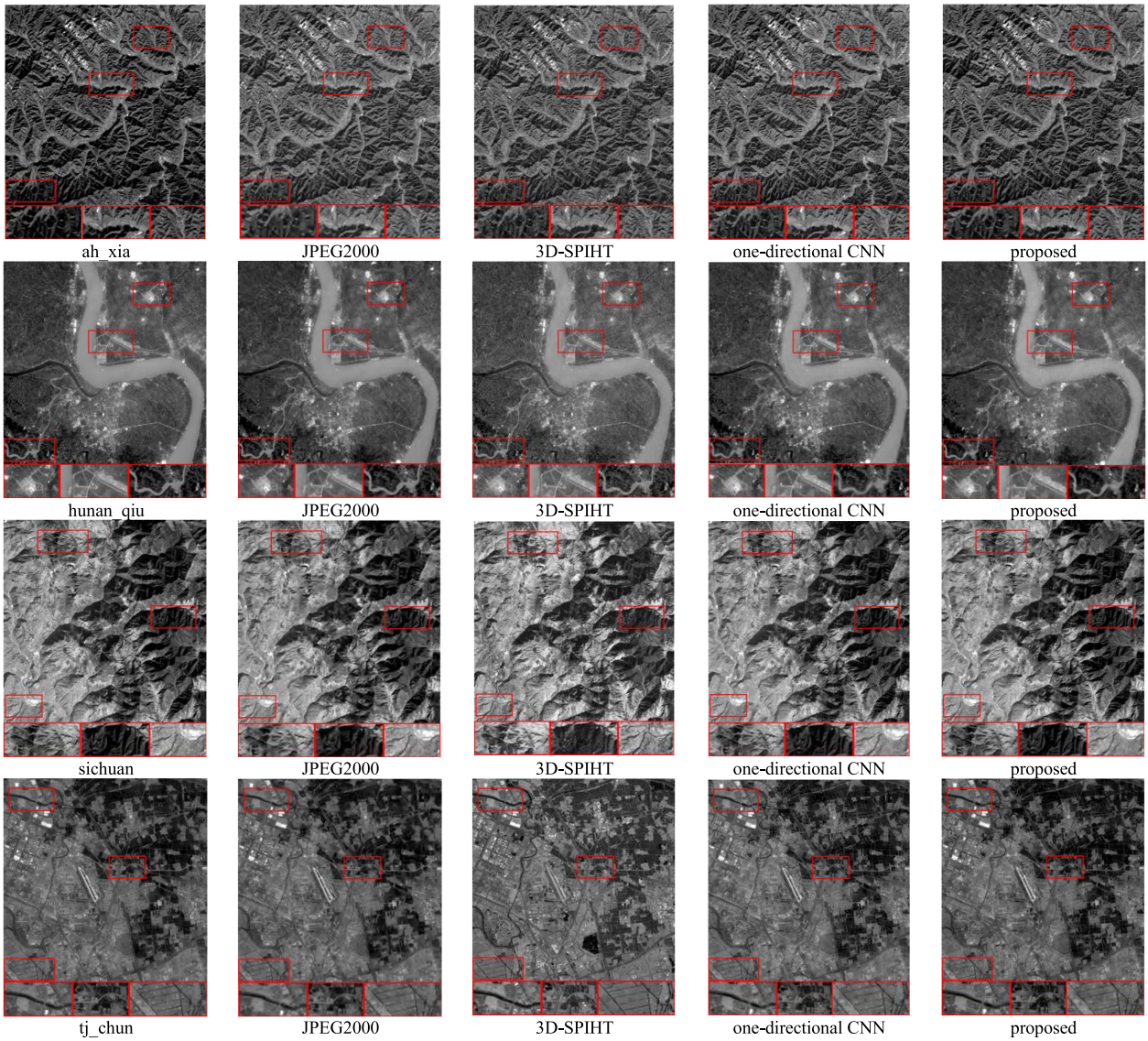

Fig. 12. Visual comparison of the recovered images of seven-band (each row represents the same image).

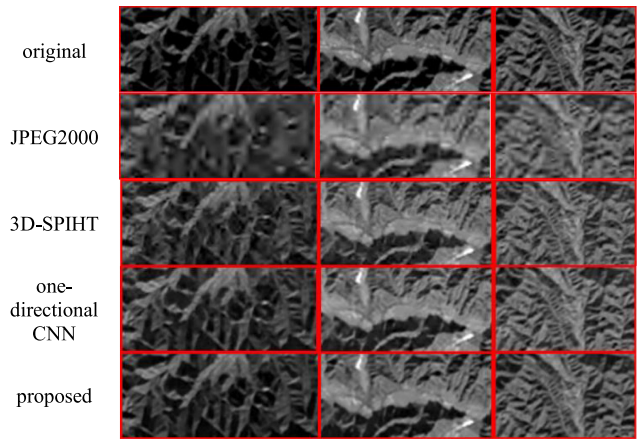

Fig. 13. Partial enlarged view of ah_xia.

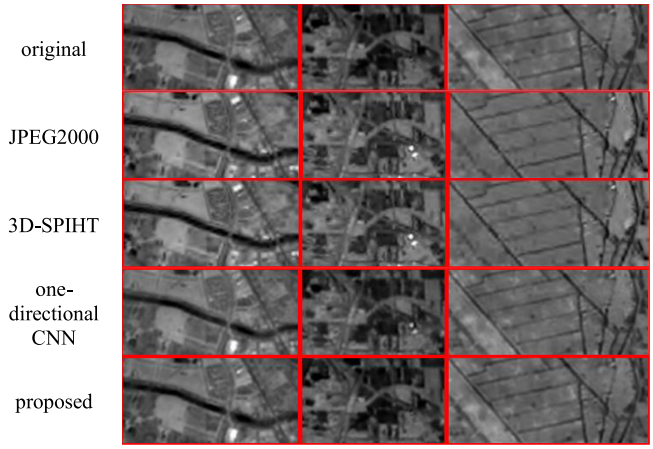

Fig. 14. Partial enlarged view of tj_chun.

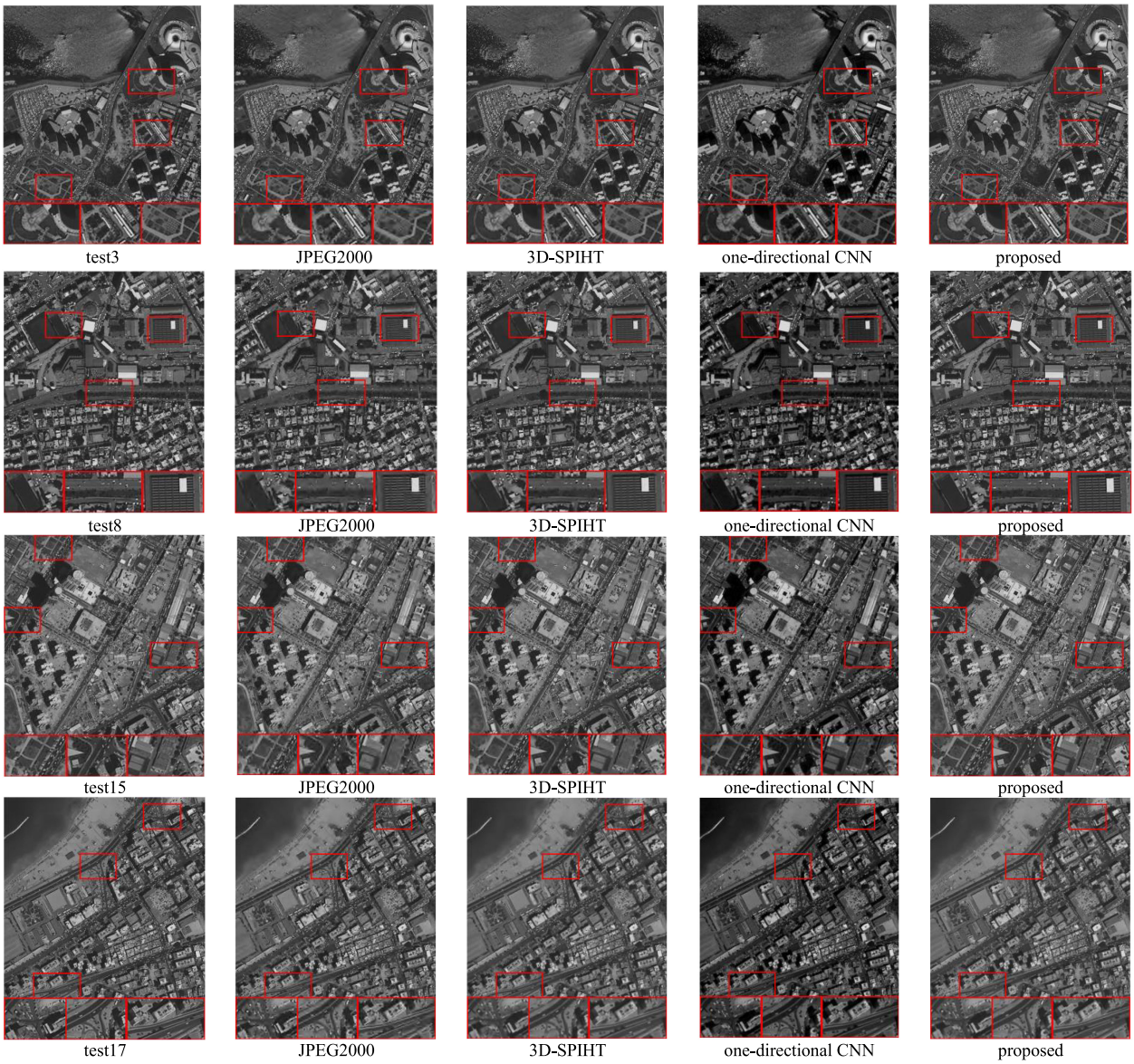

Fig. 15. Visual comparison of the recovered images of eight-band (each row represents the same image).

[41], employing point-wise convolution and depth-wise convolution together can make it possible to achieve the highest performance of the model. The spectral-spatial features are extracted separately and then concatenated together, and the fusing operation makes sure that two parts of information do not interfere with each other. Likewise, the spectral and spatial features are recovered when reconstructing the image, resulting in images with high quality. When validating the performance of the framework, we have tested and illustrated its effectiveness from three aspects: spatial information recovery; spectral information recovery; and visual effect. The results have shown that the SSPC algorithm outperforms JPEG2000, 3-D-SPIHT and one-directional-CNN-based algorithm. Thanks to the outstanding learning ability of CNN, it is easy to capture the latent representation of the multispectral image itself. Also, with the design of polydirectional structure, both the spectral and spatial correlations can be obtained, resulting in great performance in spectral and spatial recovery. In particular, the SSPC algorithm has a greater advantage over all the other three methods as the bitrate drops off, which testifies that the SSPC algorithm has strong robustness. Finally, the experimental results on the eight-band test set prove that for multispectral images, the use of rich spectral correlation is the trend for multispectral image compression, and has significant prospect for future development.

REFERENCES

[1] S. K. Roy, S. Manna, T. Song, and L. Bruzzone, "Attention-based adaptive spectral-spatial kernel ResNet for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 9, pp. 7831–7843, Dec. 2020, doi: 10.1109/TGRS.2020.3043267.

[2] X. Zhong*et al.*, "Attention_FPNet:Two-branch remote sensing image pansharpening network based on attention feature fusion," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 11879–11891, Nov. 2021, doi: 10.1109/JSTARS.2021.3126645.

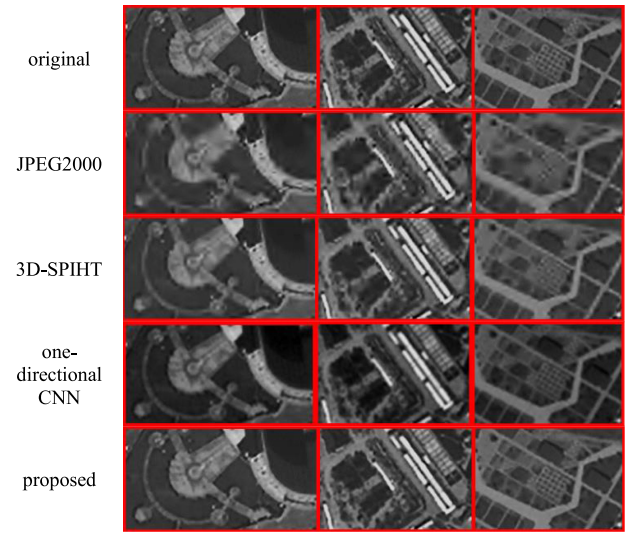

Fig. 16. Partial enlarged view of test 3.

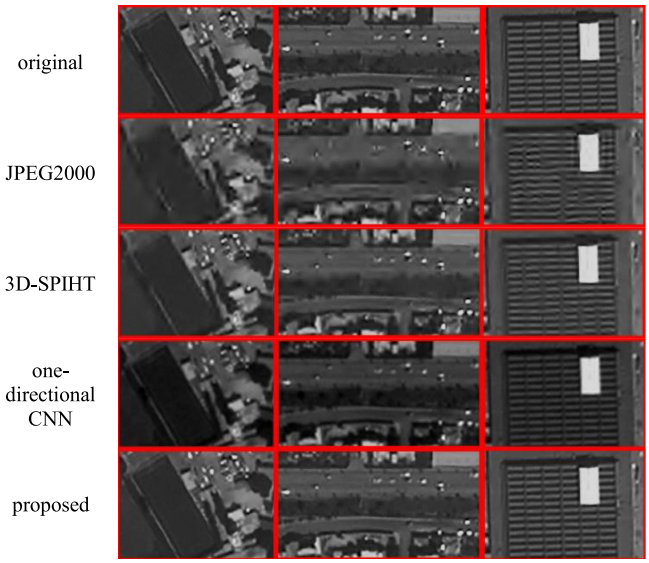

Fig. 17. Partial enlarged view of test 8.

[3] X. Yang and Y. Yu, "Estimating soil salinity under various moisture conditions: An experimental study," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2525–2533, May 2017.
[4] B. Luo, C. Yang, J. Chanussot, and L. Zhang, "Crop yield estimation based on unsupervised linear unmixing of multidate hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 1, pp. 162–173, Jan. 2013.
[5] R. Tao, X. Zhao, W. Li, H. Li, and Q. Du, "Hyperspectral anomaly detection by fractional Fourier entropy," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 12, pp. 4920–4929, Dec. 2019.
[6] X. Lu, W. Zhang, and X. Li, "A hybrid sparsity and distance-based discrimination detector for hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 3, pp. 1704–1717, Mar. 2018.
[7] Z. L. Zhou, "Research on hyperspectral image compression method," M.S. thesis, Nanjing Univ. Science and Technology, Nanjing, China, 2008.
[8] G. Gelli and G. Poggi, "Compression of multispectral images by spectral classification and transform coding," *IEEE Trans. Image Proc.*, vol. 8, no. 4, pp. 476–489, Aug. 1999, doi: 10.1109/83.753736.
[9] Y. S. Li, C. K. Wu, J. Chen, and L. B. Xiang, "Spectral satellite image compression based on wavelet transform," *Acta Opt. Sinica*, vol. 21, no. 6, pp. 691–695, 2001.
[10] Y. J. Nian, Y. Liu, and Z. Ye, "Pairwise KLT-based compression for multispectral images," *Sens. Imag.*, vol. 17, no. 1, pp. 1–15, Dec. 2016, doi: 10.1007/s11220-016-0128-5.
[11] G. Motta, F. Rizzo, and J. A. Storer, "Partitioned vector quantization: Application to lossless compression of hyperspectral images," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Jul. 2003, pp. 553–556.
[12] J. Wang, "Hyper-spectral image lossless compressing based on spectral DPCM and intra-DPCM," *Chin. Opt.*, vol. 6, no. 6, pp. 863–867, 2013, doi: 10.3788/CO.20130606.863.
[13] J. Mielikainen and P. Toivanen, "Clustered DPCM for the lossless compression of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 12, pp. 2943–2946, Dec. 2003.
[14] P. Hao and Q. Shi, "Reversible integer KLT for progressive-to-lossless compression of multiple component images," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2003, pp. 1–633, doi: 10.1109/ICIP.2003.1247041.
[15] G. P. Abousleman, M. W. Marcellin, and B. R. Hunt, "Compression of hyperspectral imagery using the 3-D DCT and hybrid DPCM/DCT and entropy-constrained trellis coded quantization," in *Proc. Conf. Data Compression*, Mar. 1995, pp. 26–35, doi: 10.1109/DCC.1995.515522.
[16] W. Sweldens, "The lifting scheme: A custom-design construction of biorthogonal wavelets," *Appl. Comput. Harmonic Anal.*, vol. 3, no. 2, pp. 186–200, 1996.
[17] P. L. Dragotti, G. Poggi, and A. R. P. Ragozini, "Compression of multispectral images by three-dimensional SPIHT algorithm," *IEEE Trans. Geosci. Remote Sens.*, vol. 38, no. 1, pp. 416–428, Jan. 2000, doi: 10.1109/36.823937.
[18] X. L. Tang and W. A. Pearlman, "Three-dimensional wavelet-based compression of hyperspectral images," in *Hyperspectral Data Compression*, New York, NY, USA: Springer, 2006, pp. 273–308.
[19] C. Dong, C. C. Loy, and K. M. He, "Learning a deep convolutional network for image super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 184–199.
[20] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998, doi: 10.1109/5.726791.
[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. 25th Int. Conf. Neural Inf. Process. Syst.*, Dec. 2012, pp. 1097–1105.
[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *Comput. Sci.*, 2014, *arXiv:1409.1556*.
[23] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.
[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.
[25] H. J. Liu, T. Chen, Q. Shen, T. Yue, and Z. Ma, "Deep image compression via end-to-end learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2575–2578.
[26] F. Jiang, W. Tao, S. Liu, J. Ren, X. Guo, and D. Zhao, "An end-to-end compression framework based on convolutional neural networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 10, pp. 3007–3018, Aug. 2018, doi: 10.1109/TCSVT.2017.2734838.
[27] F. Kong, Y. Zhou, Q. Shen, and K. Wen, "End-to-end multispectral image compression using convolutional neural network," *Chin. J. Lasers*, vol. 46, no. 10, pp. 1009001–1, 2019, doi: 10.3788/CJL201946.1009001.
[28] H. Patel and K. P. Upla, "A shallow network for hyperspectral image classification using an autoencoder with convolutional neural network," *Multimed Tools Appl.*, vol. 81, no. 1, pp. 695–714, Jan. 2022, doi: 10.1007/s11042-021-11422-w.
[29] A. Sellami and S. Tabbone, "Deep neural networks-based relevant latent representation learning for hyperspectral image classification," *Pattern Recognit.*, vol. 121, 2022, Art. no. 108224.
[30] Z. Zhong, J. Li, Z. Luo, and M. Chapman, "Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 847–858, Feb. 2018, doi: 10.1109/TGRS.2017.2755542.
[31] S. K. Roy, G. Krishna, S. R. Dubey, and B. B. Chaudhuri, "HybridSN: Exploring 3-D–2-D CNN feature hierarchy for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 277–281, Feb. 2020, doi: 10.1109/LGRS.2019.2918719.
[32] J. Yang, Y. Zhao, and J. C. Chan, "Learning and transferring deep joint spectral–spatial features for hyperspectral classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 8, pp. 4729–4742, Aug. 2017, doi: 10.1109/TGRS.2017.2698503.
[33] Z. Xue, X. Yu, B. Liu, X. Tan, and X. Wei, "HResNetAM: Hierarchical residual network with attention mechanism for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 3566–3580, Mar. 2021, doi: 10.1109/JSTARS.2021.3065987.

[34] W. Z. Shi *et al.*, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 1874–1883.
[35] G. Toderici *et al.*, "Variable rate image compression with recurrent neural networks," 2015, *arXiv: 1511.06085v2*.
[36] M. Mahoney, "ZPAQ compressor," 2016. [Online]. Available: http://mattmahoney.net/dc/zpaq.html
[37] F. Kong, S. Zhao, Y. Li, D. Li, and Y. Zhou, "A residual network framework based on weighted feature channels for multispectral image compression," *Ad Hoc Netw.*, vol. 107, 2020, Art. no. 102272.
[38] M. Li, W. Zuo, S. Gu, D. Zhao, and D. Zhang, "Learning convolutional networks for content-weighted image compression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3214–3223, doi: 10.1109/CVPR.2018.00339.
[39] Y. S. Li, F. Q. Kong, C. K. Wu, and J. Lei, "Interference multi-spectral image compression based on distributed source coding," *Acta Opt. Sinica*, vol. 28, no. 8, pp. 1463–1468, 2008, doi: 10.1117/12.793507.
[40] E. Christophe, D. Leger, and C. Mailhes, "Quality criteria benchmark for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 9, pp. 2103–2114, Aug. 2005, doi: 10.1109/TGRS.2005.853931.
[41] M. Khodadadzadeh, X. Ding, P. Chaurasia, and D. Coyle, "A hybrid capsule network for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 11824–11839, Nov. 2021, doi: 10.1109/JSTARS.2021.3126427.

Fanqiang Kong received the B.S. degree in optoelectronics technology from Xidian University, Xi'an, China, in 2002, the M.S. degree in communication and information system and the Ph.D. degree in information and communication engineering from Xidian University, Xi'an, China, in 2005 and 2008, respectively.

He is currently an Associate Professor with the College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing, China. His research interests include spectral image coding and image analysis, artificial intelligence, and pattern recognition.

Kedi Hu received the B.S. degree in information engineering in 2019 from the College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing, China, where she is currently working toward the M.S. degree in communication and Information System.

Her current research interests include multispectral image compression and deep learning.

Yunsong Li received the Ph.D. degree in signal processing from Xidian University, Xi'an, China, in 2002.

He is currently a Professor of communication with Xidian University, Xi'an, China. His research interests include spectral image coding and image analysis.

Dan Li received the B.S., M.S., and Ph.D. degrees in control science and engineering from the Harbin Institute of Technology, Harbin, China, in 2012, 2014, and 2018, respectively.

Since 2018, she has been a Lecturer with the College of Astronautics, Nanjing University of Aeronautics and Astronautics. Her research interests include Hyperspectral image classification, signal processing, sparse sampling and reconstruction technology.

Xin Liu (Senior Member, IEEE) received the M.Eng. and Ph.D. degrees in communication engineering from the Harbin Institute of Technology, Harbin, China, in 2008 and 2012, respectively.

He is currently an Associate Professor with the School of Information and Communication Engineering, Dalian University of Technology, Dalian, China. From 2012 to 2013, he was a Research Fellow with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. From 2013 to 2016, he was a Lecturer with the College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing, China.

Tariq S. Durrani received M.Sc. and Ph.D. degrees from the University of Southampton, in 1967 and 1970 respectively.

He is currently a Full Professor with the University of Strathclyde, Scotland, U.K., where he was the Deputy Principal, from 2000 to 2006.

Mr. Durrani is the Past Vice President of the Royal Society of Edinburgh and the IEEE and the Past President of the IEEE Signal Processing Society and the IEEE Engineering Management Society. He has been the General Chair of several flagship international conferences, including IEEE ICASSP-89, Transputers-91, IEEE IEMC-02, European Universities Convention-06, and IEEE ICC-07. He is a Fellow of the U.K. Royal Academy of Engineering, the U.K. Royal Society of Edinburgh, and the IEEE. He is also a Foreign Fellow of the Chinese Academy of Sciences and the U.S. National Academy of Engineering.