Hyperspectral Image Classification With Re-Attention Agent Transformer and Multiscale Partial Convolution

Junding Sun, Hongyuan Zhang, Jianlong Wang, Haifeng Sima, and Shuanggen Jin, *Senior Member, IEEE*

***Abstract*—Convolutional neural networks (CNNs) focus solely on extracting local features, lacking the ability to capture global spectral-spatial information. Meanwhile, Transformers effectively learn the overall distribution and mutual relationships of spectral features but overlook the extraction of local spatial features. To fully leverage the complementary advantages of both techniques, the article proposes a re-attention agent transformer and multiscale partial convolution (RAT-MPC) for hyperspectral image classification. It effectively utilizes the local learning capability of CNNs and the long-range modeling ability of Transformers. Specifically, the multiscale spatial-spectral feature learning module employs a strategy of split, refactoring, fusion to extract shallow feature information. Subsequently, the dual branch feature processing module handles the obtained features from both local and global perspectives. On one hand, the re-attention agent transformer branch is employed to learn complex global spectral relationships. On the other hand, multiscale partial convolutions are utilized to further learn abstract spatial features. Finally, the multilevel feature fusion attention module is designed to fully use features from different receptive fields and depths. In addition, it incorporates an enhanced coordinate attention mechanism to reinforce spatial detail features. To evaluation the proposed RAT-MPC effectiveness, 5%, 0.7%, and 0.1% of labeled samples are selected from the Indian Pines (IP), Pavia University (PU), and WHU-Hi-LongKou (LK) datasets, respectively. The experimental results demonstrate that the proposed network exhibited exceptional classification performance, achieving overall accuracies of 96.66%, 98.20%, and 98.44% on the IP, PU, and LK datasets, respectively. Compared with the latest CNN-Transformer related method DBCTNet, the proposed method achieves improvements of 1.36%, 0.68%, and 1.38% in overall accuracies, respectively.**

***Index Terms*—Attention mechanism, convolutional neural network (CNN), feature fusion, hyperspectral image classification, transformer.**

Received 23 May 2025; revised 15 July 2025; accepted 26 July 2025. Date of publication 30 July 2025; date of current version 15 August 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62201201, in part by the Natural Science Foundation of Henan under Grant 252300421232 and Grant 242102210020, and in part by the Double First-Class Construction Project of Henan Polytechnic University under Grant GCCYJKT202515 and Grant GCCYJKT202536. *(Corresponding author: Jianlong Wang.)*

Junding Sun and Hongyuan Zhang are with the School of Computer Science and Technology, Henan Polytechnic University, Jiaozuo 454003, China (e-mail: 212209010015@home.hpu.edu.cn; sunjd@hpu.edu.cn).

Jianlong Wang is with the School of Computer Science and Technology, Henan Polytechnic University, Jiaozuo 454003, China, and also with School of Surveying and Land Information Engineering, Henan Polytechnic University, Jiaozuo 454003, China (e-mail: wangjianlong24@hpu.edu.cn).

Haifeng Sima is with the School of Software, Henan Polytechnic University, Jiaozuo 454003, China (e-mail: smhf@hpu.edu.cn).

Shuanggen Jin is with the School of Surveying and Land Information Engineering, Henan Polytechnic University, Jiaozuo 454003, China, and also with Shanghai Astronomical Observatory, Chinese Academy of Sciences, Shanghai 200030, China (e-mail: sgjin@shao.ac.cn).

Digital Object Identifier 10.1109/JSTARS.2025.3593885

I. Introduction

UNLIKE ordinary color images and multispectral images, hyperspectral images provide richer spectral information by capturing a large number of narrow and continuous spectral bands. The information enables a more precise analysis of the composition and characteristics of surface features. Therefore, hyperspectral imaging has extensive applications in fields such as food safety, mineral exploration, precision agriculture, land cover mapping, and environmental conservation. Hyperspectral image classification techniques utilize rich spectral and spatial information to assign unique categories to different types of surface features, generating a classification map that reflects their distribution.

In the early stages, traditional hyperspectral image classification methods comprised two phases: Feature processing and classification. Firstly, researchers employ feature extraction or dimensionality reduction techniques to process hyperspectral data, such as Principal Component Analysis (PCA) [1] and Local Binary Patterns [2]. Subsequently, methods such as k-Nearest Neighbors [3], Support Vector Machines [4], [5], and Random Forests [6], [7] are used to generate classification maps. However, with the increase in hyperspectral image bands and the expansion of application scenarios, the feature extraction and data fitting capabilities of traditional methods struggle to achieve satisfactory results when handling complex data.

In recent years, the application of deep learning technologies has significantly advanced the performance of hyperspectral image classification [8], [9], [10], [11]. Convolutional neural networks (CNNs), which capture local features and spatial information of hyperspectral data through convolutional operations, have become a widely used approach in hyperspectral image classification tasks [12], [13], [14], [15]. Hu et al. [16] utilized a 1-D CNNs to learn spectral features from the spectral domain for classification. Cheng et al. [17] utilized convolutional kernels of varying sizes to capture spatial information across different spectral bands. Next, Pan et al. [18] employed multiscale 3-D convolutions to extract important spectral and spatial features and enhanced the connectivity between the features through a special fusion strategy. On this basis, Ghaderizadeh et al. [19]

constructed a 3-D–2-D hybrid CNN. This method utilizes depthwise separable convolutional blocks and fast convolutional blocks to extract spectral-spatial features at a lower computational cost, while employing 2-D CNN to learn additional spatial features. CNNs utilize multiscale feature extraction to capture finer spectral and spatial information [20], [21], [22]. However, the lack of global feature learning makes it challenging to handle complex types of land cover [23], [24], [25], [26].

Transformer-based model enhances understanding of spatial continuity and spectral complexity through self attention mechanisms, thereby improving its ability to recognize features in complex scenes [27], [28], [29]. Hong et al. [30] proposed SpectralFormer, which uses Transformers to capture long-range spectral dependencies and incorporates adaptive residual connections to minimize the loss of critical features. Mei et al. [31] employed a hierarchical approach to construct a Transformer that extracts discriminative spatial and spectral features, while reducing computational cost by decreasing the number of channels. Shi et al. [32] combined the strengths of Transformers and multiscale features by using token inputs of different scales in two branches for feature extraction. The strength of the Transformer in handling long-range sequences enables it to learn key features from complex scenes. However, Transformer based methods require larger training datasets and extended training time. In addition, the lack of focus on local detail limits the full utilization of hyperspectral data [33], [34], [35].

Combining CNN and Transformer architectures allows for the simultaneous utilization of local detail and global context information, providing a more comprehensive feature representation [36], [37]. Sun et al. [38] developed the spectral-spatial feature tokenization transformer, which sequentially employs CNN and Transformer to efficiently learn low-mid-depth semantic features of hyperspectral data. Hu et al. [39] proposed the multiscale and multiangle attention network, which employs two convolutional mappings to extract spatial-spectral features. Subsequently, a multiangle attention module and a window attention module are utilized for feature learning and representation. The sequential cascading of CNN and Transformer limits effective collaboration between the two architectures. Yu et al. [40] utilize CNN and ViT to extract local and nonlocal features, respectively, with interaction modules introduced to enable mutual compensation between local and global features. To fully leverage the feature information across different hierarchical levels. Yang et al. [41] proposed an interactive transformer and CNN with a multilevel feature fusion network. The framework employs four parallel layers of Transformer and CNN to interactively extract features across different perceptual domains and depths, using fused features to achieve classification results. Nevertheless, these methodologies exhibit limitations in discerning subtle variations among ground objects within complex hyperspectral scenarios.

Based on the aforementioned analysis, this article proposes a novel re-attention agent transformer and multiscale partial convolution (RAT-MPC) for hyperspectral image classification. The methodology integrates multiscale convolution, Transformer mechanisms, and multilayer feature fusion modules to capture spatial-spectral information across diverse depths and perceptual fields. Initially, a multiscale spectral-spatial feature learning module is engineered to enhance and amalgamate rich spectral-spatial characteristics, thereby elevating the representational capacity of hyperspectral data. Subsequently, inspired by re-attention [42] and agent attention [43], an agent Transformer module incorporating re-attention mechanisms is developed to capture nonlocal features. Moreover, multiscale partial convolution (MPConv) is utilized to efficiently extract local spatial features while minimizing computational redundancy. The features obtained from both branches are summed to effectively synthesize information from diverse perceptual fields, thereby enhancing feature expressiveness. Next, a multilayer feature fusion attention module consolidates the more representative information obtained across different layers. Finally, an enhanced coordinate attention mechanism augments local spatial detail features, elevating the model's capacity for fine-grained feature recognition.

The main innovations of this article are summarized as follows.

1) A Multiscale Spectral-Spatial Feature Learning (MSSFL) module is constructed, employing a split-refactoring-fusion strategy to acquire spectral-spatial fusion features of ground objects across multiple scales while simultaneously reducing computational costs.
2) A Re-attention Agent Transformer (RAT) module is engineered, incorporating agent matrices within the Transformer architecture to achieve an optimal balance between computational efficiency and learning capacity. Moreover, transformation matrices are employed to enrich Transformer attention layer features, preventing the issue of attention maps becoming homogeneous as Transformer layers deepen.
3) A Multiscale Partial Convolution (MPConv) operation is incorporated into the proposed dual-branch local-global feature processing module. This operation performs multiscale processing on local spatial information, enabling the acquisition of fine-grained spatial features while minimizing redundant information utilization.
4) The Multilevel Feature Fusion Attention (MFFA) module enhances overall information expressiveness by amalgamating low level information from shallow layers with high-level semantic information from deeper layers. Furthermore, spatial attention mechanisms are applied to the fused information to accentuate fine-grained spatial representations.

The rest of this article is organized as follows. Section II first describes the proposed RAT-MPC for hyperspectral image classification and then provides a detailed description of modules of MSSFL, RAT, MPConv, and MFFA. Section III introduces the three datasets and evaluation indicators used in this paper, as well as the hyperparameter settings. It also presents the experimental results of the proposed RAT-MPC and compares with the current eight methods in view of the quantitative and qualitative analyses. A comprehensive discussion of the proposed method are given in Section IV, including ablation experiments, influence of patch size, selection of training sample amount and high dimensional feature visualization. Finally, Section V concludes

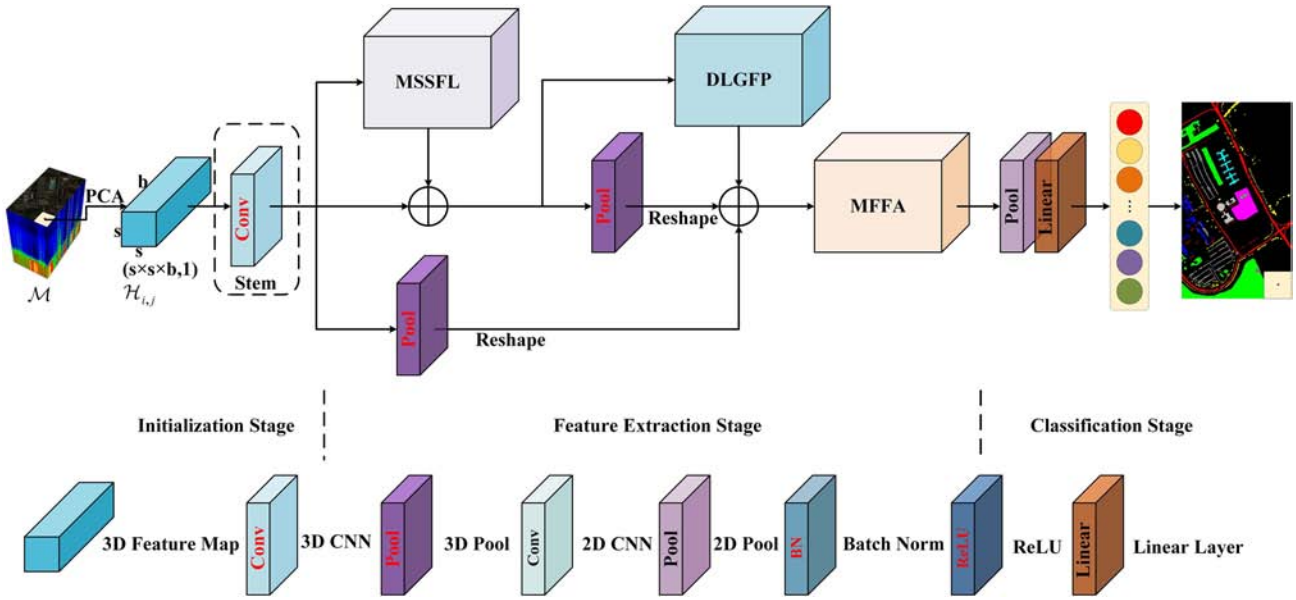

Fig. 1. Whole framework of the proposed RAT-MPC. The original hyperspectral data can be represented as $\mathcal{M} \in \mathbb{R}^{H\times W\times B}$, where *H*, *W*, and *B* denote the height, width, and number of bands of the hyperspectral image, respectively. $\mathcal{M}$ consists of labeled pixels and unlabeled pixels. Each pixel $\mathcal{M}_{i,j} \in \mathbb{R}^{1\times 1\times B}$ in $\mathcal{M}_{\{e\}}$ corresponds to a class $\mathcal{B} = \{\mathcal{B}_1, \ldots, \mathcal{B}_0\}$, $o = 1, \ldots, O$. Here, *O* represents the number of land cover classes in the dataset, and $i = 1, \ldots, H$ and $j = 1, \ldots, W$ denote the position of pixel $\mathcal{M}_{i,j}$ in the hyperspectral image $\mathcal{M}$. $\mathcal{M}_{\{e\}}$ represents the set of *e* labeled pixels in $\mathcal{M}$. In the initialization phase, the PCA algorithm is used to process hyperspectral data, reducing the number of bands from *B* to *b*. The reduces the spectral dimensionality and minimizes the impact of redundant information. The hyperspectral data after PCA dimension reduction are expressed as $\mathcal{M}_{pca} \in \mathbb{R}^{H\times W\times b}$, where *b* is the number of spectral bands after PCA. To effectively utilize the spatial context and spectral features of hyperspectral data, the dimensionally reduced hyperspectral image $\mathcal{M}_{pca}$ is segmented into small 3-D cubical patches $\mathcal{H}$, centered on labeled pixels. Each cubical patch $\mathcal{H}_{i,j} \in \mathbb{R}^{s\times s\times b}$ in $\mathcal{H}$ encompasses spectral and spatial information within a spatial window of $s \times s$. The class of the cubical patch $\mathcal{H}_{i,j}$ depends on the class of its central pixel $\mathcal{M}_{i,j}$. Subsequently, a $3 \times 3 \times 7$ 3-D convolutional layer is used for the initial extraction of spectral and spatial information and to increase the number of channels in the output feature map. This results in an output feature $\mathrm{X} \in \mathbb{R}^{c\times s\times s\times b}$. Next, MSSFL, DLGFP, and MFFA are used to extract multiscale spectral-spatial information, global spectral information, local spatial information, and multilevel feature information, respectively. Finally, global average pooling and a fully connected layer are employed to obtain the classification results.

this article with some remarks and hints at plausible future research lines.

II. Methodology

A. Whole Frame of Proposed RAT-MPC

An overview of the proposed RAT-MPC is illustrated in Fig. 1. The constructed framework divides hyperspectral images into fixed sized patches, which then undergo three main stages to extract spectral-spatial information for classification. The three main stages include: Initialization, feature extraction, and classification. In the initialization phase, Principal Component Analysis (PCA) is used to reduce the number of bands in the hyperspectral image, and a 3-D convolution is applied to adjust the number of channels. The feature extraction stage includes three parts: Multiscale spectral-spatial information extraction, local-global feature learning, and cross level feature fusion. The MSSFL module is used to extract shallow spectral-spatial information from the initialized feature map. The dual branch local-global feature processing module (DLGFP) module is employed to learn global spectral and local spatial features from the shallow features. The MFFA module is used to process information from different depths and perceptual fields. In the classification stage, the obtained features are passed through a global average pooling layer and a fully connected layer to generate the classification result. The following content will provide a detailed explanation of the specific design of each module within the RAT-MPC method.

B. Multiscale Spectral-Spatial Feature Learning Module

As shown in Fig. 1, the output feature $\mathrm{X} \in \mathbb{R}^{c\times s\times s\times b}$ from the initialization stage serves as the input to the MSSFL module. To comprehensively capture the rich spectral and spatial features of hyperspectral images and enhance adaptability to complex scenes, A multiscale spectral-spatial feature learning module, as shown in Fig. 2, is introduced. This module consists of three parts: ***Split***, ***Refactoring***, and ***Fusion***.

Split: As shown in the splitting part of Fig. 2, a given feature cube *X* is split along the channel dimension into two parts, with channel counts of αc and $(1-\alpha)c$, respectively. Here, α serves as the split ratio, with a value range of $0 \leq \alpha \leq 1$. The initial value of α is set to 0.5, meaning *X* is evenly divided into two parts along the channel dimension. After the splitting operation, *X* is divided into the upper part X_{up} and the lower part X_{low}.

Refactoring: X_{up}, serving as the spectral-spatial feature enricher, is fed back into the upper reconstruction stage. A multiscale 3-D CNN operation is applied to X_{up} to extract and enhance spectral-spatial information. The principle is that land-cover features of varying sizes and structures exhibit different characteristics at different scales, and the variations across spectral bands display diversity. Therefore, multiscale

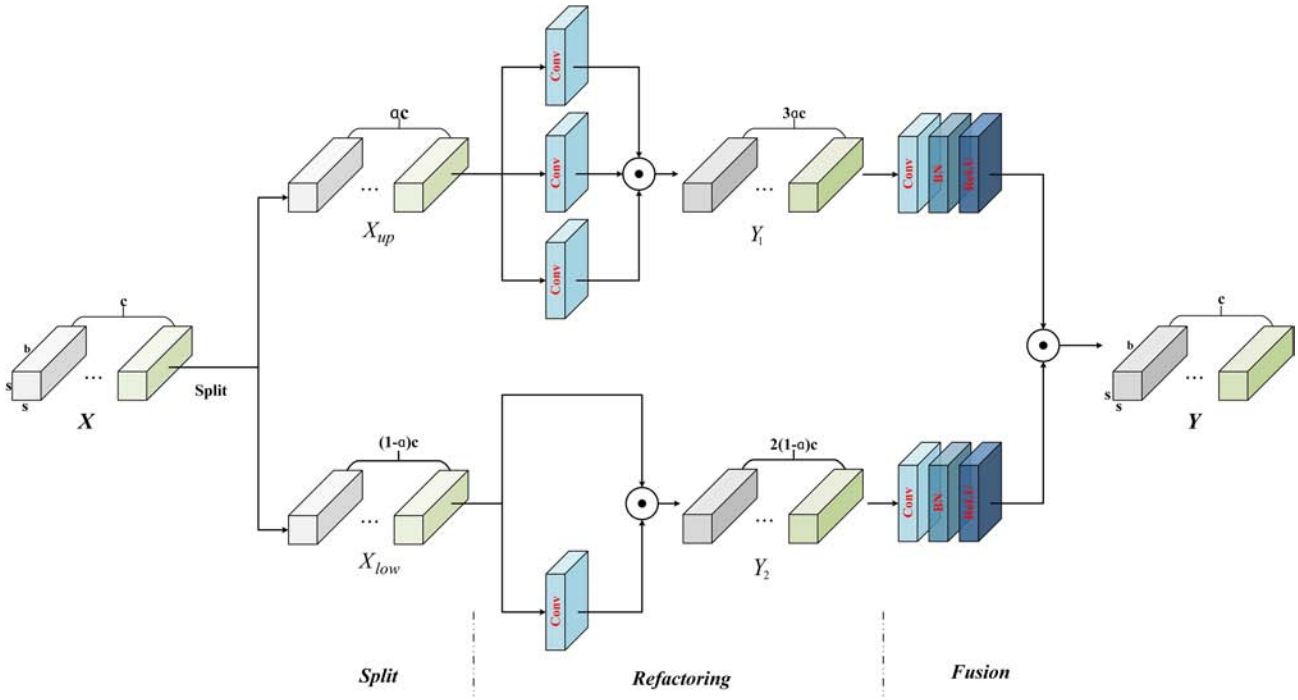

Fig. 2. Structure of the MSSFL module. MSSFL extracts multiscale spectral-spatial features through three stages: Split,Refactoring, and Fusion. Split: The feature map is divided along the channel dimension into two parts—one with αc channels and the other with αc and $(1-\alpha)c$ channels. Refactoring: The upper reconstruction stage is used to enrich spectral-spatial features, while the lower reconstruction stage serves as a complement to the upper stage. Fuse: Features from both reconstruction stages are processed via convolution and then concatenated along the channel dimension.

features are more beneficial in enhancing the ability of the subsequent module to perceive complex spectral and spatial structures. Specifically, convolution operations with different kernel sizes are applied to the same X_{up} to capture spectral and spatial information at different scales. The resulting multiscale information is then aggregated along the channel dimension, forming an information-rich feature map Y_1. As shown in the refactoring part of Fig. 2, the upper reconstruction stage can be expressed as

$$Y_1 = \text{Conv3D}_{3\times3\times3,1}(X_{up}) \odot \text{Conv3D}_{3\times3\times3,2}(X_{up}) \odot \text{Conv3D}_{3\times3\times3,3}(X_{up}) \quad (1)$$

where $\text{Conv3D}_{3\times3\times3.1}(\cdot)$, $\text{Conv3D}_{3\times3\times3,2}(\cdot)$, and $\text{Conv3D}_{3\times3\times3,3}(\cdot)$ represent 3-D convolution operations with a kernel size of $3\times3\times3$ and dilation rates of 1, 2, and 3, respectively. The symbol $\odot$ represents the concatenation operation. $X_{up} \in \mathbb{R}^{\alpha c\times s\times s\times b}$ and $Y_1 \in \mathbb{R}^{3\alpha c\times s\times s\times b}$ represent the input and output feature maps of the upper reconstruction stage, respectively. Overall, the upper reconstruction stage uses the fusion of different sizes convolution kernels on the same feature map X_{up} capturing rich spectral and spatial information in Y_1 with minimal computational cost.

X_{low} is used as the input for the lower reconstruction stage. In this stage, a 3-D convolution layer with a $1\times1\times1$ kernel is utilized to capture shallow hidden detail information as a supplement to the upper reconstruction stage. In addition, the captured detail information is fused with X_{low} to form the output Y_2 of the lower reconstruction stage, enriching the feature representation. The lower reconstruction stage can be expressed as

$$Y_2 = \text{Conv3D}_{1\times1\times1}(X_{low}) \odot X_{low} \quad (2)$$

where $\text{Conv3D}_{1\times1\times1}(\cdot)$ represents a 3-D convolution operation with a kernel size of $1\times1\times1$. $X_{low} \in \mathbb{R}^{(1-\alpha)c\times s\times s\times b}$ and $Y_2 \in \mathbb{R}^{2(1-\alpha)c\times s\times s\times b}$ denote the input and output feature maps of the lower reconstruction stage, respectively. In simple terms, the lower reconstruction stage reuses the original information X_{low} and applies a $1\times1\times1$ convolution to obtain detailed spectral-spatial feature representations as supplementary information.

Fusion: A $1\times1\times1$ convolution is applied to the output features Y_1 and Y_2 to reduce the number of channels and enhance information flow between channels. Subsequently, the dimension-reduced information is concatenated along the channel dimension, resulting in a more comprehensive feature representation *Y*. The fusion stage is as follows:

$$Y = \text{ReLU}(\text{BN}(\text{Conv3D}_{1\times1\times1}(Y_1))) \odot \text{ReLU}(\text{BN}(\text{Conv3D}_{1\times1\times1}(Y_2))) \quad (3)$$

where $\text{BN}(\cdot)$ represents batch normalization, and $\text{RELU}(\cdot)$ denotes the activation function. $Y \in \mathbb{R}^{\alpha c\times1\times1\times1}$ represents the output feature map obtained from the MSSFL module.

The MSSFL model employs depthwise dilated convolutions with varying receptive field sizes to capture multiscale spectral-spatial features and utilizes $1\times1\times1$ convolutions to learn hidden detail information as a supplement. At a lower computational cost, this approach enriches the representation of information, improving the classification performance of the model.

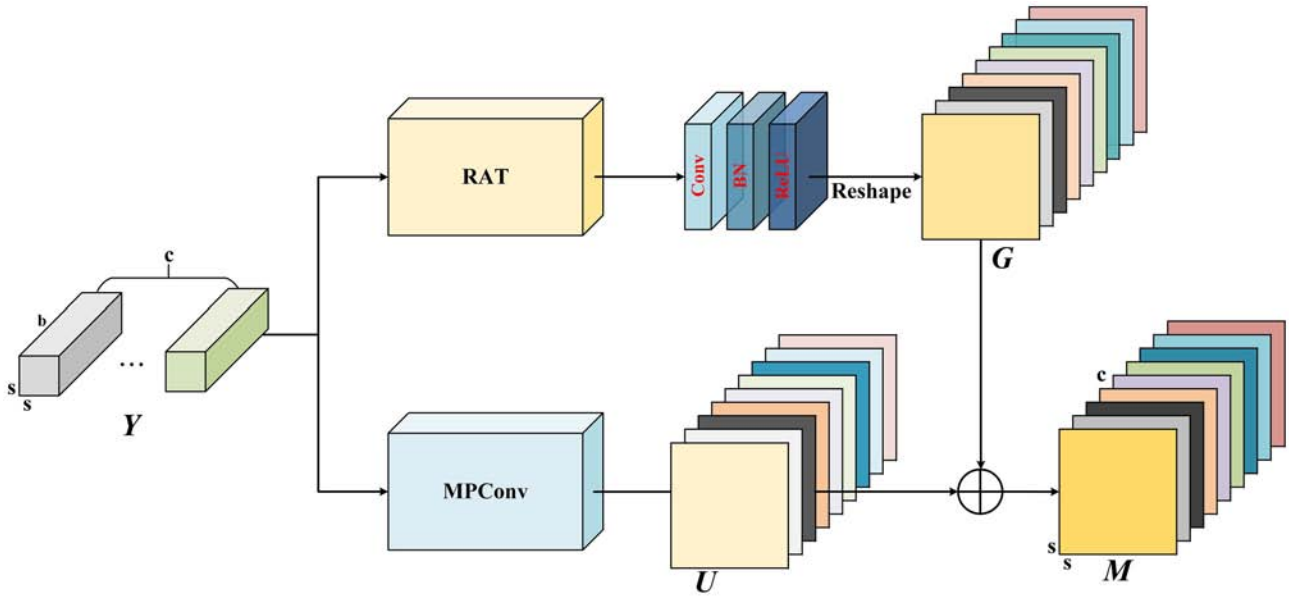

Fig. 3. Structure of the DLGFP module. DLGFP consists of two components: the RAT branch and the MPConv branch. The RAT branch is used to capture global spectral dependencies, while the MPConv branch is employed to extract local spatial information.

C. Dual Branch Local-Global Feature Processing Module

To effectively learn data diversity from shallow features, this article proposes the DLGFP module. This module learns global spectral and local spatial information through two separate branches to obtain more discriminative feature representations for classification tasks. Fig. 3 illustrates the structure of the proposed DLGFP model. The first branch, known as the (RAT branch, captures complex relationships between different bands. The second branch, called the multiscale partial convolution branch, utilizes MPConv to further extract abstract local spatial information. Subsequently, with the help of 3-D convolution, the feature dimensions extracted by the RAT branch are transformed to match the size of those from the MPConv branch. Finally, element-wise addition is used to fuse the global spectral information and local spatial information.

1) Re-Attention Agent Transformer Branch: CNN can effectively extract local spatial features such as textures and edges, making better use of the spatial correlation between adjacent pixels in hyperspectral images. However, it struggles to capture long-range dependencies and global relationships between different bands. In contrast, the Transformer can capture feature correlations at any position within the input data, making it particularly suitable for modeling long-range dependencies between different bands in hyperspectral images. However, the computational complexity of its self attention mechanism grows quadratically with the increase in data size, resulting in high computational costs when processing high-dimensional data like hyperspectral images. The method of using a proxy matrix in the Transformer reduces the number of query tokens to balance the computational efficiency and learning capacity of the Transformer. In addition, the re-attention method enhances the diversity of hyperspectral image features to prevent attention maps from becoming overly similar as Transformer layers deepen. As shown in Fig. 4, the RAT primarily consists of convolutional layers, an Agent Re-Attention (ARA) mechanism, and normalization layers.

As observed in Fig. 4, a $3 \times 3 \times 3$ 3-D convolution is used to integrate channel information. The resulting feature G' provides a more distinctive feature representation for subsequent classification tasks. Subsequently, ARA applies a $3 \times 3 \times 1$ convolution to the integrated features to generate the query matrix Q, key matrix K, and value matrix V. This operation maintains consistency along the spectral dimension while incorporating surrounding spatial information. Fig. 5 illustrates the workflow of ARA. A pooling operation is used to aggregate the neighboring band information of Q, generating a proxy matrix A to reduce the computational cost of the model. The proxy matrix A is treated as the query matrix to perform attention calculations with the key matrix K and value matrix V, effectively learning long-range dependencies between different bands and yielding the global spectral feature V_A. Subsequently, the original query matrix Q is used to perform a second attention calculation, with the proxy matrix A as the key matrix and V_A as the value matrix. This operation broadcasts the global spectral information of V_A to each band in the query matrix, enhancing the capture of detailed information. In addition, a learnable transformation matrix θ is used to dynamically aggregate multihead attention maps into a new attention map, enhancing the diversity of the attention representations. Finally, a $3 \times 3 \times 1$ single channel 3-D convolution layer is used to enhance the feature representation capability. The calculation process of ARA can be expressed as

$$Q, K, V = \text{Reshape}(\text{Split}(\text{Conv3D}_{3\times3\times1}(G'))) \tag{4}$$

$$A = \text{AvgPool3D}(Q) \tag{5}$$

$$V_A = \theta_1^T \left(\text{Softmax}\left(\frac{AK^T}{\sqrt{d}} \right) \right) V \tag{6}$$

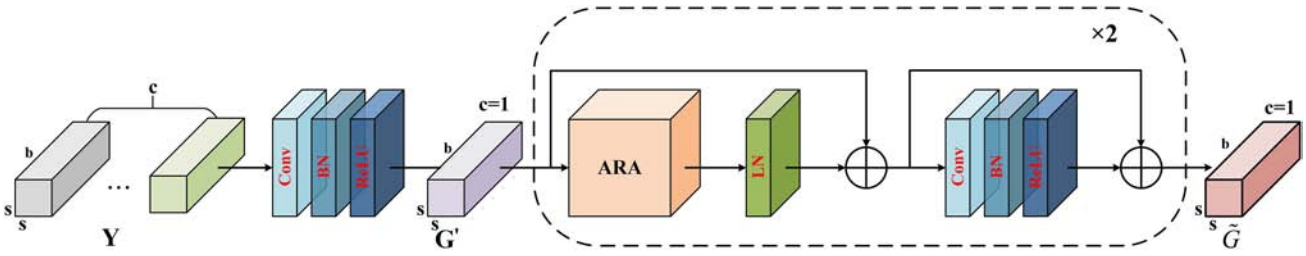

Fig. 4. Structure of the RAT module. RAT replaces the self-attention mechanism in the Transformer with ARA to capture global spectral features.

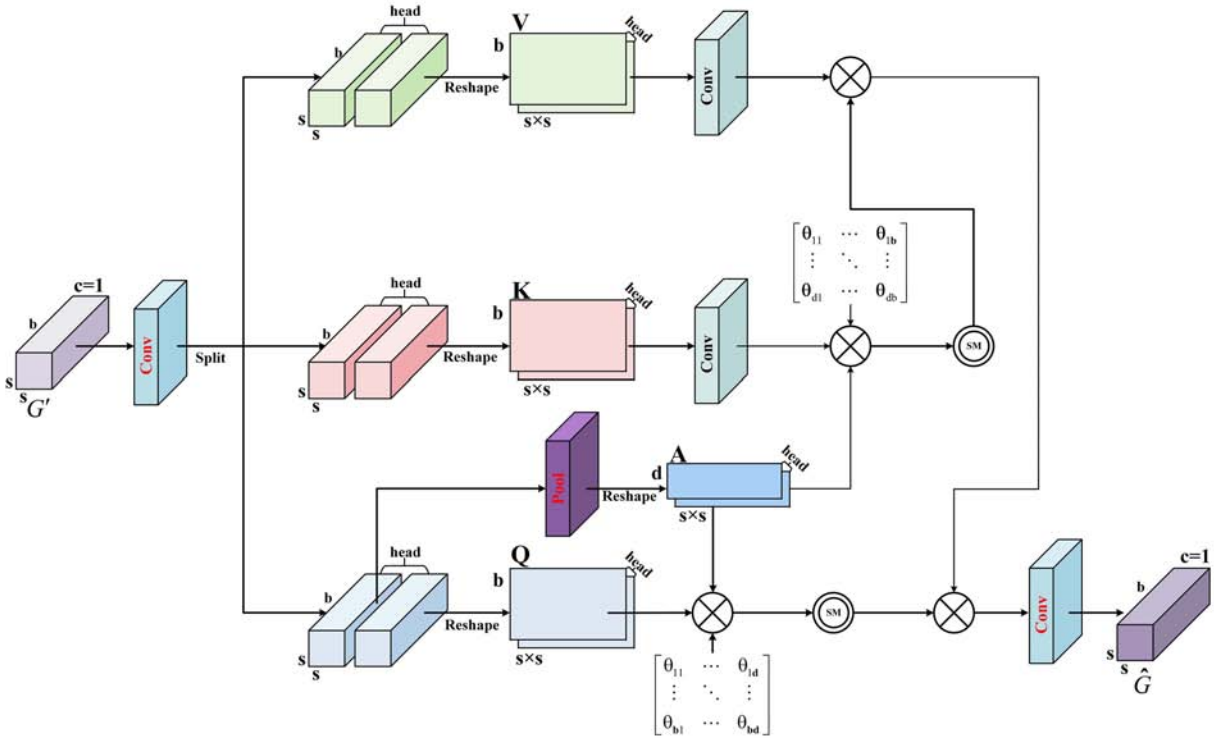

Fig. 5. Structure of the ARA module. ARA employs a 3D-CNN to generate the query matrix Q, key matrix K, and value matrix V, and uses a pooling operation to aggregate neighboring spectral band information from Q to obtain the proxy matrix A. Subsequently, the proxy matrix A is treated as the query matrix, and attention is computed with the key matrix K and value matrix V to obtain the global spectral feature V_A. Next, the original matrix Q is used as the query, the proxy matrix A as the key, and V_A as the value for a second attention computation, generating global spectral dependencies. Finally, the transformation matrix θ is used to aggregate the attention maps from different heads.

$$\hat{G} = \text{Conv3D}_{3\times3\times1}\left(\theta_2^T\left(\text{Softmax}\left(\frac{QA^T}{\sqrt{d}}\right)\right)V_A\right) \tag{7}$$

where $\theta_1^{\mathrm{T}} \in \mathbb{R}^{d\times b}$ and $\theta_2^{\mathrm{T}} \in \mathbb{R}^{b\times d}$ represent the learnable transformation matrices used in the two attention operations, respectively. Reshape$(\cdot)$ denotes the reshaping operation on the feature map, and Split$(\cdot)$ refers to the splitting operation, which divides the feature map along the channel dimension. Softmax$(\cdot)$ represents the Softmax activation function. $\hat{G} \in \mathbb{R}^{1\times s\times s\times b}$ denotes the output feature map obtained from the RAT module.

2) Multiscale Partial Convolution Branch: Based on PConv [44], the MPConv branch is designed. It uses 2-D CNNs with different receptive field sizes to supplement local spatial features, while maintaining low computational cost. Compared to composite neighborhood-aware convolution [45] and content-guided convolution [46], PConv applies standard convolution only to a subset of the input channels for spatial feature extraction, leaving the remaining channels unchanged, thereby improving the model's computational efficiency. Fig. 6 illustrates the workflow of the proposed MPConv.

Before performing the 2-D convolution operation, a $1 \times 1 \times b$ 3-D convolution is applied to compress the spectral dimension of the MSSFL output Y. The resulting feature map is then reshaped to obtain a format suitable for 2-D CNN operations. Subsequently, it is split along the channel dimension into two parts: feature $U' \in \mathbb{R}^{c_p\times s\times s}$, containing c_p channels, and remaining information $\widehat{U} \in \mathbb{R}^{(c-c_p)\times s\times s}$. U' is used as a representative of the entire feature map to extract local spatial features, while the remaining feature maps remain unchanged. It is worth noting that multiscale feature extraction employs a strategy that combines depthwise and dilated convolutions, allowing for a more comprehensive extraction of spatial features at a lower computational cost. The specific operation involves using

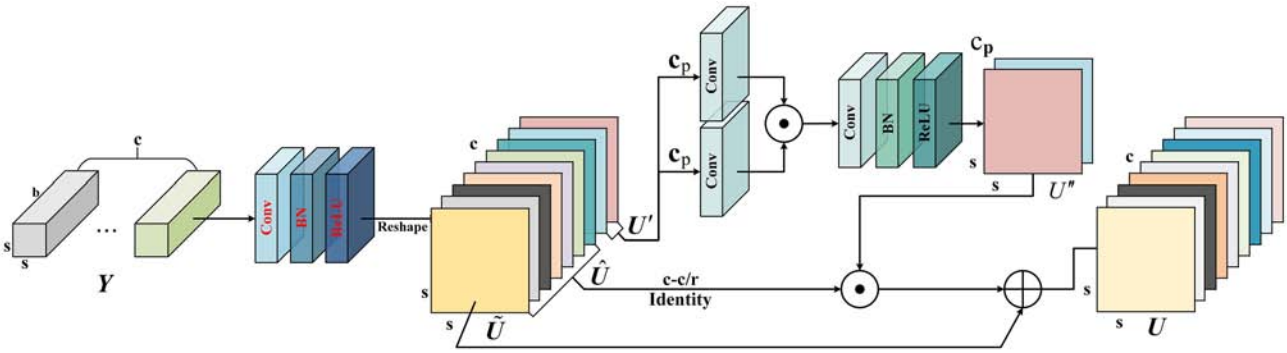

Fig. 6. Structure of the MPConv module. MPConv selects part of the original feature map content for multiscale feature extraction.

depthwise convolutions with a kernel size of 3×3 and dilation factors of 1 and 2 to obtain multiscale features $U'' \in \mathbb{R}^{c_p \times s \times s}$. Subsequently, a 2-D convolution layer with a kernel size of 1×1 is used to integrate information from different scales. Finally, the multiscale information U'' and the residual feature map $\hat{U}$ are fully fused using concatenation. In addition, residual connections are used to enable the reuse of shallower features, thereby enhancing the comprehensive spatial awareness of the branch. The information extraction process and calculation formulas of the MPConv module are shown as follows:

$$\tilde{U} = \text{Reshape}(\text{ReLU}(\text{BN}(\text{Conv3D}_{1\times1\times b}(Y)))) \tag{8}$$

$$U', \hat{U} = \text{Split}(\tilde{U}) \tag{9}$$

$$U'' = \text{ReLU}(\text{BN}(\text{Conv2D}_{1\times1}(\text{Conv2D}_{3\times3,1}(U') \odot \text{Conv2D}_{3\times3,2}(U')))) \tag{10}$$

$$U = U'' \odot \hat{U} \oplus \tilde{U} \tag{11}$$

where $\text{Conv3D}_{1\times1\times b}(\cdot)$ denotes a 3-D convolution operation with a kernel size of $1 \times 1 \times b$. $\text{Conv2D}_{3\times3,1}(\cdot)$ and $\text{Conv2D}_{3\times3,2}(\cdot)$ represent 2-D convolution operations with a kernel size of 3×3 and dilation rates of 1 and 2, respectively. $U \in \mathbb{R}^{c\times s\times s}$ denotes the input and output feature maps of the MPConv module. $\oplus$ represents the element-wise addition operation.

D. Multilevel Feature Fusion Attention Module

As shown in Fig. 7, the working mechanism of the MFFA module is illustrated. First, the fused information is projected along the vertical and horizontal directions to obtain features $\tilde{Z}$ and $\hat{Z}$. The projection operation is accomplished using adaptive global average pooling. Subsequently, the feature set is divided into three equally sized, independent subfeatures, and each subfeature is processed using convolution kernels of different sizes. Then, the different subfeatures are aggregated and normalized using group normalization. Finally, a spatial attention map is generated using the Sigmoid function.

In the first part, the different levels of information *X*, *Y*, and *M*, obtained after processing through the initialization, MSSFL module, and GLDFP module, are fused. The fused feature map $Z \in \mathbb{R}^{c\times s\times s}$ is obtained. The fusion process is represented as follows:

$$Z = \text{AvgPool}(X) \oplus \text{AvgPool}(X \oplus Y) \oplus M \tag{12}$$

where $\text{AvgPool}(\cdot)$ denotes adaptive global average pooling, used to compress spectral dimension information.

The second part introduces an improved spatial attention mechanism to further process the fused features, enhancing the extraction of fine-grained spatial features. First, global average pooling is applied to the fused feature map *Z* along the vertical and horizontal dimensions, resulting in two directional feature maps: $\tilde{Z} \in \mathbb{R}^{c\times1\times s}$ and $\hat{Z} \in \mathbb{R}^{c\times s\times1}$.

To learn different spatial distributions and enhance the extraction of fine-grained features, $\tilde{Z}$ and $\hat{Z}$ are each divided into three equally sized, independent subfeatures. The decomposition process of the sub-features is as follows:

$$\tilde{Z}_n = Z\left[:, (n-1) \times \frac{C}{3} : n \times \frac{C}{3}, :, :\right] \tag{13}$$

$$\hat{Z}_n = Z\left[:, (n-1) \times \frac{C}{3} : n \times \frac{C}{3}, :, :\right] \tag{14}$$

where $\tilde{Z}_n$ and $\hat{Z}_n$ represent the *n*th subfeature, $n = 1, 2, 3$. The grouped subfeatures correspond to different regions in the image, enhancing sensitivity to variations in local areas.

Subsequently, 2-D convolutions with kernel sizes of 1×1, 1×3, and 1×5 are sequentially applied to the three subfeatures of $\tilde{Z}$. Similarly, 2-D convolutions with kernel sizes of 1×1, 3×1, and 5×1 are applied sequentially to the three subfeatures of $\hat{Z}$. The approach enables the learning of diverse spatial contextual relationships to enrich feature representation. Furthermore, the different subfeatures are concatenated to obtain the feature maps $\tilde{Z}'$ and $\hat{Z}'$

$$\tilde{Z}' = \text{Conv2D}_{1\times1}(\tilde{Z}_1) \odot \text{Conv2D}_{1\times3}(\tilde{Z}_2) \odot \text{Conv2D}_{1\times5}(\tilde{Z}_3) \tag{15}$$

$$\hat{Z}' = \text{Conv2D}_{1\times1}(\hat{Z}_1) \odot \text{Conv2D}_{3\times1}(\hat{Z}_2) \odot \text{Conv2D}_{5\times1}(\hat{Z}_3) \tag{16}$$

where $\text{Conv2D}_{1\times3}(\cdot)$ represents a 2-D convolution operation with a kernel size of 1×3. $\tilde{Z}' \in \mathbb{R}^{c\times1\times s}$ and $\hat{Z}' \in \mathbb{R}^{c\times s\times1}$ denote the feature maps obtained after aggregating the different subfeatures.

Furthermore, the different subfeatures are concatenated and normalized using group normalization with three groups. Finally, the Sigmoid activation function is applied to convert the

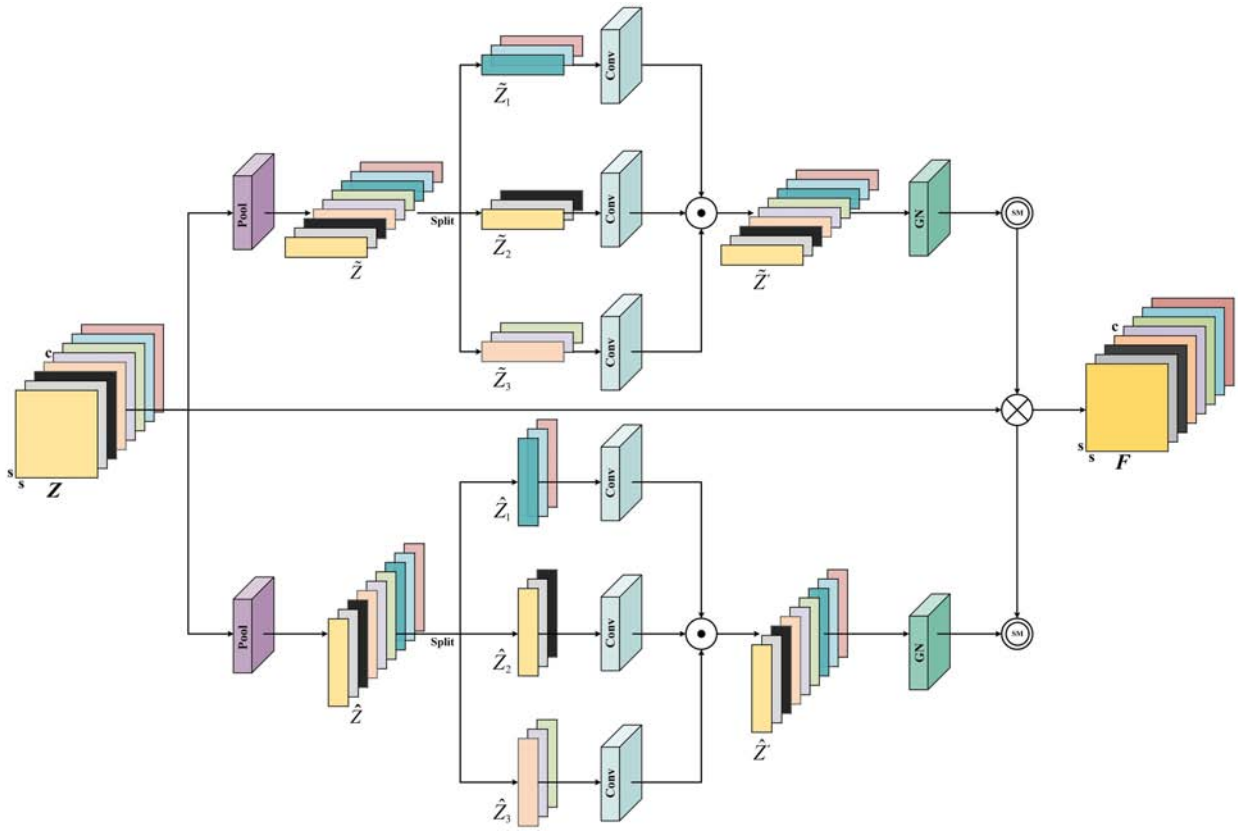

Fig. 7. Structure of the MFFA module. First, MFFA fuses feature maps with different depths and receptive fields using element-wise addition, followed by global average pooling in both horizontal and vertical directions. Subsequently, the feature map is divided into three equally sized sub-features, each processed employing convolution operations at different scales. Next, group normalization is applied to normalize the feature maps, and the Sigmoid function is used to generate directional feature weight maps. Finally, element-wise multiplication is applied to the weight maps to generate spatial attention feature maps, which are then applied to the original feature map.

resulting feature map into different weights, which are then multiplied to generate the spatial attention feature map. The spatial attention map is multiplied element-wise with the input feature map Z to highlight key spatial regions. The enhanced feature map is added element-wise to the original feature map to improve the expressiveness of the features

$$F = Z \otimes \text{Sigmoid}(\text{GN}(\tilde{Z}')) \otimes \text{Sigmoid}(\text{GN}(\hat{Z}')) \oplus Z \quad (17)$$

where $\text{GN}(\cdot)$ represents the group normalization operation, and $\text{Sigmoid}(\cdot)$ denotes the Sigmoid activation function. $\otimes$ represents the element-wise multiplication operation. $F \in \mathbb{R}^{c\times s\times s}$ denotes the feature information generated by the MFFA module.

The third part uses a global average pooling layer and a fully connected layer to perform learnable label recognition and weight learning on the obtained features, generating the final classification result.

III. Experiment Results and Analysis

In this section, the discourse commences with an introduction to three renowned hyperspectral datasets employed in the experimentation. Subsequently, the hyperparameter configurations and experimental environment utilized in the study are described. Finally, extensive experiments and analyses are conducted on three real hyperspectral datasets to evaluate the proposed RAT-MPC method, comparing its performance with other advanced hyperspectral image classification techniques. Next, ablation studies are first conducted to illustrate the impact of various modules within RAT-MPC on the model's classification performance.

A. Datasets Description

To validate the performance and effectiveness of the proposed RAT-MPC model, evaluations are conducted on three well known hyperspectral datasets: Indian Pines (IP), Pavia University (PU), and WHU-Hi-LongKou (LK). Among these datasets, the IP and PU datasets effectively validate the classification performance of the model in agricultural and urban road scenes. The LK dataset, with its high spatial resolution, enables the assessment of model classification performance under intricate scenarios.

IP: This dataset is captured by AVIRIS over the Indian Pines test site. It includes 224 spectral bands within the $0.4 \sim 2.5$ μm range. In the experiments, 24 bands affected by water absorption are removed, resulting in 200 bands available for analysis. The IP dataset consists of an image with a spatial size of 145×145 pixels, containing a total of 10 249 labeled samples divided into 16 land-cover classes. The false-color image, ground truth map, and color codes are presented in Fig. 8, respectively.

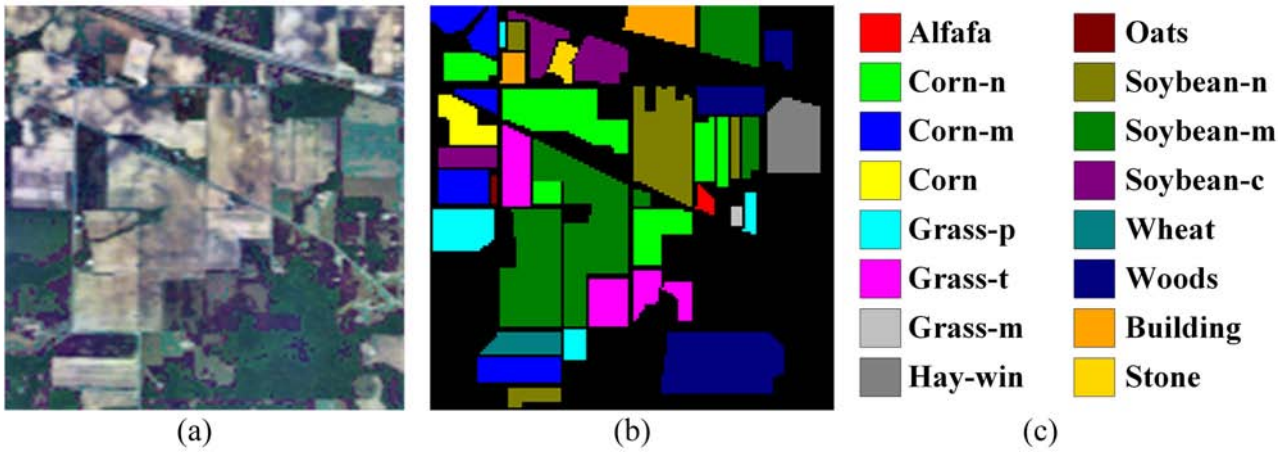

Fig. 8. Indian Pines Datasets. (a) False-color image, (b) Ground truth map, (c) Color codes.

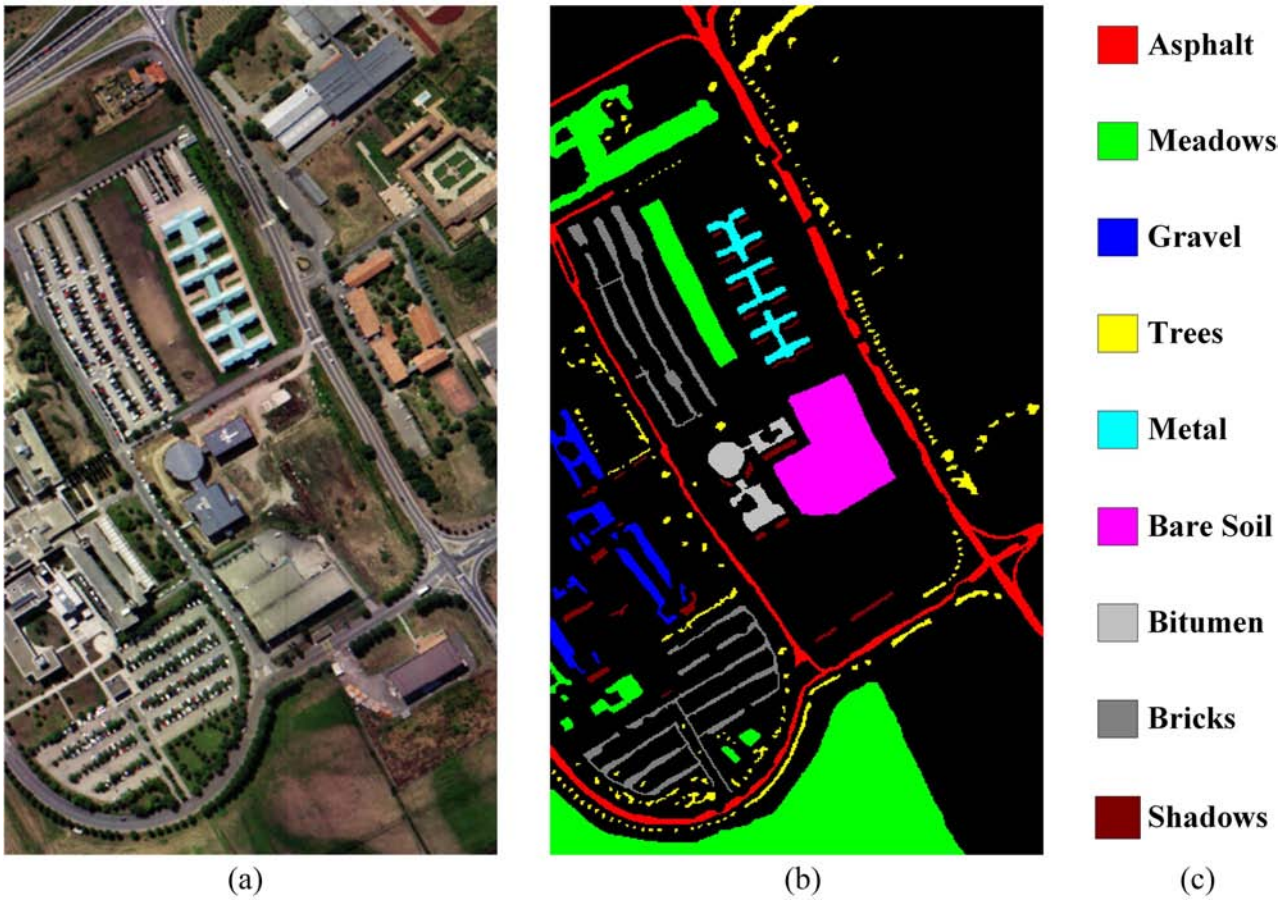

Fig. 9. Pavia University Datasets. (a) False-color image, (b) Ground truth map, (c) Color codes.

PU: This dataset is collected by the ROSIS sensor over Pavia University. It includes 115 spectral bands within the $0.43 \sim 0.86$ μm range. After removing 12 bands affected by noise and water absorption, 103 bands are retained for analysis. The PU dataset consists of an image with a spatial size of 610×340 pixels, containing a total of 42 776 labeled samples divided into nine land-cover classes. The false-color image, ground truth map, and color codes are shown in Fig. 9, respectively.

LK: This dataset is captured in Longkou, Hubei Province, using the Headwall Nano-Hyperspectral imaging sensor mounted on a DJI M600 Pro drone. It includes 270 spectral bands within the $400 \sim 1000$ nm range. The LK dataset consists of an image with a spatial size of 550×400 pixels and a high spatial resolution of 0.463 m. This dataset contains a total of 204 542 labeled samples, divided into 9 land cover classes. The false-color image, ground truth map, and color codes are displayed in Fig. 10, respectively.

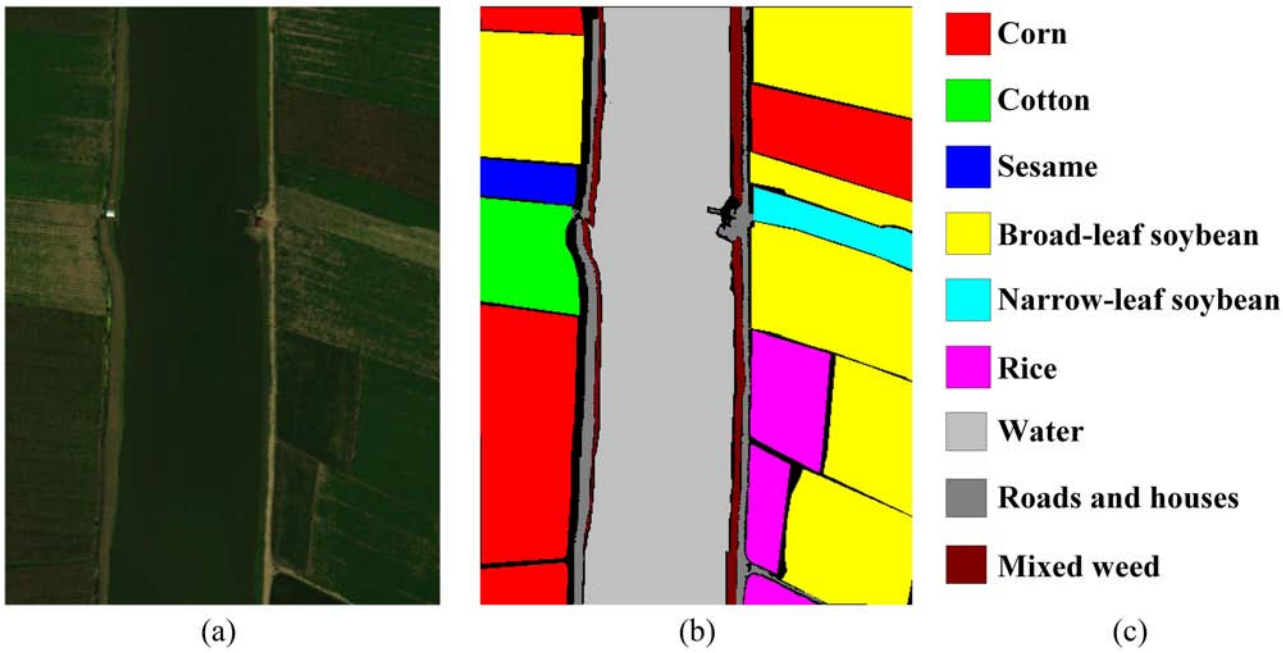

Fig. 10. WHU-Hi-LongKou Datasets. (a) False-color image, (b) Ground truth map, (c) Color codes.

TABLE I
SAMPLE AMOUNTS OF TRAINING, VALIDATION AND TEST SET FOR THREE DATASETS

No.	Indian Pines(IP)				Pavia University(PU)				WHU-Hi-LongKou(LK)			
	Class name	Train	Val	Test	Class name	Train	Val	Test	Class name	Train	Val	Test
1	Alfalfa	5	3	38	Asphalt	47	47	6537	Corn	35	35	34441
2	Corn-n	72	72	1284	Meadows	131	131	18387	Cotton	9	9	8356
3	Corn-m	42	42	746	Gravel	15	15	2069	Sesame	5	4	3022
4	Corn	12	12	213	Trees	22	22	3020	Broad-leaf-soybean	64	64	63084
5	Grass-p	25	25	433	Metal	10	10	1352	Narrow-leaf soybean	5	5	4141
6	Grass-t	37	37	656	Bare Soil	36	36	4957	Rice	12	12	11830
7	Grass-m	5	3	20	Bitumen	10	10	1310	Water	68	68	66920
8	Hay-win	24	24	430	Bricks	26	26	3630	Roads and house	8	8	7108
9	Oats	5	3	12	Shadows	7	7	933	Mixed weed	6	6	5217
10	Soybean-n	49	49	874								
11	Soybean-m	123	123	2209								
12	Soybean-c	30	30	533								
13	Wheat	11	11	183								
14	Woods	64	64	1137								
15	Building	20	20	346								
16	Stone	5	5	83								
Total		529	523	9197		304	304	42168		212	211	204119

To objectively evaluate the proposed network, the IP, PU, and LK datasets are divided into training, validation, and test sets. For the three datasets, 5%, 0.7%, and 0.1% of the labeled samples are randomly selected as the training set. The validation set follows the same sampling ratio as the training set, with the remaining samples designated as the test set. The hyperspectral datasets have an imbalance in the number of classes, and using a smaller sampling ratio results in some classes lacking selected samples, which impacts the performance of the model. A minimum sampling threshold is applied during the selection of training samples to ensure that each class contains at least a specified number of samples. In this study, the minimum sampling thresholds for the training and validation sets are set to 5 and 3, respectively. Table I presents the names of land cover classes, along with the number of training, validation, and test samples for the three datasets.

B. Experimental Settings

Evaluation Metrics: This article uses three quantitative evaluation metrics–Overall Accuracy (OA), Average Accuracy (AA), and the Kappa coefficient (Kappa)–to assess the performance of each method. OA represents the proportion of correctly classified samples among all samples, serving as a measure of the overall classification performance of the model. AA calculates the ratio of correctly predicted samples to the total number of

samples within each class. The Kappa coefficient measures the consistency between the predicted results and the actual ground truth. The values of the three quantitative evaluation metrics increase as the classification performance of the model improves. In addition, the number of parameters(Paras) and FLOPs are used to evaluate the computational overhead of the model.

Comparison Methods: To analyze the performance of the proposed RAT-MPC method, comparative experiments are conducted with methods based on CNN, methods that combine CNN with attention mechanisms, and methods that integrate CNN with Transformer architectures. The following content provides a detailed description of the comparative methods used in the experiments:

1) *SSRN:* SSRN employs a 3D CNN to construct two residual blocks–a spectral residual block and a spatial residual block–to sequentially capture more discriminative spectral and spatial features from hyperspectral images. This network segments the original hyperspectral image into patches as input, with each input patch sized $7 \times 7 \times \mathbf{b}$, where *b* represents the number of spectral bands in the input.
2) *CVSSN:* CVSSN integrates the input 1-D sequence information with 3-D cube data to effectively learn spectral and spatial features. Subsequently, Euclidean distance similarity measure and cosine angle similarity measure are used to calculate the self similarity representation oriented by the central spectral vector, enhancing the spatial information representation. In addition, this network introduces Euclidean distance similarity to measure the similarity between the central feature vector and its neighboring feature vectors, thoroughly exploring the spatial relationships between the central and adjacent feature vectors. The input patch sizes are $1 \times 1 \times \mathbf{b}$ and $9 \times 9 \times \mathbf{b}$.
3) *A^2S^2KResNet:* A^2S^2KResNet is designed with 3-D ResBlocks to jointly extract more robust and discriminative spectral-spatial features. This framework employs a spectral attention mechanism to capture long-range nonlinear cross channel correlations. The input patch size is $9 \times 9 \times \mathbf{b}$.
4) *DBMA:* DBMA establishes two parallel branches: The first branch employs dense 3-D CNN for spectral feature extraction, incorporating channel attention to enhance spectral representation. The second branch utilizes dense 3-D CNN for spatial feature extraction, integrating spatial attention mechanisms to amplify focus on the most information rich regions. The input patch size is $9 \times 9 \times \mathbf{b}$.
5) *DBDA:* DBDA proposes a dual branch dual attention network, which includes a spectral branch and a spatial branch to extract spectral features and spatial features, respectively. The self attention mechanism is applied separately to the spectral and spatial dimensions to optimize and refine feature maps. The input patch size is $9 \times 9 \times \mathbf{b}$.
6) *SSFTT:* The SSFTT method combines CNN and Transformer from shallow to deep layers. This network uses CNN to extract shallow spectral-spatial features. Subsequently, the shallow features are transformed by a Gaussian weighted feature tokenizer into tokenized semantic features, and the data is fed into the Transformer encoder to learn relationships among high level semantic features. The input patch size is $13 \times 13 \times 30$.
7) *BS2T:* The BS2T framework includes a spectral branch and a spatial branch, dedicated to extracting spectral information and spatial information, respectively. Each branch comprises three distinct phases. The first stage uses 3-D CNN to capture local information from hyperspectral images. The second stage employs Transformer to obtain long-range global dependencies from the local information. Finally, the features obtained from both branches are fused for classification. The input patch size is $9 \times 9 \times \mathbf{b}$.
8) *DBCTNet:* The DBCTNet network uses 3-D CNN in the early stages to enrich spectral features. Then, a dual branch module of 3-D CNN and Transformer is constructed to fully integrate local and global features. The input patch size is $9 \times 9 \times \mathbf{b}$.

Implementation Details: All experiments in this study are implemented in the PyTorch environment and trained on a machine equipped with an NVIDIA P100 16 GB GPU. To avoid the influence of initialization and ensure fairness, all experiments are independently repeated 20 times. The 5 results with the lowest and highest OA are discarded, and the mean and variance of the remaining 10 results are calculated. To better illustrate the model's optimal classification capability, the experimental results are visualized by selecting the model parameter configuration with the highest OA from 20 experiments. The cross entropy loss function is used to calculate the loss, and the Adam optimizer is employed for model optimization. The batch size is set to 16, the initial learning rate to 0.001, and the number of epochs to 200. To accelerate model convergence, cosine annealing scheduling is used to dynamically adjust the learning rate over 200 epochs. In addition, for the proposed RAT-MPC method, a patch size of 9×9 is configured for each of the three datasets, and the number of spectral bands after PCA dimensionality reduction is set to 30.

C. Ablation Experiment

In this section, to verify the effectiveness of each part of the RAT-MPC network for hyperspectral image classification, ablation experiments are conducted on four different components across the three datasets. In the experiments, 5%, 0.7%, and 0.1% of samples are selected from the IP, PU, and LK datasets for training, respectively. The spectral dimension is reduced to 30 using the PCA method. The experimental results are obtained by performing 20 iterations, with the 5 highest and 5 lowest values removed, leaving the mean of the remaining 10 results. The ConvTE branch from DBCTNet [47] is used as the baseline model for ablation experiments on the different modules within RAT-MPC. A detailed analysis is conducted on five different model combinations, and changes in the OA, AA, and Kappa evaluation metrics are observed to assess the impact of each component on RAT-MPC. The ablation experiment results for the five different model combinations are presented in Table II. In the table, the symbol "✓" indicates that the module is used, while "×" denotes that the module is not used.

RAT refers to the operation of adding an agent matrix to ConvTE and using a re-attention mechanism to exchange

TABLE II
RESULTS ABLATION STUDIES OF THE PROPOSED MODULE ON DIFFERENT DATASETS

Datasets	Models	Components				Metric		
		RAT	MPConv	MSSFL	MFFA	OA	AA	Kappa
IP	ConvTE	×	×	×	×	0.8968±0.0053	0.8495±0.0613	0.8823±0.0059
	RAT	✓	×	×	×	0.9212±0.0047	0.9290±0.0151	0.9101±0.0055
	DLGFP	✓	✓	×	×	0.9461±0.0064	0.9569±0.0061	0.9386±0.0073
	MSSFL-DLGFP	✓	✓	✓	×	0.9584±0.0028	0.9632±0.0090	0.9526±0.0031
	RAT-MPC	✓	✓	✓	✓	**0.9666±0.0025**	**0.9669±0.0086**	**0.9619±0.0029**
PU	ConvTE	×	×	×	×	0.9359±0.0096	0.8881±0.0201	0.9145±0.0129
	RAT	✓	×	×	×	0.9451±0.0031	0.9143±0.0081	0.9269±0.0041
	DLGFP	✓	✓	×	×	0.9648±0.0017	0.9444±0.0059	0.9532±0.0023
	MSSFL-DLGFP	✓	✓	✓	×	0.9770±0.0022	0.9632±0.0039	0.9694±0.0030
	RAT-MPC	✓	✓	✓	✓	**0.9820±0.0011**	**0.9696±0.0020**	**0.9762±0.0015**
LK	ConvTE	×	×	×	×	0.9538±0.0035	0.8119±0.0196	0.9390±0.0047
	RAT	✓	✓	×	×	0.9688±0.0024	0.8985±0.0161	0.9590±0.0032
	DLGFP	✓	✓	×	×	0.9756±0.0027	0.9220±0.0139	0.9679±0.0036
	MSSFL-DLGFP	✓	✓	✓	×	0.9806±0.0011	0.9381±0.0126	0.9745±0.0014
	RAT-MPC	✓	✓	✓	✓	**0.9844±0.0014**	**0.9472±0.0125**	**0.9794±0.0018**

The Bold One is the Optimal Result.

information across different attention heads. In Table II, it can be observed that, compared to the ConvTE method, RAT shows a significant improvement in classification performance across the three datasets. Specifically, the OA, AA, and Kappa indices increase by 2.44%, 7.95%, and 2.78% on the IP datasets; by 0.92%, 2.62%, and 1.24% on the PU dataset; and by 1.5%, 8.66%, and 2% on the LK dataset. The demonstrates that the designed RAT module can more effectively capture long-range dependencies among different spectral bands, resulting in improved classification performance.

DLGFP is a method that adds a parallel MPConv branch to the RAT branch. The data in Table II reveals that employing the RAT branch independently resulted in a notable decline in classification accuracy compared to the integrated utilization of RAT and MPConv. In the IP, PU, and LK datasets, OA decreases by 2.49%, 1.97%, and 0.68%, while Kappa decreases by 2.85%, 2.63%, and 0.89%, respectively. This is because RAT demonstrates strong capability in learning long-range dependencies among different spectral bands but struggles to effectively capture local features such as texture, shape, and edges of land cover elements, thereby reducing classification performance.

MSSFL-DLGFP represents the application of the proposed MSSFL module before DLGFP to preliminarily extract spectral-spatial features. The data presented in Table II demonstrates that DLGFP incorporating MSSFL achieved superior classification performance across all three datasets compared to DLGFP employed independently. Compared to the performance of DLGFP on the PU dataset, incorporating MSSFL increases OA, AA, and Kappa by 1.22%, 1.88%, and 1.62%, respectively. It confirms the effectiveness of the proposed MSSFL module. The MSSFL module enriches spectral-spatial information through multiscale deep convolutions and uses point convolutions to reconstruct hidden detail features as a supplement, thereby enhancing the representation capacity of shallow features.

RAT-MPC includes the MSSFL, DLGFP, and MFFA modules. In Table II, it can be observed that this method achieves superior classification performance across all three datasets compared to previous module combinations. In the MFFA module, the fusion of features from different perceptual fields and depths is enhanced with coordinate attention, improving fine-grained feature extraction and boosting the classification performance of the model. Compared to MSSFL-DLGFP, OA, AA, and Kappa increase by 0.82%, 0.37%, and 0.93% on the IP dataset; by 0.5%, 0.64%, and 0.68% on the PU dataset; and by 0.38%, 0.91%, and 0.49% on the LK dataset. Overall, each module in the proposed RAT-MPC method plays a crucial role in the feature extraction process.

D. Comparative Experiments

In this section, diverse models are applied to three hyperspectral datasets, with analyses conducted from qualitative, quantitative, and computational complexity perspectives to validate the classification efficacy of the proposed methodology. The classification results of different methods on the three datasets are presented in Tables III, IV, and V, while the visualizations of different models are shown in Figs. 11, 12, and 13. The highest classification accuracy values, as well as the lowest parameter amount and FLOPs, are highlighted in bold. To distinctly demonstrate the visualization results of different models, specific regions within the classification maps are demarcated with white rectangular frames and subsequently magnified.

1) Classification Maps and Categorized Results for the IP: The classification results of different algorithms on the IP dataset are shown in Table III. It can be observed that the pro-posed RAT-MPC achieves commendable classification results, with OA and Kappa reaching 96.66% and 96.19%, respectively. Compared to the CNN-based SSRN method, the approach achieves an improvement of 0.46% in OA and 0.49% in the Kappa coefficient. Meanwhile, the proposed RAT-MPC method has the lowest parameter count and FLOPs, with approximately 30 K and 13 M, respectively. Compared with A^2S^2KResNet and BS2T, RAT-MPC achieves a 0.87% and 0.68% improvement in OA. The RAT-MPC also achieves an 1% and 0.77% improvement in Kappa. Moreover, RAT-MPC achieved outstanding classification results with only one-twelfth of the parameter count

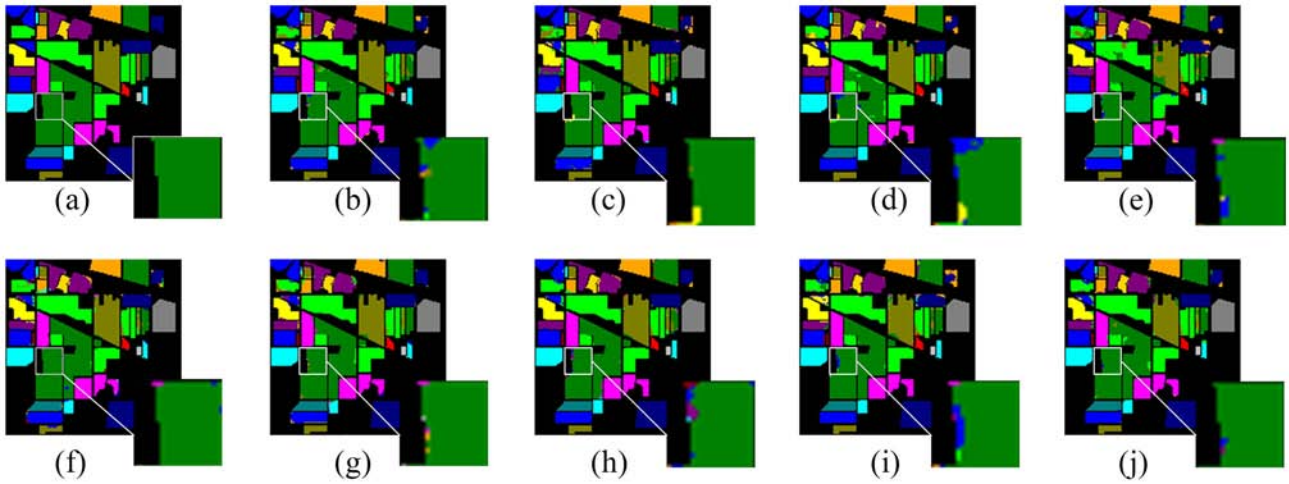

Fig. 11. Classification Result Images achieved by different methods for the IP dataset. (a) Ground Truth. (b) SSRN(OA:96.23%). (c)CVSSN (OA:94.61%). (d) A^2S^2KResNet(OA:95.79%). (e) DBMA(OA:95.04%). (f) DBDA(OA:94.76%). (g) SSFTT(OA:95.34%). (h) BS2T(OA:95.98%). (i) DBCTNet(OA:95.33%). (j) RAT-MPC(OA:96.69%).

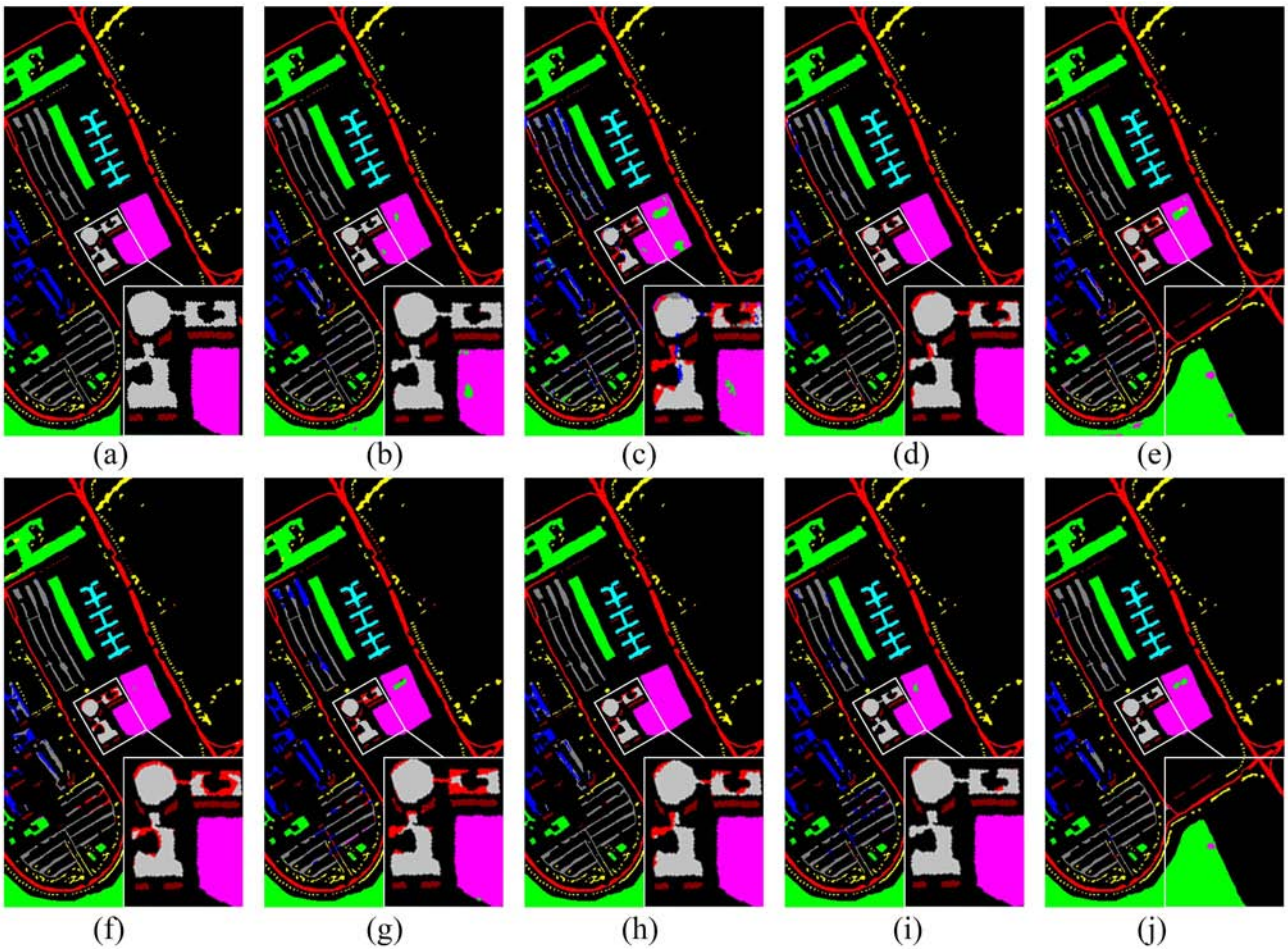

Fig. 12. Classification Result Images achieved by different methods for the PU dataset. (a) Ground Truth. (b) SSRN(OA:96.89%). (c)CVSSN(OA:92.85%). (d) A^2S^2KResNet(OA:97.26%). (e) DBMA(OA:95.43%). (f) DBDA(OA:95.99%). (g) SSFTT(OA:97.52%). (h) BS2T(OA:97.51%). (i) DBCTNet(OA:97.52%). (j) RAT-MPC(OA:98.20%).

TABLE III
CLASSIFICATION PERFORMANCE BY DIFFERENT METHODS FOR IP DATASET

Class	Classical CNN-based Method	Classical-based Method with Attention				Hybrid CNN-Transformer Method			Proposed
	SSRN	CVSSN	A^2S^2KResNet	DBMA	DBDA	SSFTT	B2ST	DBCTNet	RAT-MPC
1	0.7500±0.2923	0.8816±0.1443	0.9868±0.0224	0.9553±0.0497	0.8921±0.1070	0.9842±0.0184	0.9763±0.0455	**0.9921±0.0178**	0.9868±0.0256
2	0.9572±0.0241	0.9100±0.0384	**0.9612±0.0173**	0.9194±0.0349	0.9513±0.0223	0.9488±0.0219	0.9607±0.0339	0.9467±0.0226	0.9509±0.0203
3	0.9583±0.0317	0.9441±0.0248	0.9479±0.0334	0.9426±0.0389	0.9473±0.0342	0.9413±0.0241	0.9401±0.0467	0.9504±0.0345	**0.9598±0.0210**
4	0.9319±0.0699	0.9127±0.0568	0.9141±0.0595	0.9305±0.0505	0.9113±0.0544	0.8789±0.0827	**0.9624±0.0475**	0.9850±0.0167	0.9099±0.0590
5	**0.9543±0.0223**	0.9520±0.0182	0.9478±0.0358	0.9196±0.0298	0.8975±0.0415	0.9245±0.0234	0.9406±0.0321	0.9212±0.0505	0.9455±0.0340
6	0.9886±0.0140	0.9895±0.0124	0.9816±0.0149	0.9718±0.0140	0.9709±0.0164	0.9733±0.0124	0.9904±0.0092	**0.9919±0.0058**	0.9889±0.0070
7	0.9500±0.0100	0.9550±0.1117	**1.0000±0.0000**	**1.0000±0.0000**	0.9650±0.1107	0.9750±0.0635	0.9800±0.0632	**1.0000±0.0000**	**1.0000±0.0000**
8	0.9967±0.0095	0.9979±0.0035	0.9921±0.0147	0.9974±0.0034	0.9886±0.0164	0.9967±0.0072	**0.9995±0.0010**	0.9953±0.0087	0.9974±0.0060
9	0.9967±0.0703	0.9833±0.0527	**1.0000±0.0000**	0.9417±0.0562	0.9417±0.0686	0.9417±0.1043	**1.0000±0.0000**	0.9833±0.0527	0.9500±0.1315
10	0.9451±0.0344	0.9204±0.0245	0.9200±0.0321	0.9019±0.0449	0.9196±0.0345	0.9247±0.0365	0.9403±0.0126	0.9335±0.0455	**0.9481±0.0270**
11	0.9602±0.0171	0.9574±0.0228	0.9665±0.0159	0.9702±0.0138	0.9404±0.0300	0.9616±0.0250	0.9449±0.0392	0.9305±0.0325	**0.9721±0.0119**
12	0.9437±0.0373	0.8765±0.0445	0.9002±0.0441	0.9126±0.0489	0.9323±0.0441	0.9236±0.0502	0.9298±0.0609	0.9349±0.0283	**0.9548±0.0210**
13	**0.9973±0.0039**	0.9962±0.0055	0.9945±0.0103	0.9934±0.0136	0.9907±0.0082	0.9934±0.0141	0.9945±0.0136	0.9923±0.0146	0.9934±0.0106
14	0.9846±0.0142	0.9759±0.0114	0.9814±0.0146	0.9836±0.0111	0.9776±0.0184	0.9792±0.0147	**0.9909±0.0104**	0.9820±0.0244	**0.9909±0.0066**
15	0.9399±0.0472	0.9344±0.0519	0.9399±0.0455	0.9451±0.0392	0.9428±0.0583	0.9587±0.0275	**0.9743±0.0255**	0.9708±0.0263	0.9526±0.0195
16	0.9819±0.0307	0.9386±0.0503	0.9795±0.0327	0.9831±0.0190	0.9771±0.0237	0.8904±0.0981	0.9819±0.0351	**0.9867±0.0192**	0.9699±0.0307
OA	0.9623±0.0028	0.9461±0.0037	0.9579±0.0043	0.9504±0.0036	0.9476±0.0073	0.9534±0.0073	0.9598±0.0087	0.9533±0.0042	**0.9666±0.0025**
AA	0.9504±0.0193	0.9453±0.0124	0.9633±0.0046	0.9543±0.0069	0.9466±0.0139	0.9498±0.0084	**0.9692±0.0088**	0.9686±0.0065	0.9669±0.0086
Kappa	0.9570±0.0033	0.9385±0.0042	0.9519±0.0049	0.9433±0.0042	0.9403±0.0082	0.9469±0.0083	0.9542±0.0099	0.9468±0.0047	**0.9619±0.0029**
Paras	364.1680K	260.9020K	370.7600K	606.9060K	606.1010K	148.4880K	377.2360K	30.6240K	**30.5580K**
FLOPs	158.3820M	21.0748M	171.5971M	245.9659M	245.5944M	182.4404M	108.1469M	14.8305M	**12.9790M**

The Bold One is the Optimal Result.

TABLE IV
CLASSIFICATION PERFORMANCE BY DIFFERENT METHODS FOR PU DATASET

Class	Classical CNN-based Method	Classical-based Method with Attention				Hybrid CNN-Transformer Method			Proposed
	SSRN	CVSSN	A^2S^2KResNet	DBMA	DBDA	SSFTT	B2ST	DBCTNet	RAT-MPC
1	0.9710±0.0186	0.9263±0.0610	0.9700±0.0172	0.9727±0.0206	0.9644±0.0319	0.9766±0.0194	0.9802±0.0160	0.9740±0.0193	**0.9862±0.0066**
2	0.9909±0.0092	0.9938±0.0074	0.9970±0.0026	0.9877±0.0102	0.9956±0.0063	0.9989±0.0021	0.9979±0.0044	0.9895±0.0079	**0.9990±0.0013**
3	0.7780±0.1124	0.7710±0.1309	0.8336±0.0937	0.7367±0.1333	0.7420±0.0877	0.8707±0.0392	0.8008±0.0855	**0.8934±0.0713**	0.8869±0.0456
4	0.9352±0.0239	0.9438±0.0291	**0.9761±0.0098**	0.9183±0.0563	0.9167±0.0299	0.9403±0.0317	0.9378±0.0191	0.9491±0.0188	0.9587±0.0140
5	0.9972±0.0070	0.9688±0.0365	0.9950±0.0109	**0.9981±0.0025**	0.9937±0.0089	0.9973±0.0066	0.9955±0.0080	0.9922±0.0128	0.9933±0.0102
6	0.9762±0.0203	0.8881±0.0380	0.9890±0.0059	0.9685±0.0538	0.9777±0.0253	0.9606±0.0386	**0.9994±0.0012**	0.9974±0.0046	0.9838±0.0154
7	0.9569±0.0541	0.7479±0.1317	0.9087±0.0488	0.8343±0.1392	0.8389±0.0970	**0.9909±0.0175**	0.8810±0.1269	0.9753±0.0348	0.9880±0.0124
8	0.9666±0.0482	0.7711±0.1695	0.9167±0.0671	0.9171±0.0550	0.9388±0.0488	0.9526±0.0399	**0.9777±0.0238**	0.9361±0.0556	0.9550±0.0278
9	**0.9992±0.0013**	0.9786±0.0309	0.9956±0.0061	0.9411±0.0661	0.9569±0.0307	0.9534±0.0246	0.9595±0.0407	0.9791±0.0451	0.9759±0.0178
OA	0.9689±0.0051	0.9285±0.0067	0.9726±0.0037	0.9543±0.0073	0.9599±0.0058	0.9752±0.0037	0.9751±0.0041	0.9752±0.0029	**0.9820±0.0011**
AA	0.9524±0.0131	0.8877±0.0152	0.9535±0.0086	0.9194±0.0203	0.9250±0.0104	0.9601±0.0043	0.9478±0.0145	0.9651±0.0087	**0.9696±0.0020**
Kappa	0.9587±0.0067	0.9045±0.0091	0.9637±0.0049	0.9393±0.0097	0.9467±0.0078	0.9670±0.0050	0.9669±0.0054	0.9672±0.0039	**0.9762±0.0015**
Paras	216.5370K	247.2590K	220.7280K	321.4910K	320.6860K	148.0330K	201.2610K	**16.6810K**	30.4390K
FLOPs	81.2128M	20.0260M	87.8925M	146.8706M	146.5088M	182.4330M	55.4228M	**7.5209M**	12.9789M

The Bold One is the Optimal Result.

of A^2S^2KResNet and BS2T. A^2S^2KResNet is the best model for classification on the joint CNN and attention mechanism-based methods, while BS2T stands as the top-performing framework among the hybrid CNN and Transformer-based approaches.

SSFTT integrates 3-D–2-D convolutional layers and Transformer modules, effectively mitigates classification errors caused by "same spectral foreign objects" and "same object different spectral." However, the continuous extraction of spectral and spatial information poses challenges in effectively distinguishing and utilizing different features. As a result, when dealing with land cover categories with limited samples, the classification performance fails to achieve satisfactory results. The classification accuracy for category 4 (corn) and category 16 (stone) is 87.89% and 89.04%, respectively, showing a decrease of 3.1% and 7.95% compared to the proposed RAT-MPC method. This is because the RAT-MPC method incorporates the DLGFP module, which effectively integrates Transformer and MPConv. The architecture enables the simultaneous extraction of both local and global information, ensuring the comprehensive utilization of spectral and spatial features.

In addition, the IP dataset includes categories with highly similar spectral characteristics and spatial structures, such as the three subclasses of "soybean." These categories significantly

TABLE V
CLASSIFICATION PERFORMANCE BY DIFFERENT METHODS FOR LK DATASET

Class	Classical CNN-based Method	Classical-based Method with Attention				Hybrid CNN-Transformer Method			Proposed
	SSRN	CVSSN	A^2S^2KResNet	DBMA	DBDA	SSFTT	B2ST	DBCTNet	RAT-MPC
1	0.9979±0.0012	0.9954±0.0045	0.9910±0.0129	0.9847±0.0186	0.9924±0.0068	0.9973±0.0023	**0.9989±0.0013**	0.9962±0.0027	0.9985±0.0010
2	0.8956±0.1034	0.8917±0.0925	0.9116±0.0586	0.8389±0.1157	0.7858±0.1432	**0.9460±0.0660**	0.9316±0.0498	0.9170±0.0683	0.9434±0.0487
3	0.8963±0.0802	0.8702±0.0723	0.8522±0.0728	0.9281±0.0386	0.8245±0.2824	0.9450±0.0474	**0.9455±0.0448**	0.9391±0.0661	0.9252±0.0360
4	0.9693±0.0237	0.9748±0.0127	0.9794±0.0140	0.9764±0.0133	0.9831±0.0082	0.9816±0.0099	**0.9915±0.0052**	0.9570±0.0144	0.9899±0.0072
5	0.5568±0.2700	0.8909±0.0546	0.7963±0.1549	0.6196±0.1823	0.7283±0.1539	0.8742±0.0822	0.8864±0.0939	**0.9521±0.0354**	0.9014±0.0543
6	0.9892±0.0154	0.9681±0.0185	0.9805±0.0108	0.9596±0.0311	0.9729±0.0294	0.9597±0.0415	0.9916±0.0063	0.9859±0.0158	**0.9944±0.0051**
7	0.9993±0.0010	0.9962±0.0038	0.9976±0.0025	0.9971±0.0042	**0.9988±0.0014**	0.9959±0.0057	0.9976±0.0024	0.9941±0.0058	0.9992±0.0012
8	0.8383±0.0849	0.7657±0.0921	0.8507±0.0856	0.7727±0.1487	0.8447±0.1222	0.8483±0.0817	0.9127±0.0500	0.8840±0.0873	**0.9508±0.0302**
9	0.7434±0.1708	0.7867±0.0594	0.8352±0.0941	0.7267±0.1097	0.6965±0.0691	0.8091±0.0924	0.8203±0.0522	**0.8748±0.0664**	0.8223±0.0860
OA	0.9623±0.0082	0.9662±0.0030	0.9708±0.0054	0.9566±0.0051	0.9615±0.0076	0.9744±0.0034	0.9824±0.0022	0.9708±0.0040	**0.9844±0.0014**
AA	0.8762±0.0382	0.9044±0.0120	0.9105±0.0283	0.8671±0.0231	0.8697±0.0298	0.9286±0.0158	0.9418±0.0147	0.9445±0.0130	**0.9472±0.0125**
Kappa	0.9505±0.0108	0.9555±0.0039	0.9616±0.0072	0.9429±0.0067	0.9492±0.0100	0.9664±0.0045	0.9768±0.0029	0.9618±0.0052	**0.9794±0.0018**
Paras	471.5130K	269.2290K	479.8200K	809.3390K	808.5340K	148.0330K	504.0690K	40.5850K	**30.4390K**
FLOPs	214.6509M	21.8330M	232.6313M	334.2129M	333.8414M	182.4333M	146.5886M	20.1594M	**12.9789M**

The Bold One is the Optimal Result.

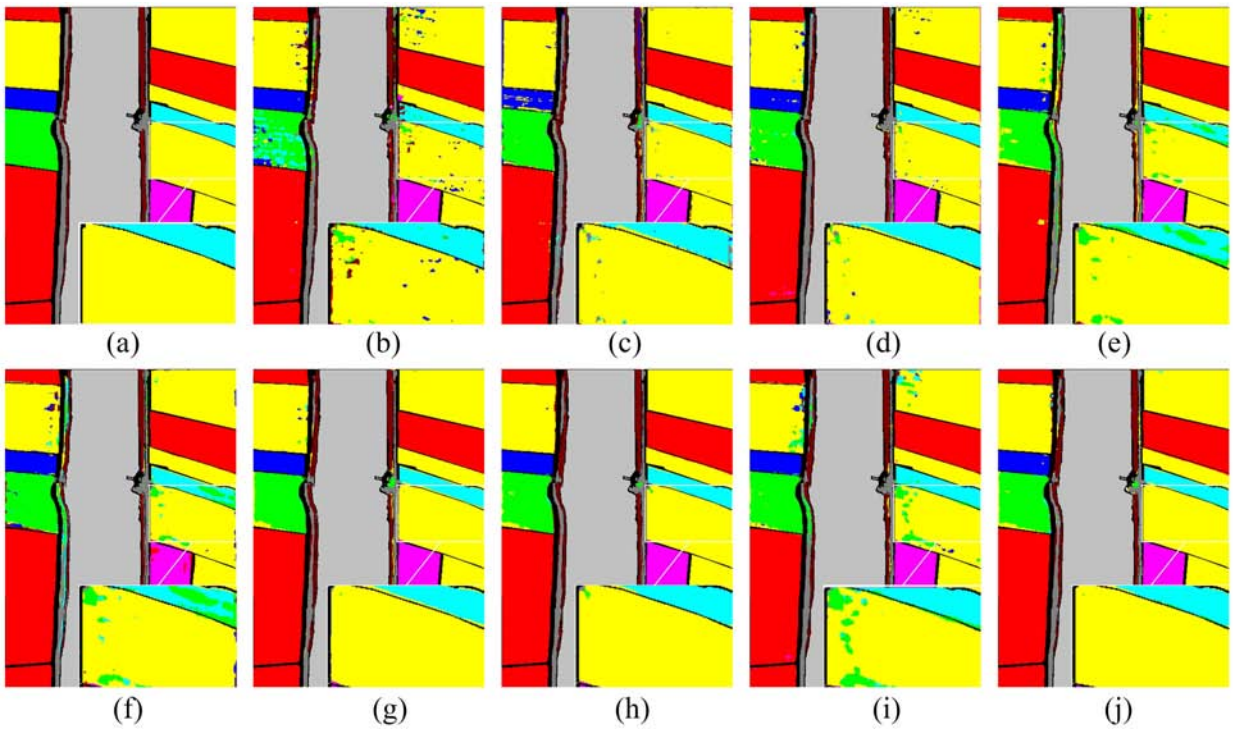

Fig. 13. Classification Result Images achieved by different methods for the LK dataset. (a) Ground Truth. (b) SSRN(OA:96.23%). (c) CVSSN(OA:96.62%). (d) A^2S^2KResNet(OA:97.08%). (e) DBMA(OA:95.66%). (f) DBDA(OA:96.15%). (g) SSFTT(OA:97.44%). (h) BS2T(OA:98.24%). (i) DBCTNet(OA:97.08%). (j) RAT-MPC(OA:98.44%).

impact the classification performance of the model. As shown in Table III, the comparative methods exhibit relatively poor classification performance in distinguishing the three subclasses of "soybean." The proposed RAT-MPC method incorporates the RAT module, which models the global dependencies and complex nonlinear relationships between different bands in hyperspectral images. The enables the model to focus on the frequency bands that contribute more significantly to the classification task, thereby facilitating the distinction between these subclasses. As a result, the classification outcomes for the three subclasses are exceptional, with accuracies of 94.81%, 97.21%, and 95.48%, respectively. Compared to other methods, the proposed approach achieves the largest accuracy improvements for the three subclasses of "soybean" by 4.62%, 4.16%, and 7.83%, respectively. The minimum enhancements achieved are 0.3%, 0.19%, and 1.11%, respectively.

The visualization results of different methods for the IP dataset are shown in Fig. 11. Fig. 11(a) shows the ground truth

of the IP dataset. Fig. 11(b) shows the prediction results of the SSRN method. It can be observed that the Alfalfa category, represented by red regions, contains numerous Dark Gray errors. This is because CNN struggles to effectively learn the spectral and spatial features of categories with a smaller number of pixels during the classification process. Fig. 11(d) presents the classification results of A^2S^2KResNet. The central green region, representing the Soybean-m category, exhibits more pronounced classification errors, with noticeable noise along the edges. Corn-m is distributed across five distinct regions, each varying in shape and number of pixels, which increases the difficulty for the model to achieve accurate classification. Fig. 11(g) displays the classification result obtained using the SSFTT method. In the dark blue triangular region on the left, representing Corn-m, various heterogeneous pixels appear, including categories such as Soybean-c, Corn, and Corn-n. The indicates that the sequential use of CNN and Transformer methods struggles to effectively integrate local and global information, making the classification of Corn-m susceptible to interference from heterogeneous pixels. Fig. 11(h) presents the classification result obtained using the BS2T method. BS2T effectively utilizes spectral and spatial information through its dual-branch design, avoiding the occurrence of multiple heterogeneous pixels. However, it is influenced by the similar category Corn-n, leading to significant misclassification of Corn-m as Corn-n in the mixed regions of the two categories. Fig. 11(i) shows the classification map obtained using the DBCTNet method. Due to the significant influence of the similar category Corn, nearly all the pixels in this region are misclassified as Corn. Fig. 11(j) illustrates the classification result obtained using the proposed RAT-MPC method. Compared to other methods, RAT-MPC achieves the highest prediction accuracy for this category. Although some noise is present in the visualization map, it is relatively minimal. From the magnified regions, it can be seen that there is a lot of noise in the edge area in Fig. 11(c), (e), and (f) resulting in blurred category boundaries. In contrast, the proposed RAT-MPC method demonstrates superior classification results by integrating features from different depths and perceptual domains, effectively reducing noise in edge regions. By combining the objective evaluation metrics in III with the prediction results in Fig. 11 and comparing them with other methods, it can be concluded that the proposed method achieves the best performance on the IP dataset.

2) Classification Maps and Categorized Results for the PU: The classification results of different algorithms on the PU dataset are shown in Table IV. The proposed RAT-MPC demonstrated outstanding classification performance, achieving OA, AA, and Kappa values of 98.20%, 96.96%, and 97.62%, respectively. RAT-MPC outperforms other models, achieving the highest improvements of 5.35%, 8.19%, and 7.17% in OA, AA, and Kappa coefficients, respectively, while maintaining the smallest enhancements of 0.68%, 0.45%, and 0.9%. Compared to the hybrid CNN and Transformer networks, SSFTT and BS2T, the proposed method achieves improvements of 0.68% and 0.69% in OA, 0.95% and 2.18% in AA, and 0.92% and 0.93% in the Kappa coefficient, respectively. This is because RAT-MPC integrates the MSSFL module, which employs a split-refactoring-fusion strategy to optimize the representation of hyperspectral data. In addition, the RAT-MPC method achieves standard deviations of just 0.11% for OA, 0.2% for AA, and 0.15% for the Kappa coefficient, all of which are lower than those of the compared methods. The demonstrates that the proposed method exhibits high stability and robustness. This is attributed to the use of MFFA for integrating features from different stages and the enhanced fine-grained feature representation achieved through the improved coordinate attention mechanism. Moreover, the RAT-MPC method has significantly fewer parameter count and FLOPs compared to these models,about one-eleventh of DBMA, giving it a strong advantage in resource-constrained environments.

DBCTNet stands as the model with the highest OA, AA, and Kappa values among the fusion methods based on CNN and Transformer. It combines CNN and Transformer architectures to effectively utilize local and global features, thereby improving classification performance. In addition, DBCTNet employs pseudo-3D convolution and downsampling operations during the feature extraction process, significantly reducing the parameter count and FLOPs. Among the comparative methods, it ranks second only to the proposed method. Its OA, AA, and Kappa values reached 97.52%, 96.51%, and 96.72%, respectively, representing decreases of 0.68%, 0.95%, and 0.92% compared to the proposed RAT-MPC method. The model even achieved the highest classification accuracy for the third class (Gravel). The RAT-MPC method emphasizes the effective utilization of global and local multiscale features, achieving superior performance in urban areas. Although the proposed method does not achieve the best classification performance in the “Gravel” category, trailing the DBCTNet method by 0.65%. This is because it introduces the focal loss function, which effectively alleviates the impact of category imbalance during model training and enhances the model’s attention to difficult-to-classify samples such as minority classes. The standard deviation for RAT-MPC in this category is 4.56%, which is 2.57% lower than DBCTNet’s 7.13%. The fully demonstrates that the proposed method is more robust in the classification performance for this category.

The visualization results of different methods for the PU dataset are shown in Fig. 12. Fig. 12(a) shows the ground truth of the PU dataset. It can be observed from Fig. 12(b) that the SSRN method exhibits classification errors in the yellow region in the lower-left corner, which represents the Tree category. In the PU dataset, Tree is often distributed in dotted or patchy patterns around other buildings, containing a significant number of edge pixels. In addition, it is susceptible to interference from other categories, leading to mispredictions at category boundaries. Using only CNN for classification makes it challenging to fully capture the subtle differences between different spectral bands. In Fig. 12(c), CVSSN demonstrates relatively good classification performance in certain category regions. However, its classification results for many other categories are less satisfactory. In Fig. 12(d), A^2S^2KResNet exhibits a significant amount of noise in the red region representing Asphalt. Fig. 12(e) shows the classification results of DBMA. It can be observed that a large number of classification errors occur in the lower-left corner, along with mispredictions in the internal pixels of the light green region at the bottom. In Fig. 12(f), the DBDA method struggles to

distinguish between Gravel and Bricks, resulting in large-scale misclassification. Fig. 12(g) presents the classification results of SSFTT. SSFTT is highly susceptible to interference from surrounding pixels when identifying Bricks. It misclassifies bricks into multiple categories, such as Asphalt, Gravel, and Bare soil. Fig. 12(h) shows the classification results of BS2T, a method combining CNN and Transformer. It accurately predicted most categories, but exhibited classification errors in the Bitumen category. Fig. 12(i) illustrates the classification results of the DBCTNet method. The lower-left corner contains a significant amount of noise and a high proportion of misclassified edge pixels. The annotated regions in the image contain three classes: Bitumen, Bare Soil, and Shadows. However, many comparative methods misclassify some Bitumen and Shadows as Asphalt and Bare Soil as Meadows. In particular, CVSSN misclassifies some Bitumen as Gravel, resulting in considerable noise within the region. In comparison, the proposed RAT-MPC method also exhibits a small amount of noise but maintains clear boundaries.

3) Classification Maps and Categorized Results for the LK: Unlike the other two datasets, the LK dataset features centimeter level spatial resolution, which is more conducive to extracting spatial features. The classification results of different models on the LK dataset are shown in Table V. The classification results of the proposed method surpass those of the comparative methods, achieving OA, AA, and Kappa values of 98.44%, 94.72%, and 97.94%, respectively. The CNN-based SSRN method achieves OA, AA, and Kappa values of only 96.23%, 87.62%, and 95.05% on LK dataset, with classification performance lagging behind the proposed method by 2.21%, 7.1%, and 2.89%, respectively. In addition, compared to the A^2S^2 KResNet model, which achieves the best classification performance among CNN and attention-based fusion methods. The proposed method outperformed it by 1.36%, 3.67%, and 1.78% in OA, AA and Kappa values, respectively. BS2T is the best-performing model among the fusion methods based on CNN and Transformer. Its OA, AA, and Kappa values are very close to those of the proposed RAT-MPC method, reaching 98.24%, 94.18%, and 97.68%, with differences of only 0.2%, 0.54%, and 0.26%, respectively. This is because BS2T employs a parallel branch structure that more comprehensively captures local and global spectral-spatial information, achieving commendable classification performance. However, its parameter count is approximately 504 K, while RAT-MPC achieves outstanding classification results with only 30 K parameter count. In addition, RAT-MPC has the lowest FLOPs, approximately 13 M.

In the LK dataset, Class 8 (Roads and Houses) is often found surrounding the crops. It is influenced by the surrounding land cover during the classification process, making it difficult to achieve satisfactory classification performance. RAT-MPC demonstrated optimal effectiveness in classifying Roads and Houses, achieving an accuracy of 95.08%. Compared to other methods, it outperformed by a maximum of 18.51% and a minimum of 3.87%. Compared to SSRN, A^2S^2KResNet, and BS2T, the classification performance improved by 11.25%, 10.01%, and 3.87%, respectively. This is because it can fully capture subtle spectral differences between different land cover types, efficiently filter, and integrate feature information across different levels.

The visualization results of different methods for the LK datasets are shown in Fig. 13. Fig. 13(a) shows the ground truth of the PU dataset. Fig. 13(b) presents the classification results of SSRN. It can be observed that this model struggles to distinguish between Cotton and Narrow-leaf Soybean, resulting in poor classification performance for Narrow-leaf Soybean, which aligns with the objective evaluation metrics in Table IV. Fig. 13(c) reveals the classification results of the CVSSN model. A large number of noise points appear within the Broad-leaf Soybean region, indicating insufficient spatial information representation capability. From Fig. 13(d), it can be observed that a significant amount of Broad-leaf Soybean appears within the dark blue Sesame region, and a host of heterogeneous pixels are present in the yellow region. In Fig. 13(e), the red region representing Corn contains Broad-leaf Soybean, which is likely due to the DBMA model's poor spatial information extraction capability. Fig. 13(f) shows the prediction results of the DBDA model. It can be exhibited that multiple different categories appear in the yellow region in the upper left corner. Both CNN-based methods and those combining CNN with attention mechanisms exhibit significant noise, demonstrating a clear tendency toward misclassification. This is primarily because CNN focuses on extracting local features, neglecting the long-range dependencies in spectral and spatial dimensions. Fig. 13(g) indicates that the SSFTT model is highly susceptible to the influence of surrounding categories, such as Corn, Cotton, and Broad-leaf Soybean, when classifying Roads and Houses. In Fig. 13(h), the BS2T model demonstrates relatively good overall performance, ranking as the second-best model after the proposed method. From Fig. 13(i), it can be observed that although the DBCTNet model leverages the Transformer architecture to capture long-range spectral dependencies, its insufficient utilization of spatial features results in the presence of multiple heterogeneous pixels in the yellow region in the upper right corner. Fig. 13(j) presents the classification results of the proposed method. The classification results for pixels within each category are smoother, and the edges are clearer. In the LK dataset, land cover classes generally occupy large areas, so leveraging spatial consistency can enhance the classification performance of the model. The visualization maps of the SSFTT and BS2T methods are similar to those of the proposed method but still exhibit relatively more noise. The fundamental reason for this difference is that the proposed method applies an improved coordinate attention mechanism to fused features at different levels, capturing more detailed features and enhancing classification performance.

IV. Discussion

In this section, patches of diverse sizes are used as the input model to evaluate the effect of varying amounts of input information on the network. Subsequently, distinct proportions of labeled samples are selected for training on the three hyperspectral datasets to verify the stability and robustness of both the comparative methods and the proposed method. Finally,

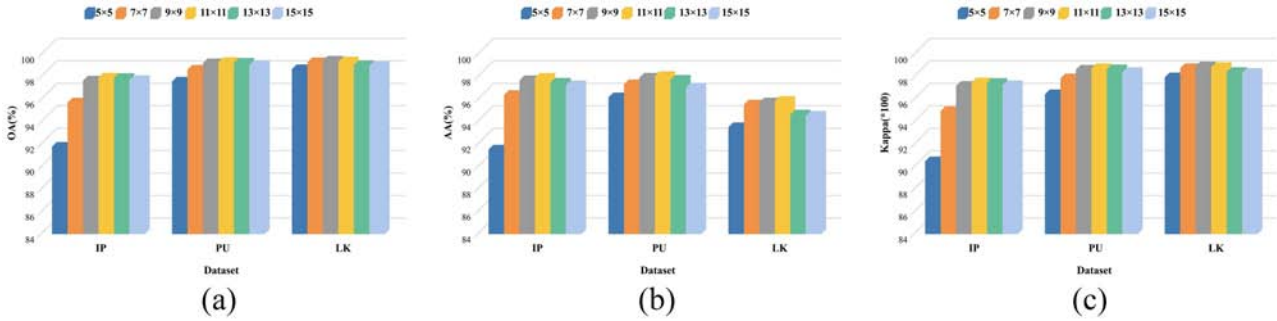

Fig. 14. Comparison of objective evaluation indicators for three datasets with different patch sizes. (a) OA. (b) AA. (c) Kappa.

the t-SNE algorithm is used to visualize the high-dimensional output features of different methods on the IP, PU, and LK datasets, demonstrating the capability of the proposed network in processing hyperspectral data.

A. Influence of Patch Size

In hyperspectral image classification, the size of the input patch determines the amount of local contextual information available to aid in classifying the central pixel, thereby influencing the overall classification performance of the model. Experiments are conducted on three hyperspectral datasets to illustrate the impact of input patch size on the classification performance of the model in different scenarios. Three datasets with varying spatial sizes, pixel counts, spatial resolutions, and scene complexities are selected to enhance the generality of the experimental results. Specifically, six different patch sizes are investigated: 5×5, 7×7, 9×9, 11×11, 13×13, and 15×15. The OA, AA, and Kappa for different input patch sizes across the three datasets are shown in Fig. 14. The illustration clearly demonstrates that as input patch size increases, the classification performance of the proposed method progresses through three distinct phases: Rapid enhancement, gradual improvement, and eventual decline. In the IP and PU datasets, classification performance increases rapidly as the patch size grows from 5×5 to 9×9. It then improves gradually, reaching optimal classification performance at 11×11, after which it begins to decline. When utilizing patch sizes of 9×9 and 11×11 as input, the model demonstrates limited performance improvement across the IP and PU datasets. Therefore, to reduce computational overhead without significantly compromising classification performance, a smaller patch size of 9×9 is selected as the input for the IP and PU datasets. In the LK datasets, classification performance increases rapidly as the patch size grows from 5×5 to 7×7. It then improves gradually, reaching optimal OA and Kappa at 9×9, after which it begins to decline. To avoid the need for optimization specific to a single datasets, a patch size of 9×9 is used as the input for the LK datasets.

B. Selection of the Proportion of Training Sample

In hyperspectral image classification, data acquisition and sample labeling incur substantial costs and require expert knowledge. Moreover, the proposed method uses a supervised learning approach, where the scale of training samples impacts the effectiveness of model training and overall performance. To verify the stability and robustness of the proposed RAT-MPC method, the performance of nine methods is observed under varying numbers of training samples. Specifically, training sample ratios of 5%, 7%, 15%, and 30% are used in the IP dataset. In the PU dataset, training sample ratios of 0.5%, 0.7%, 1.5%, and 3% are applied. For the LK dataset, the sample ratios are set to 0.1%, 0.2%, 0.5%, and 1%. Figs. 15, 16, and 17 display the OA, AA, and Kappa of the nine methods at different training ratios, respectively. In the case of a small number of samples, the proposed RAT-MPC method also achieves satisfactory classification results. As the number of training samples increases, the classification performance of the proposed method gradually improves across the three datasets, further validating its stability. The primary reason is that the RAT-MPC method utilizes a dual branch structure to effectively learn local spatial features and global spectral features, and it employs an attention mechanism to enhance fine-grained representation of the fused features, thereby improving classification performance.

C. Visualization Analysis of Features

In this section, the t-SNE algorithm is used to visualize the high-dimensional output features of different models across the three datasets. Figs. 18, 19, and 20 present the t-SNE visualizations of data distributions on the IP, PU, and LK datasets, respectively. In the experiment, the best-performing model from each category of methods is selected for comparison with the proposed model. In the IP and LK datasets, the comparative methods are SSRN, A^2S^2KResNet, and BS2T. In the PU dataset, comparisons are made with SSRN, A^2S^2KResNet, and SSFTT. In the IP dataset, the BS2T method shows significant intra-class dispersion, particularly with more pronounced intra-class separation in Corn-notill, Corn-mintill, Grass-pasture-mowed, Soybean-mintill, and Woods. In the PU dataset, the A^2S^2KResNet method shows the Bitumen class scattered within the Asphalt class. The indicates that A^2S^2KResNet struggles to effectively distinguish between these two categories, thereby reducing its classification performance. In the LK dataset, the SSRN method exhibits overlap when classifying Roads and Houses and Mixed Weed. From the visualizations of the three datasets, it can be observed that the proposed method achieves a more dis-tinct feature distribution. Although the proposed method shows minor

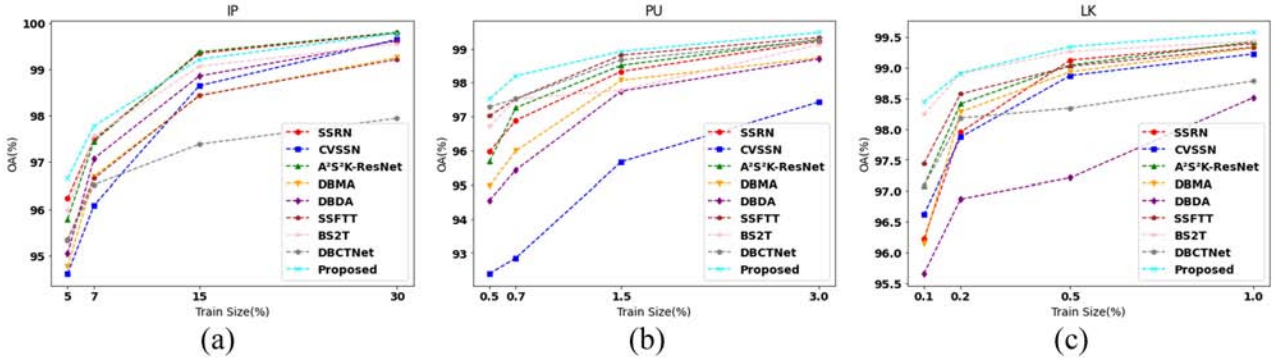

Fig. 15. Curves of OA at different percentages of training data with different methods:(a) IP. (b) PU. (c) LK.

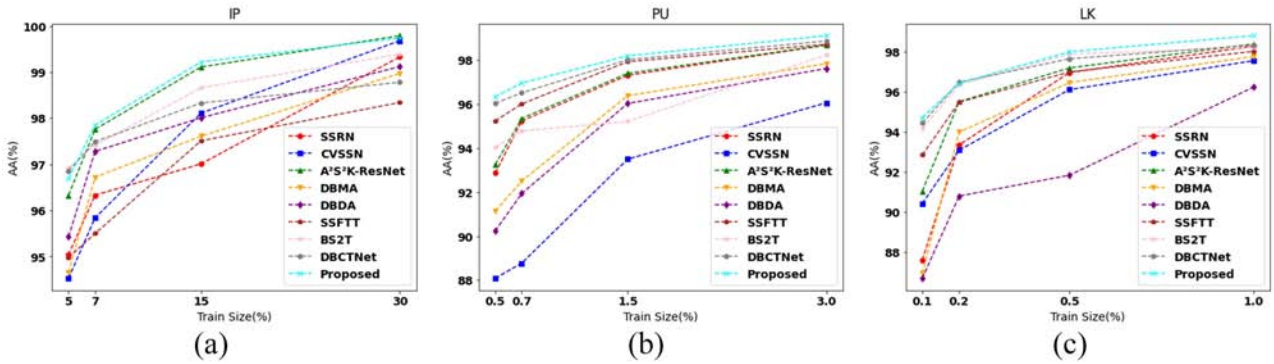

Fig. 16. Curves of AA at different percentages of training data with different methods:(a) IP. (b) PU. (c) LK.

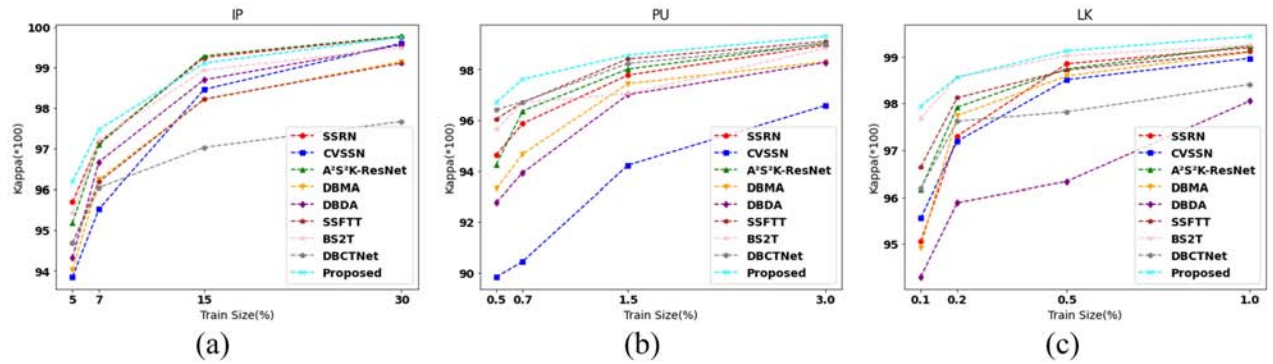

Fig. 17. Curves of Kappa at different percentages of training data with different methods: (a) IP. (b) PU. (c) LK.

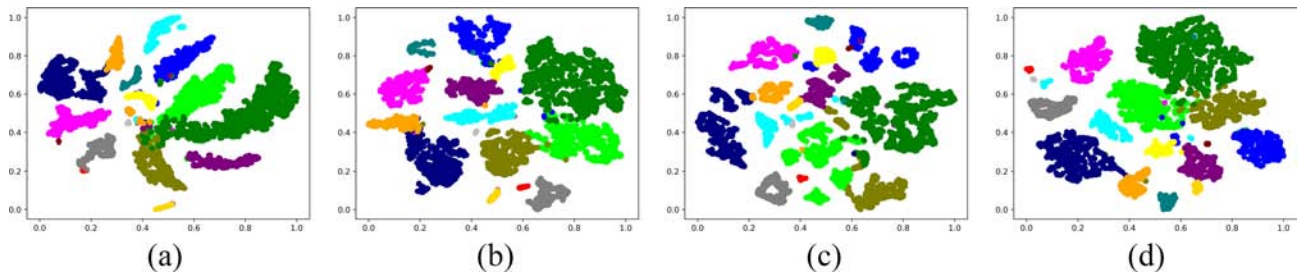

Fig. 18. Visualization analysis of features on the IP dataset. (a) SSRN. (b) A^2S^2KResNet. (c) BS2T. (d) RAT-MPC.

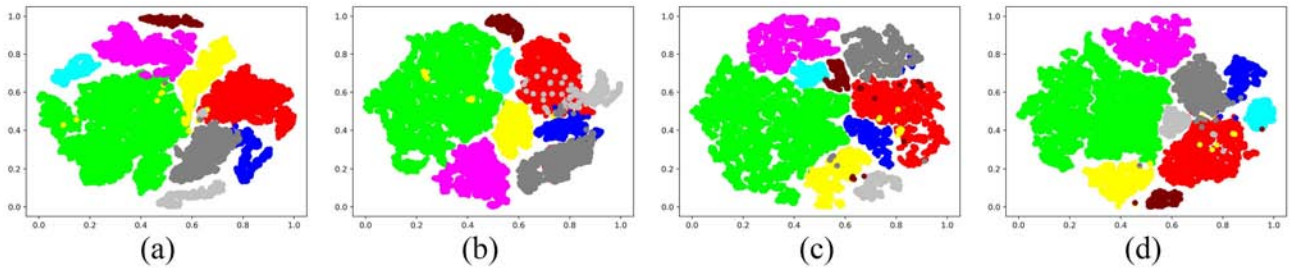

Fig. 19. Visualization analysis of features on the PU dataset. (a) SSRN. (b) A^2S^2KResNet. (c) SSFTT (d) RAT-MPC.

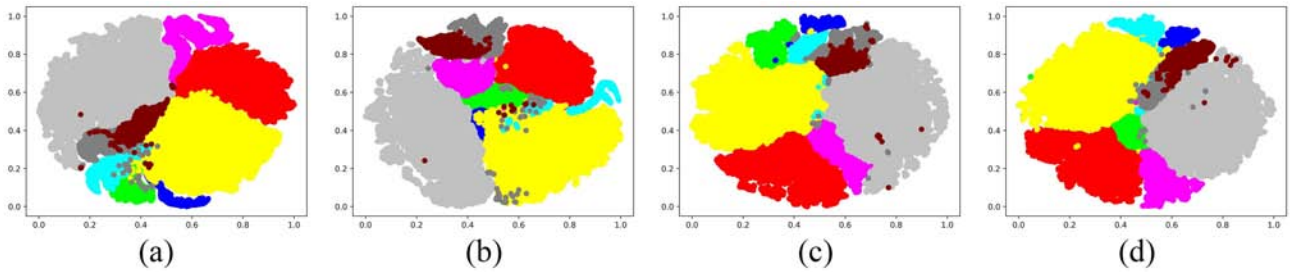

Fig. 20. Visualization analysis of features on the LK datasets. (a) SSRN. (b) A^2S^2KResNet. (c) BS2T. (d) RAT-MPC.

TABLE VI
CLASSIFICATION PERFORMANCE OF THE MODEL ON IP, PU AND LK DATASETS UNDER DIFFERENT GAUSSIAN NOISE PERTURBATION LEVELS

Datasets	Metrics	RAT-MPC	GaussianNoise-2	GaussianNoise-5	GaussianNoise-10
IP	OA	0.9666±0.0025	0.9564±0.0028	0.9499±0.0031	0.9235±0.0058
	AA	0.9669±0.0086	0.9575±0.0062	0.9503±0.0155	0.9196±0.0138
	Kappa	0.9619±0.0029	0.9502±0.0033	0.9428±0.0035	0.9126±0.0068
PU	OA	0.9820±0.0011	0.9753±0.0027	0.9683±0.0036	0.9547±0.0048
	AA	0.9696±0.0020	0.9605±0.0062	0.9532±0.0061	0.9400±0.0056
	Kappa	0.9762±0.0015	0.9672±0.0036	0.9579±0.0048	0.9400±0.0063
LK	OA	0.9844±0.0014	0.9807±0.0017	0.9784±0.0017	0.9659±0.0021
	AA	0.9472±0.0125	0.9388±0.0144	0.9353±0.0177	0.9048±0.0209
	Kappa	0.9794±0.0018	0.9746±0.0023	0.9716±0.0022	0.9552±0.0028

instances of class confusion, the feature variance within each class is minimal, and the separation between classes is notably clear.

D. Model Robustness Analysis

To further validate the model's adaptability to spectral perturbations in real-world scenarios, we introduce perturbed samples during the training phase to bolster its robustness against spectral interference. In the experiment, perturbations are applied to the PCA-reduced data, more precisely simulating interference at the final input level of the model, thereby validating its robustness against real-world spectral disturbances more effectively. Incorporating perturbations into the training set enhances the model's ability to adapt to spectral interference, uncertainties, and anomalies, thereby strengthening its robustness in complex environments. The validation set employs the original clean data to ensure stability during the model selection process. In the test set, 50% of the samples undergo random perturbations while the remainder remain intact, thereby more accurately reflecting the distribution characteristics of partially compromised samples in real-world scenarios. Moreover, the number of perturbed bands is set to 2, 5, and 10 to examine the classification performance under varying degrees of spectral disturbance.

1) Gaussian Noise Disturbance: In the experiments, three noise augmentation strategies of varying intensities are employed. Specifically, the designations GaussianNoise-2, GaussianNoise-5, and GaussianNoise-10 signify the injection of noise into two, five, and ten spectral bands of each image, respectively. The experimental results are shown in Table VI, and the overall trend shows that the OA, AA, and Kappa of the three datasets show different degrees of decrease with the gradual imposition of the noise intensity. This trend primarily stems from the fact that high-intensity noise disrupts the intrinsic structure of spectral data, thereby impeding the model's ability to extract meaningful features.

Under mild noise interference, GaussianNoise-2 exhibits only a slight decrease in accuracy compared to the original RAT-MPC model across all three datasets. This indicates that introducing mild noise into the training set enhances the model's ability to

TABLE VII
EFFECT OF DISCARD DIFFERENT NUMBERS OF SPECTRAL BANDS ON HYPERSPECTRAL IMAGE CLASSIFICATION PERFORMANCE

Datasets	Metrics	RAT-MPC	SpectralDiscard-2	SpectralDiscard-5	SpectralDiscard-10
IP	OA	0.9666±0.0025	0.9583±0.0043	0.9481±0.0032	0.9228±0.0026
	AA	0.9669±0.0086	0.9607±0.0089	0.9517±0.0089	0.9020±0.0302
	Kappa	0.9619±0.0029	0.9525±0.0049	0.9407±0.0037	0.9118±0.0029
PU	OA	0.9820±0.0011	0.9753±0.0024	0.9703±0.0021	0.9520±0.0057
	AA	0.9696±0.0020	0.9617±0.0059	0.9556±0.0062	0.9317±0.0107
	Kappa	0.9762±0.0015	0.9673±0.0032	0.9606±0.0028	0.9363±0.0074
LK	OA	0.9844±0.0014	0.9814±0.0011	0.9748±0.0019	0.9701±0.0022
	AA	0.9472±0.0125	0.9402±0.0133	0.9255±0.0127	0.9132±0.0202
	Kappa	0.9794±0.0018	0.9756±0.0014	0.9669±0.0025	0.9607±0.0029

adapt to slight spectral disturbances. In the LK dataset, its OA decreases from 98.44% to 98.07%, marking a drop of merely 0.37%. When the noise intensity is raised to a moderate level (GaussianNoise-5), a more pronounced dip in model performance occurs. In the PU dataset, the OA of GaussianNoise-5 stands at 96.83%, representing a decline of 1.37% compared to RAT-MPC. This trend stems primarily from the fact that high-intensity noise undermines the intrinsic structure of spectral data, thereby impeding the capacity of the model to extract meaningful features. This makes the model more prone to confusion, thereby resulting in a decline in classification performance. When high-intensity noise (GaussianNoise-10) is applied, the model performance deteriorates significantly, reflecting its sensitivity to substantial spectral perturbations. In the IP dataset, the OA of GaussianNoise-10 reaches 92.35%, marking a decline of nearly 4%. This indicates that high-amplitude noise perturbations have disrupted the discriminative structure of the original spectra, causing the model to become confused when distinguishing between spectrally similar classes.

In summary, moderate noise augmentation (such as GaussianNoise-2) can effectively enhance the model's robustness and stability in noisy environments without substantially compromising its accuracy. However, once the noise intensity surpasses a certain threshold, it significantly undermines the model's ability to capture critical spectral features. In practical applications, the magnitude of spectral augmentation should be meticulously calibrated to the dataset's characteristics and the task's requirements. This approach strikes an optimal balance between enhancing model robustness and maintaining high accuracy, thereby bolstering its utility and reliability in complex remote-sensing scenarios.

2) Spectral Band Dropout Perturbation: In the experiments, three distinct spectral dropout strategies of varying severity are devised. Specifically, SpectralDiscard-2, SpectralDiscard-5, and SpectralDiscard-10 represent the random discarding of 2, 5, and 10 spectral bands during training, with the corresponding bands set to zero to simulate scenarios of channel failure or occluded information loss. The experimental results, as shown in Table VII, indicate that as the number of discarded bands increases, the model performance gradually declines. This phenomenon suggests that the model possesses a certain degree of robustness against band information loss. However, once the loss exceeds a specific threshold, the integrity of the features is compromised, resulting in diminished classification capability.

Under mild dropout conditions (SpectralDiscard-2), the classification accuracy across all three datasets declines only slightly compared to the original RAT-MPC model, and performance remains at a high level. In the IP dataset, the OA decreases from 96.66% to 95.83%, a drop of merely 0.83%. Furthermore, the declines in AA and Kappa remain within 1%. In the LK dataset, the OA drops from 98.44% to 98.14%, marking a decrease of only 0.3%. This indicates that the absence of a small number of spectral bands has a limited impact on the model's overall discriminative capacity, demonstrating its robustness to minor spectral disturbances. When the dropout level increases to a moderate degree (SpectralDiscard-5), the model performance begins to decline noticeably. In the PU dataset, the OA drops from 98.20% to 97.03%, the AA falls to 95.56%, and the Kappa decreases to 96.06%, with an overall decline of approximately 1.5%. This phenomenon suggests that as the number of discarded bands increases, certain critical discriminative spectral bands may be obscured. When inter-class differences are concentrated primarily within specific bands, the model becomes more susceptible to misclassification. Under severe dropout conditions (SpectralDiscard-10), the model performance deteriorates markedly across all datasets. In the LK dataset, the OA drops from 98.44% to 97.01%. In the IP dataset, the OA decreases from 96.66% to 92.28%, with the Kappa showing a decline of up to 5%. This further illustrates that losing an excessive number of spectral bands severely disrupts the spectral structure, making it difficult for the model to effectively extract discriminative features, thereby significantly impairing its classification performance.

In summary, moderately introducing spectral dropout during training (such as SpectralDiscard-2) can enhance the model's adaptability to local band loss or channel failure, strengthening its robustness while maintaining high accuracy. However, as the number of discarded bands increases, the model's ability to acquire effective discriminative information declines noticeably, resulting in a rapid deterioration of performance. Therefore, in practical deployment, the dropout ratio should be judiciously controlled according to the task's requirements for information integrity, so as to achieve an optimal balance between interference resistance and recognition accuracy.

3) Generalizability Analysis: To verify the model's adaptability and generalization potential under real-world scenario variations, a series of cross-dataset experiments is devised. Specifically, the experiments employ the Pavia University (PU)

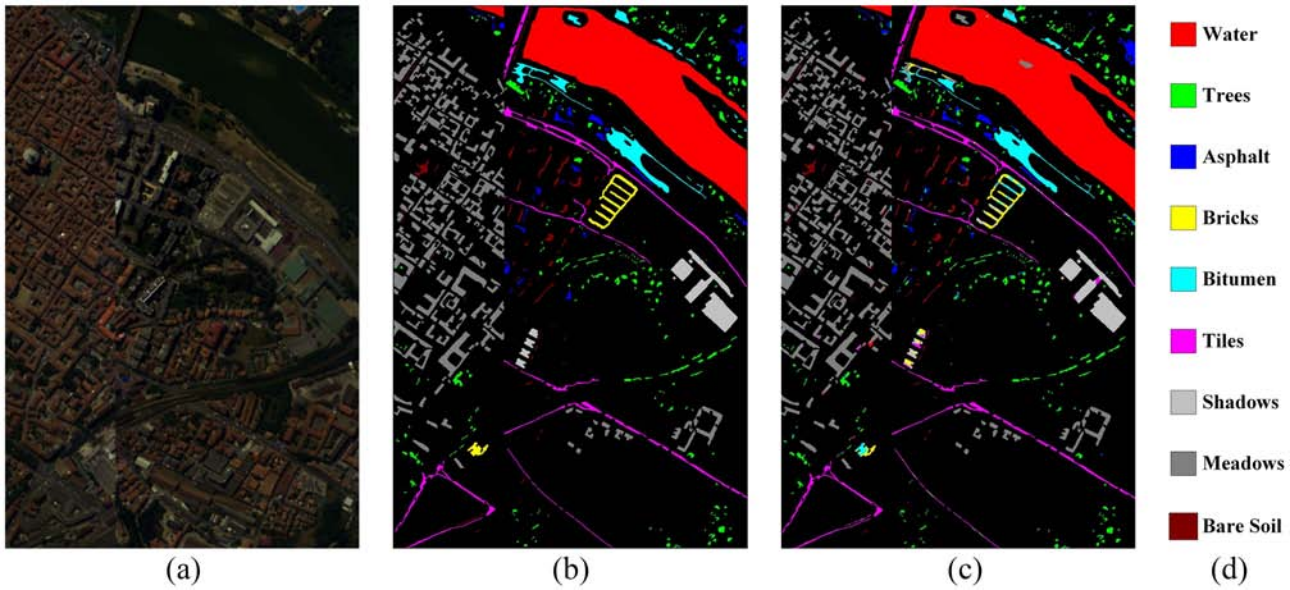

Fig. 21. Classification map on the pc dataset using the model trained on pu dataset. (a) False-color image. (b) Ground truth map. (c) Classification diagram after migration. (d) Color codes.

and Pavia Center (PC) datasets, which diverge in both spatial dimensions and spectral coverage, thus constituting an ideal proving ground for assessing the model's cross-scenario generalization capability. In the experiments, the RAT-MPC model is first trained on the PU dataset, with its configuration parameters retained. Subsequently, a set of model parameters achieving an OA close to the average level on the PU test set is selected and directly transferred to the PC dataset. To further enhance the model's adaptability to new scenarios, its parameters are subjected to lightweight transfer fine-tuning. Specifically, all network layers except the fully connected layer remain frozen, with updates applied solely to the fully connected layer. During the fine-tuning phase, 0.7% of the labeled samples are randomly selected from the PC dataset for training, adhering to the same configuration as the main experiment to ensure experimental comparability.

The PC dataset is captured by the ROSIS sensor over the central area of Pavia in northern Italy. It comprises 102 spectral bands, covering a range slightly different from that of the PU dataset, yet still spanning the 0.43 μm to 0.86 μm spectral region. Compared to the PU dataset, the PC dataset boasts higher-resolution imagery at 1096×1096 pixels and encompasses 148,152 labeled samples across nine distinct land-cover classes. The false-color image, ground truth map, and color codes are shown in Fig. 21 respectively. The subtle differences in image size and spectral coverage between the PU and PC datasets provide a valuable reference for evaluating the model's generalization performance across data sources.

The experimental results are presented in Table VIII. From a quantitative perspective, the RAT-MPC model continues to exhibit remarkable classification performance following its transfer to the PC dataset. Despite being initially trained solely on the PU dataset and undergoing only lightweight fine-tuning on the fully connected layer, the model achieves impressive results on the PC dataset, with an OA of 95.76%, an AA of 86.56%, and a Kappa coefficient of 93.98%. Although classification performance dips compared to PU, the model sustains elevated accuracy, thereby highlighting its formidable capacity for cross-scenario transfer. Notably, the decline in AA is relatively pronounced, indicating that under conditions of substantial interclass distribution disparities, the model's recognition accuracy for certain categories is adversely affected. Nevertheless, the steadfast performance of both accuracy and consistency metrics vividly underscores the RAT-MPC model's robustness and its potential to generalize across diverse scenario distributions.

TABLE VIII
CLASSIFICATION PERFORMANCE WHEN TRANSFERRING PU-TRAINED PARAMETERS TO THE PC DATASET

Metrics / Dataset	OA	AA	Kappa
PC	0.9576 $\pm$ 0.0016	0.8656 $\pm$ 0.0107	0.9398 $\pm$ 0.0022
PU	0.9820 $\pm$ 0.0011	0.9696 $\pm$ 0.0020	0.9762 $\pm$ 0.0015

The visualization results are illustrated in Fig. 21 From a qualitative perspective, the classification maps produced by RAT-MPC on the PC dataset reveal a generally clear and coherent spatial distribution across most land-cover categories. In the PC dataset, several representative land cover regions, such as large bodies of water, roads, buildings, and bare soil, present clearly defined boundaries and well-preserved structural layouts. This indicates that the model is capable of effectively capturing spatial contextual information. In complex areas such as urban cores and major transportation routes, the classification results closely align with the actual spatial layout of land cover features. This demonstrates the model's strong discriminative capability and robust resistance to noise. Although a few edge regions exhibit class confusion or localized fragmentation, the overall visual quality remains high. This further attests to the RAT-MPC model's robust generalization across varied scenarios captured by the same sensor.

V. Conclusion

In this article, a RAT-MPC method for hyperspectral image classification is proposed, enhancing feature representation through the integration of three distinct approaches. First, the MSSFL module is designed to enrich feature representation by leveraging multiscale local spectral-spatial information. Secondly, the DLGFP module is constructed to jointly extract global spectral and local spatial information. This module includes two branches: the RAT branch and the MPConv branch. On one hand, the RAT branch uses ARA to model long-range correlations between spectral bands. In addition, a proxy matrix is designed within ARA to balance computational efficiency and learning capacity, and a transformation matrix is introduced to enhance the diversity of the attention map. On the other hand, multiscale partial convolution is used to further extract abstract spatial information with fewer parameters. Third, the MFFA module aggregates diverse information from different levels and uses spatial attention mechanisms to generate more discriminative image features, further improving classification results. Further, the results of the ablation experiments demonstrate the contribution of each module within the RAT-MPC network to improving classification performance. Among methods that combine CNN and Transformer, the proposed method shows only modest improvements in classification performance compared to SSFTT and BS2T. However, its parameter count and FLOPs are reduced by approximately one-fifth and one-eighth, respectively. Compared to the DBCT method, with similar parameter counts and FLOPs, the proposed method achieves an approximate 1% improvement in OA. In brief, the RAT-MPC method adopts a dual-branch structure to fully leverage both local and global features, and introduces a proxy matrix to balance learning capacity and computational complexity of the model.

The proposed method achieves satisfactory classification results, yet there remains room for improvement. For example, in the proposed DLGFP module, the extracted global spectral information and local spatial information are fused through element-wise addition. This approach overlooks the differences in information extracted by each branch, leading to insufficient feature representation. In future work, exploring improved fusion methods to connect different features may enhance classification accuracy. In addition, the integration of multitemporal hyperspectral data with a temporal consistency modeling strategy is intended to enhance the model's robustness to seasonal spectral variations. Meanwhile, class-aware attention weighting or soft-labeling mechanisms are explored to enhance classification performance on small and edge classes.

VI. Acknowledgment

The authors would like to thank the anonymous reviewers for their helpful comments and constructive suggestions for this article.

References

[1] G. Licciardi, P. R. Marpu, J. Chanussot, and J. A. Benediktsson, "Linear versus nonlinear PCA for the classification of hyperspectral data based on the extended morphological profiles," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 3, pp. 447–451, May 2012.

[2] W. Li, C. Chen, H. Su, and Q. Du, "Local binary patterns and extreme learning machine for hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 7, pp. 3681–3693, Jul. 2015.

[3] C. Cariou and K. Chehdi, "A new k-nearest neighbor density-based clustering method and its application to hyperspectral images," in *Proc. 2016 IEEE Int. Geosci. Remote Sens. Symp.*, IEEE, 2016, pp. 6161–6164.

[4] Y.-N. Chen, T. Thaipisutikul, C.-C. Han, T.-J. Liu, and K.-C. Fan, "Feature line embedding based on support vector machine for hyperspectral image classification," *Remote Sens.*, vol. 13, no. 1, 2021, Art. no. 130.

[5] Q. Ye, P. Huang, Z. Zhang, Y. Zheng, L. Fu, and W. Yang, "Multiview learning with robust double-sided twin SVM," *IEEE Trans. Cybern.*, vol. 52, no. 12, pp. 12745–12758, Dec. 2022.

[6] Z. Liu, B. Tang, X. He, Q. Qiu, and F. Liu, "Class-specific random forest with cross-correlation constraints for spectral–spatial hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 2, pp. 257–261, Feb. 2017.

[7] Y. Zhang, G. Cao, X. Li, and B. Wang, "Cascaded random forest for hyperspectral image classification," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 11, no. 4, pp. 1082–1094, Apr. 2018.

[8] B. Tu, C. Zhou, J. Peng, G. Zhang, and Y. Peng, "Feature extraction via joint adaptive structure density for hyperspectral imagery classification," *IEEE Trans. Instrum. Meas.*, vol. 70, 2021, Art. no. 5006916.

[9] Y. Ding, Y. Guo, Y. Chong, S. Pan, and J. Feng, "Global consistent graph convolutional network for hyperspectral image classification," *IEEE Trans. Instrum. Meas.*, vol. 70, 2021, Art. no. 5501516.

[10] X. Wu, D. Hong, and J. Chanussot, "Convolutional neural networks for multimodal remote sensing data classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5517010.

[11] S. Bera, V. K. Shrivastava, and S. C. Satapathy, "Advances in hyperspectral image classification based on convolutional neural networks: A review.," *CMES- Comput. Model. Eng. Sci.*, vol. 133, no. 2, pp. 219–250, 2022.

[12] H. Xu, W. Yao, L. Cheng, and B. Li, "Multiple spectral resolution 3D convolutional neural network for hyperspectral image classification," *Remote Sens.*, vol. 13, no. 7, 2021, Art. no. 1248.

[13] H. Gao, Y. Yang, C. Li, L. Gao, and B. Zhang, "Multiscale residual network with mixed depthwise convolution for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 4, pp. 3396–3408, Apr. 2021.

[14] T. Wang, H. Liu, and J. Li, "Spectral–spatial classification of few shot hyperspectral image with deep 3-D convolutional random fourier features network," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5532318.

[15] H. Yu, H. Zhang, Y. Liu, K. Zheng, Z. Xu, and C. Xiao, "Dual-channel convolution network with image-based global learning framework for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6005705.

[16] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *J. Sensors*, vol. 2015, no. 1, 2015, Art. no. 258619.

[17] G. Cheng, Z. Li, J. Han, X. Yao, and L. Guo, "Exploring hierarchical convolutional features for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 11, pp. 6712–6722, Nov. 2018.

[18] H. Pan, Y. Zhu, H. Ge, M. Liu, and C. Shi, "Multiscale cross-fusion network for hyperspectral image classification," *Egyptian J. Remote Sens. Space Sci.*, vol. 26, no. 3, pp. 839–850, 2023.

[19] S. Ghaderizadeh, D. Abbasi-Moghadam, A. Sharifi, N. Zhao, and A. Tariq, "Hyperspectral image classification using a hybrid 3D-2D convolutional neural networks," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 7570–7588, 2021.

[20] Z. Lu, B. Xu, L. Sun, T. Zhan, and S. Tang, "3-D channel and spatial attention based multiscale spatial–spectral residual network for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 4311–4324, 2020.

[21] X. Qiao, H. Wu, S. K. Roy, and W. Huang, "Hyperspectral image classification based on 3D sharpened cosine similarity operation," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, IEEE, 2023, pp. 7669–7672.

[22] S.-Y. Chen, K.-H. Hsu, and T.-H. Kuo, "Hyperspectral target detection-based 2D-3D parallel convolutional neural networks for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 9451–9469, 2024.

[23] R. Song, Y. Feng, W. Cheng, Z. Mu, and X. Wang, "BS2T: Bottleneck spatial–spectral transformer for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5532117.

[24] Y. Gao, M. Zhang, J. Wang, and W. Li, "Cross-scale mixing attention for multisource remote sensing data fusion and classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5507815.

[25] K. Wu, J. Fan, P. Ye, and M. Zhu, "Hyperspectral image classification using spectral–spatial token enhanced transformer with hash-based positional embedding," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5507016.
[26] B. Zhang, Y. Chen, Y. Rong, S. Xiong, and X. Lu, "MATNet: A combining multi-attention and transformer network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5506015.
[27] Y. Qing, W. Liu, L. Feng, and W. Gao, "Improved transformer net for hyperspectral image classification," *Remote Sens.*, vol. 13, no. 11, 2021, Art. no. 2216.
[28] X. Qiao, S. K. Roy, and W. Huang, "Multiscale neighborhood attention transformer with optimized spatial pattern for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5523815.
[29] H. Yu, Z. Xu, K. Zheng, D. Hong, H. Yang, and M. Song, "MSTNet: A multilevel spectral–spatial transformer network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5532513.
[30] D. Hong et al., "SpectralFormer: Rethinking hyperspectral image classification with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5518615.
[31] S. Mei, C. Song, M. Ma, and F. Xu, "Hyperspectral image classification using group-aware hierarchical transformer," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5539014.
[32] C. Shi, S. Yue, and L. Wang, "A dual-branch multiscale transformer network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5504520.
[33] J. Zou, W. He, and H. Zhang, "LESSFormer: Local-enhanced spectral-spatial transformer for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5535416.
[34] D. Niu, X. Zhang, L. Li, and Y. Zhou, "Hsi-sstrans: Hyperspectral image classification with spectral and space transformer," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2023, pp. 7625–7628.
[35] J. Zhu, G. Cao, J. Bei, Y. Zhang, and Y. Han, "Fusion of dilated convolution in CNN and transformer networks for hyperspectral image classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2024, pp. 9991–9995.
[36] Y. Cao, Y. Wang, Z. Yin, and Z. Yang, "Mixed residual convolutions with vision transformer in hyperspectral image classification," in *Proc. IEEE 22nd Int. Conf. Commun. Technol.*, 2022, pp. 1595–1599.
[37] R. Sun, J. Xiang, and L. Wang, "Joint multi-scale CNN and vision transformer for hyperspectral image classification," in *Proc. IEEE 2nd Int. Conf. Control Electron. Comput. Technol.*, 2024, pp. 364–369.
[38] L. Sun, G. Zhao, Y. Zheng, and Z. Wu, "Spectral–spatial feature tokenization transformer for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5522214.
[39] J. Hu, B. Tu, Q. Ren, X. Liao, Z. Cao, and A. Plaza, "Hyperspectral image classification via multiscale multiangle attention network," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5510718.
[40] Q. Yu, W. Wei, D. Li, Z. Pan, C. Li, and D. Hong, "HyperSINet: A synergetic interaction network combined with convolution and transformer for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5508118.
[41] H. Yang, H. Yu, K. Zheng, J. Hu, T. Tao, and Q. Zhang, "Hyperspectral image classification based on interactive transformer and CNN with multilevel feature fusion network," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, 2023, Art. no. 5507905.
[42] D. Zhou et al., "DeepViT: Towards deeper vision transformer," 2021, *arXiv:2103.11886*.
[43] D. Han et al., "Agent attention: On the integration of softmax and linear attention," in *Proc. Eur. Conf. Comput. Vis.*, 2024, pp. 124–140.
[44] 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. 12021–12031.
[45] Q. Liu, L. Xiao, N. Huang, and J. Tang, "Composite neighbor-aware convolutional metric networks for hyperspectral image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 7, pp. 9297–9311, Jul. 2024.
[46] Q. Liu, L. Xiao, J. Yang, and J. C.-W. Chan, "Content-guided convolutional neural network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 9, pp. 6124–6137, Sep. 2020.
[47] R. Xu, X.-M. Dong, W. Li, J. Peng, W. Sun, and Y. Xu, "DBCTNet: Double branch convolution-transformer network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5509915.

Junding Sun received the B.S. degree in computer application and M.S. degree in control theory and control engineering from Henan Polytechnic University, Jiaozuo, Henan, China, in 1998 and 2001, respectively, and the Ph.D. degree in computer application from Xidian University, Xi'an, China, in 2005.

He is currently a Professor with the School of Computer Science and Technology, Henan Polytechnic University. His major research interests include image processing, image retrieval, and pattern recognition.

Hongyuan Zhang received the M.S. degree in computer science and technology from Henan Polytechnic University, Jiaozuo, China, in 2025. He is currently working toward the Ph.D. degree in software engineering with the School of Computer and Information Engineering, Henan University.

His primary research interests include hyperspectral image classification.

Jianlong Wang received the B.S. degree in computer science and technology from Zhengzhou University, Zhengzhou, China, in 2012, the M.S. degree in computer science and technology from Henan Polytechnic University, Jiaozuo, China, in 2015, and the Ph.D. degree in electronic science and technology from Xidian University, Xi'an, China, in 2021.

He is currently a Associate Professor with the School of Computer Science and Technology, Henan Polytechnic University. His research interests include the application of image processing, machine learning, deep learning in smart agriculture, and polarimetric synthetic aperture radar land cover classification.

Haifeng Sima received the B.E. and M.E. degrees in computer science from Zhengzhou University, Zhengzhou, China, in 2004 and 2007, respectively, and the Ph.D. degree in software and theory from the Beijing Institute of Technology, Beijing, China, in 2015.

Since 2007, he has been with the Faculty of Henan Polytechnic University, Jiaozuo, China, where he is currently an Associate Professor with the School of Computer Science and Technology. His current research interests include pattern recognition, image processing, image segmentation, and image classification.

Shuanggen Jin (Senior Member, IEEE) was born in Anhui, China, in September 1974. He received the B.Sc. degree in geodesy from Wuhan University, Wuhan, China, in 1999, and the Ph.D. degree in geodesy from the University of Chinese Academy of Sciences, Beijing, China, in 2003.

He has authored or coauthored more than 500 papers in peer-reviewed journals and proceedings, 30 patents/software copyrights, 3 codes and standards, and 15 books/monographs with more than 16 000 citations and H-index>70. He is currently Vice-President and Professor with Henan Polytechnic University, Jiaozuo, China, and also Professor with Shanghai Astronomical Observatory, CAS, Shanghai, China. His main research interests include satellite navigation, remote sensing, and space/planetary exploration.

Dr. Jin was the recipient of 100- Talent Program of CAS, IUGG Fellow, IETI Fellow, IAG Fellow, AAIS Fellow, EMA Fellow, World Class Professor of Ministry of Education and Cultures, Indonesia, Chief Scientist of National Key R & D Program, China, Member of European Academy of Sciences, Member of Turkish Academy of Sciences, and Member of Academia Europaeahas been President of International Association of Planetary Sciences (IAPS) (2015–2019), President of the International Association of Chinese Professionals in GNSS (CPGNSS) (2016–2017), Chair of IUGG Union Commission on Planetary Sciences (UCPS) (2015–2027), Editor-in-Chief of *International Journal of Geosciences*, Associate Editor for IEEE TRANSACTIONS ON GEOSCIENCE & REMOTE SENSING and *Journal of Navigation*, Editorial Board member of *GPS Solutions* and *Journal of Geodynamics*.