A Cross-Attention-Based Multi-Information Fusion Transformer for Hyperspectral Image Classification

Jinghui Yang, Anqi Li, Jinxi Qian, Jia Qin, and Liguo Wang

***Abstract*—In recent years, deep-learning-based classification methods have been widely used for hyperspectral images (HSIs). However, in the existing transformer-based HSI classification methods, how to effectively and comprehensively utilize the rich information still has room for improvement, for example, when utilizing multiple-image information, the comprehensive interaction between information has insufficient consideration. To address the above issues, cross-attention interaction, class token and patch token information, and multiscale spatial information are addressed in a unified framework, and a cross-attention-based multi-information fusion transformer (CAMFT) for HSI classification was proposed, which includes the multiscale patch embedding module, the residual connection-based DeepViT (RCD) module, and the double-branch cross-attention (DBCA) module. First, the multiscale patch embedding module is formed for multi-information preprocessing, accompanied by the built of different scale processing branches and the addition of learnable class tokens. Second, the RCD module is designed to utilize rich information from different layers; this module includes reattention and residual connection. Third, a DBCA module is constructed to obtain more representative multi-information fusion features; this module not only integrates multiscale patch information but also effectively utilizes complementary information between class tokens and patch tokens in the interaction of two branches. Moreover, numerous experiments demonstrate that, compared with other state-of-the-art classification methods, the proposed CAMFT method achieves the optimal classification performance, especially with a small training sample size, but it still has excellent performance.**

***Index Terms*—Classification, cross-attention, hyperspectral image (HSI), multi-information fusion, transformer.**

I. Introduction

With the advancements in equipment, remote sensing (RS) imaging technology is constantly evolving, and the spectral resolution is improving, thus giving rise to hyperspectral RS technology. Unlike the traditional single-band RS and multispectral RS, hyperspectral RS captures a more comprehensive range of spectral information [1] and acquires 3-D image data in the continuous electromagnetic spectrum using spectral cameras. During observation, hyperspectral RS images capture the information of covered objects from both spatial and spectral dimensions simultaneously and continuously image them in continuous spectral bands, ultimately forming a 3-D cube of data, known as a hyperspectral image (HSI).

With the maturity of hyperspectral RS imaging technology, the obtained HSI data have become more accurate due to improvements in spatial and spectral resolution [2], [3]. Benefiting from the rich spectral–spatial information, HSIs have important applications in many fields, such as urban planning, precision agriculture, mineral exploration [4], biomedicine [5], food safety, and environmental monitoring [6]. Due to the increasing quality and widespread application of HSIs, researchers have focused more on intelligent processing methods. Therefore, many urgent issues have been raised [7], and how to achieve high-precision and high-efficiency classification of HSIs [8], [9] is one of the research issues to be solved.

In recent years, deep-learning theory has led to a wave of artificial intelligence and has gained the favor of researchers in various fields [10], [11], [12]. It has made significant breakthroughs in image processing [13], speech recognition, and text processing [14] and has also promoted technological innovation in the field of HSI processing [15], [16]. Currently, many deep-learning-based classification methods have been proposed for HSIs [17], [18], [19], [20].

With the popularity of convolutional neural networks (CNNs), CNN-based HSI classification methods have been proposed. Zhao and Du [21] proposed a 2-D CNN-based HSI classification model by employing PCA for dimensionality reduction and 2-D CNN for spectral–spatial feature extraction. Zhong et al. [22] developed an end-to-end spectral–spatial residual network that continuously learns discriminative features from rich spectral and spatial context information by using spectral and spatial residual blocks. A double-branch multiattention mechanism network [23] extracts spectral and spatial features separately through two branches to reduce the interference between different features. Roy et al. [24] developed a hybrid spectral CNN (HybridSN) model, a hierarchical convolutional network that combines 3-D and 2-D CNNs. In addition to effectively extracting spatial–spectral features, the HybridSN model can also reduce computational complexity and improve classification accuracy. Zhu et al. [25] designed the residual spectral–spatial attention network model, which considers the information of

Manuscript received 15 April 2024; revised 20 May 2024; accepted 11 July 2024. Date of publication 17 July 2024; date of current version 5 August 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62001434 and Grant 62071084, and in part by the National Key R&D Program of China under Grant 2023YFF1303804. *(Corresponding author: Jinghui Yang.)*

Jinghui Yang, Anqi Li, and Jia Qin are with the School of Information Engineering, China University of Geosciences (Beijing), Beijing 100083, China (e-mail: yang06081102@163.com; liqaq@email.cugb.edu.cn; qjia@email.cugb.edu.cn).

Jinxi Qian is with the Institute of Telecommunication and Navigation Satellites, China Academy of Space Technology, Beijing 100094, China (e-mail: emperor07@163.com).

Liguo Wang is with the College of Information and Communication Engineering, Dalian Minzu University, Dalian 116600, China (e-mail: wangliguo@hrbeu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2024.3429492

spectral bands and spatial positions through spectral attention modules and spatial attention modules, respectively. Zhang et al. [26] proposed a double-branch structure comprising CNN and transformer branches to capture local–global hyperspectral features. A group parallel residual block was used before the double-branch structure to capture local spectral–spatial features in HSI patches. Additionally, a convolution operation was introduced into the multihead self-attention (MHSA) mechanism to enhance the classification accuracy further. Although CNN dominates many computer vision fields due to its powerful local feature extraction capability, it is unable to effectively capture the global features of HSIs due to the limited receptive field of the convolution kernel.

Recently, transformers have achieved great success in natural language processing, challenging the dominance of CNNs in computer vision tasks. The vision transformer (ViT) is one of the first attempts to achieve performance comparable to CNNs on image classification tasks using a pure transformer architecture [27]. However, due to its high model complexity, ViT requires pretraining on larger datasets to achieve relatively good performance. To address the problem of data efficiency, data-efficient image transformers (DeiT) [28] deploy knowledge distillation to train models with larger pretrained teacher models. Through this approach, DeiT can perform well on ImageNet-1k without requiring pretraining on a larger dataset. The pyramid ViT [29] integrates transformers into CNNs and can be trained on dense partitions of images to achieve high output resolution. Tokens-to-token ViT [30] recursively aggregates adjacent tokens into one token, and the image is gradually structured into tokens, providing a deeper and narrower high-efficiency backbone. MobileViT [31] significantly outperforms other lightweight networks by combining the inverse residual and ViT. When the ViT layers are progressively deepened, the attention collapse problem may occur. To address this issue, Zhou et al. [32] proposed a simple but effective method called DeepViT, which regenerates attention maps to increase the diversity in different layers. Chen et al. [33] proposed CrossViT, which combines different image patch sizes to produce more discriminative features. Fan et al. [34] proposed a multiscale visual transformer for image and video recognition by integrating the hierarchical structure of multiscale features with the transformer model. The Swin transformer (SwinT) [35] restricts self-attention computations to nonoverlapping localized windows by shifting them while allowing cross-window connections. Conformer [36] designed a feature coupling unit by combining the advantages of CNN and transformer; this unit interacts with the local and global features of each stage. Srinivas et al. [37] regarded ResNet bottleneck blocks with self-attention as transformer blocks and improved upon the baselines significantly on instance segmentation and object detection while also reducing the number of parameters. Wu et al. [38] improved ViT performance and efficiency by incorporating convolutions, with fewer parameters and FLOPs.

Similarly, the transformer has been applied to HSI classification in many studies. Hong et al. [39] used ViT for HSI classification and achieved good classification performance. He et al. [40] proposed a spatial–spectral feature transformer model that utilizes a convolutional network structure similar to the visual geometry group network to extract spatial features and construct the relationship between adjacent spectra using densely connected transformers. Zhong et al. [41] proposed the spectral–spatial transformer network (SSTN), which comprises spatial attention and spectral correlation modules, to overcome the limitations of convolution kernels. Sun et al. [42] designed a spectral–spatial feature tokenization transformer (SSFTT) method, which captures both spectral–spatial and high-level semantic features simultaneously. Liu et al. [43] proposed a deep-spatial–spectral transformer and utilized the self-attention mechanism by replacing convolutional layers with transformer layers to improve HSI classification performance. The spectrally enhanced and densely connected transformer model [44] adds dense connectivity to fuse features from shallow to deep layers while also utilizing the transformer's ability to extract global features. In [45], an end-to-end inception transformer network (IFormer) was proposed to extract high- and low-frequency information. Feng et al. [46] introduced a new spectral transformer model with dynamic spatial sampling and Gaussian position embedding. Liu et al. [47] proposed an HSI transformer iN transformer to fuse local and global features. Xue et al. [48] proposed a novel local transformer with spatial partition restore network for HSI classification. Zhou et al. [49] proposed the mobile 3-D convolutional vision transformer (MDvT), which combines 3-D convolution with the transformer. Peng et al. [50] proposed a novel convolutional transformer-based few-shot learning method for cross-domain HSI classification. Zhao et al. [51] proposed a multiple-vision architectures-based hybrid network (MVAHN) to consider multiple-feature information. Yang et al. [52] proposed a novel HSI classification method based on the pyramid feature extraction with deformable-dilated convolution (PD^{2}C). Gao et al. [53] proposed a method named main–sub transformer network with spectral–spatial separable convolution. In addition, drawing on cross-attention techniques, He et al. [54] proposed a cross-spectral vision transformer to extract pixelwise multiscale features. Peng et al. [55] proposed a spatial–spectral transformer with cross attention; it can capture spatial–spectral semantic feature information. In [56], the cross spatial–spectral dense transformer (CS2DT) is proposed to extract spatial and spectral features.

It can be seen that how to fully utilize the rich information in HSIs is a research hotspot in classification. In the existing transformer-based HSI classification methods, how to effectively and comprehensively utilize the rich information still has room for improvement, for example, when utilizing multiple-image information, the comprehensive interaction between information has insufficient consideration. Inspired by the studies of DeepViT [32] and CrossViT [33], to address the above issues, a cross-attention-based multi-information fusion transformer (CAMFT) for HSI classification was proposed, which mainly includes a multiscale patch embedding module, a residual connection-based DeepViT (RCD) module, and a double-branch cross-attention (DBCA) module. First, according to the central pixel, patches of two different scales based on large and small scales are formed, and two processing branches are built. By passing through linear layers, the patches are mapped into patch embeddings, and the learnable class token

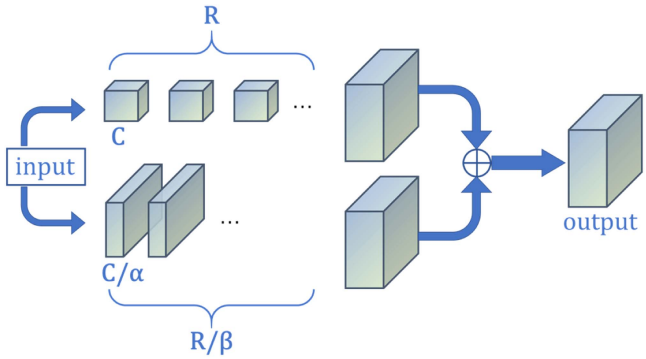


Fig. 1. Illustration of the big–little Net module.

and position embeddings are also added. Subsequently, through the RCD module, the diversity of features in different layers is considered, the information in the intermediate layer is effectively utilized, and the class token and patch token information are also integrated. Next, the DBCA module not only integrates multiscale patch information but also effectively utilizes complementary information between class tokens and patch tokens in the interaction of two branches to obtain more representative features. Finally, the classification task is accomplished based on the obtained more representative multi-information fusion features.

In summary, the main contributions of this work are as follows.

1) As far as we know, cross-attention interaction, class token and patch token information, and multiscale spatial information are addressed in a unified framework for the first time for HSI classification, and a cross-attention-based multi-information fusion transformer, namely, CAMFT, is proposed. The CAMFT method not only considers the interaction information between class tokens and patch tokens but also fully utilizes multiscale patch information while combining the generated multilayer diversity information, ultimately forming more discriminative image features based on multi-information fusion.
2) A DBCA module is constructed to obtain more representative features. The DBCA module not only integrates multiscale patch information through two branches—a large-scale branch and a small-scale branch—but also effectively utilizes complementary information between class tokens and patch tokens in the interaction of two branches to obtain more representative features. Moreover, by using the convolution–pooling operation, more nonlinear features are obtained, and the unnecessary information in intermediate processes is reduced, thus reducing computational costs.
3) An RCD module is designed to utilize rich information from different layers. The RCD module regenerates attention maps by exchanging information from different attention heads, preserving feature diversity in different layers. Moreover, through residual connections, information in the intermediate layer is effectively utilized, smoothing the forward and backward propagation of information and reducing the problem of network degradation. Additionally, according to the input, the class token and patch token information are also integrated.

The rest of this article is organized as follows. Section II provides a brief review of related works. Section III describes the proposed CAMFT method in detail. The experiments and analyses are given in Section IV. Finally, Section V concludes this article.

II. Related Works

This section briefly reviews the principles of the transformer and introduces the big–little net and CrossViT network frameworks.

A. Transformer

Through the introduction of the above references [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], it can be seen that the transformer was initially used in natural language processing. More recently, it has also been applied to computer vision, producing the ViT and demonstrating strong performance in image classification.

The ViT model consists of three main components: a linear layer for patch embedding, a stack of transformer blocks with multiple self-attention and feedforward layers for feature encoding, and a linear layer for classification prediction.

The transformer is widely used in natural language processing to encode input word token sequences into embedding sequences. To follow this sequence-to-sequence learning structure, when processing images, ViT first evenly divides the input image into patches and encodes each patch as a token embedding. All these tokens are then fed into the transformer block stack along with the class tokens. Each transformer block consists of an MHSA layer and a feedforward multilayer perceptron (MLP). The MHSA generates a trainable associative memory with a query (Q) and a pair of key (K) value (V) pairs by linearly transforming the inputs and obtaining the output.

B. Big–Little Net

Multiscale feature representation has been proved to be effective for various tasks; however, its computational complexity remains challenging. The computational cost of CNNs is closely related to the size of the input image. Based on this, Chen et al. [57] developed a two-branch structure called big–little Net, in which each branch has a different scale and can fuse multiscale image features.

Fig. 1 shows an illustration of the big–little Net module, where the upper branch is the big branch with a small image scale and deep layers, and the lower branch is the little branch with a large image scale and shallow layers. R is the number of layers in the big branch, C is the number of channels, and α and β control the width and depth of the little branch, respectively.

It can be seen that after passing through the big–little Net module, the output consists of the fused features of the big branch

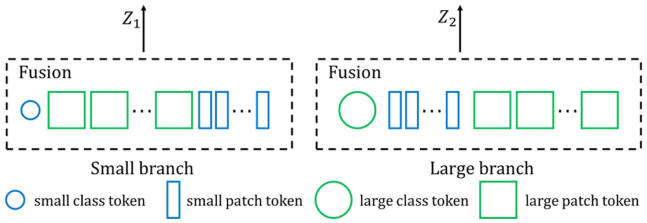


Fig. 2. Illustration of cross-attention fusion.

and little branch as shown in the following equation:

$$\text{output} = F(w_1 S_1(f_1(\text{input})) + w_2 S_2(f_2(\text{input}))) \quad (1)$$

where subscript 1 denotes the big branch, and subscript 2 denotes the little branch, and $f(\cdot)$ denotes the sequence of convolution layers of the branch. $S(\cdot)$ is an operation used to match the output size of the branch. This can be achieved either by increasing the number of feature maps through a 1×1 convolution or by an upsampling operation. The weights of different branches are denoted by w, and $F(\cdot)$ denotes the fusion layer.

C. CrossViT

Inspired by big–little Net, CrossViT [33] first introduces a double-branch ViT based on the small and large branches, where each branch operates at different scale sizes and then achieves information interaction by fusing the information between branches. CrossViT not only adds a position embedding at the head position of the two branches but also adds an additional class token. Then, all the tokens are passed through a stacked transformer encoder, followed by cross attention to obtain the fused information of different branches; finally, the fused information is used for classification.

Fig. 2 shows an illustration of the cross-attention information fusion of the small branch and large branch. Taking the small branch as an example, its input contains the class token of the small branch and the patch tokens of small and large branches. Subsequently, information fusion is carried out to obtain the output z_1 of the corresponding small branch, as shown in the following equation:

$$z_1 = \text{concat}\left(X^S_{\text{patch}}, X^{S'}_{\text{cls}}\right) \quad (2)$$

where X^S_{patch} is the patch token of the small branch and $X^{S'}_{\text{cls}}$ denotes the output result obtained after the attention calculation is performed on the class token of the small branch and the patch token of the large branch.

Similarly, the final output of the large branch z_2 is shown as follows:

$$z_2 = \text{concat}\left(X^L_{\text{patch}}, X^{L'}_{\text{cls}}\right) \quad (3)$$

where X^L_{patch} is the patch token of the large branch and $X^{L'}_{\text{cls}}$ denotes the output result obtained after the attention calculation is performed on the class token of the large branch and the patch token of the small branch.

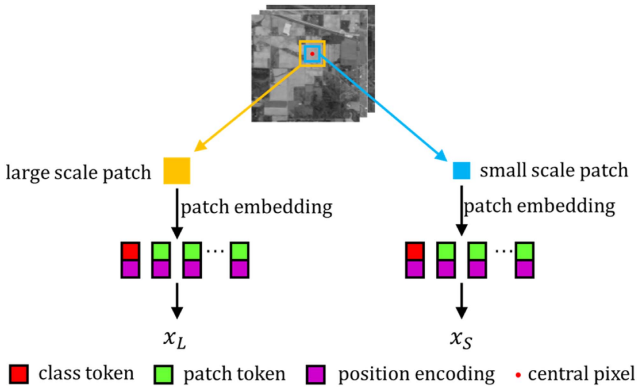


Fig. 3. Illustration of the multiscale patch embedding module.

III. Proposed CAMFT Method

To fully utilize the rich information in HSIs, a cross-attention-based multi-information fusion transformer for HSI classification, namely, CAMFT, is proposed. It mainly includes the multiscale patch embedding module, the RCD module, and the DBCA module.

A. Multiscale Patch Embedding Module

The HSI data are divided into patches according to the central pixel. Patches of two different scales based on the large and small scales are formed, and two processing branches are built. By passing through linear layers, the patches in each branch are mapped into patch embeddings, and the learnable class token and position embeddings are also added. The final token sequences x_S and x_L are used as inputs for the subsequent operations. The subscripts S and L denote the small branch and large branch, respectively, as shown in Fig. 3. The processing of the multiscale patch embedding module can facilitate subsequent operations to capture information at different scales. Moreover, this approach can maintain local continuity in HSIs, effectively capturing contextual information from local regions and integrating this information into a global feature representation, contributing to the comprehensive representation of multiple features in images.

B. RCD Module

To utilize rich information from different layers in HSIs and to avoid the problem that the attention maps gradually become similar during the process of deepening the number of transformer block layers, the RCD module based on DeepViT was designed. The RCD module regenerates attention maps by exchanging information from different attention heads, preserving the diversity of features in different layers. Moreover, through residual connections, the network can effectively utilize the information in the intermediate layer. Additionally, according to the input, the class token and patch token information are also integrated. A specific illustration of the proposed RCD module is shown in Fig. 4, and the main core includes reattention and residual connection. Reattention [32] uses head attention

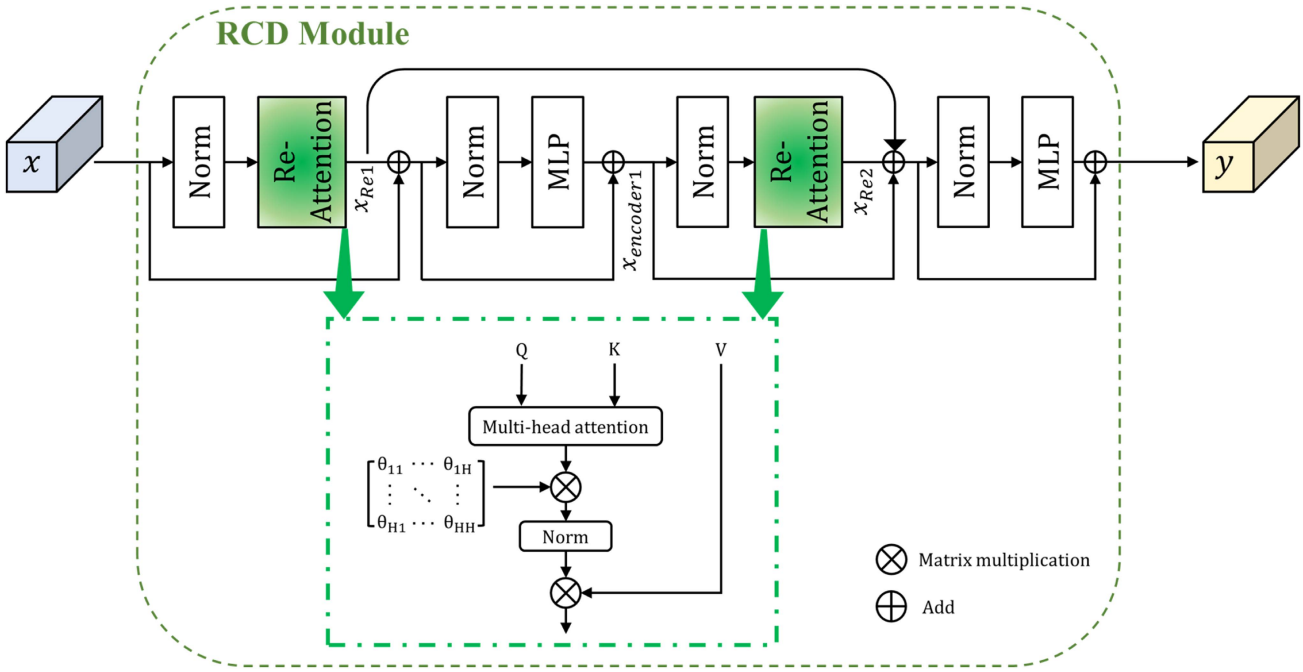


Fig. 4. Illustration of the proposed RCD module.

maps as the base and generates a new set of attention maps by dynamically aggregating them. It mixes the multihead attention maps into regenerated new attention maps through a learnable transformation matrix $\theta \in \mathbb{R}^{H\times H}$, which is then multiplied by V as shown in the following equations:

$$\text{Reattention}(Q, K, V) = \text{Norm}(\theta^{\text{T}} A)V \tag{4}$$

$$A = \text{Softmax}\left(\frac{QK^{\text{T}}}{\sqrt{d}}\right) \tag{5}$$

where the superscript T denotes the transpose operation and $\sqrt{d}$ is the scaling factor. Norm is the layer normalization function used to reduce the layerwise variance.

The DeepViT encoder can be represented as follows:

$$x_{\text{Re}} = \text{Reattention}(\text{LN}(x)) \tag{6}$$

$$x_{\text{encoder}} = \text{MLP}(\text{LN}(x_{\text{Re}} + x)) + (x_{\text{Re}} + x) \tag{7}$$

where x is the input, x_{Re} is the result of reattention, LN is the layer norm operation, MLP is the multilayer perceptron, and x_{encoder} is the output of the DeepViT encoder.

Additionally, exchanging information from different attention heads to regenerate attention maps can preserve the diversity of HSI features in different layers.

To fully utilize the intermediate layer information, the DeepViT encoder was stacked twice and the intermediate features were integrated with residual connection. The output information of the first reattention is connected to the output of the second reattention through a skip connection, thus creating intermediate layer information transmission between the two encoders and allowing for full utilization of the information in the intermediate layer. This approach can smooth forward and backward propagation of information and reduce the network degradation problem to a certain extent. The output y of the RCD module is presented in (8)–(11)

$$\begin{aligned} y &= \text{RCD}(x) \\ &= \text{MLP}(\text{LN}(x_{\text{Re2}} + x_{\text{Re1}} + x_{\text{encoder1}})) + x_{\text{Re2}} \\ &\quad + x_{\text{Re1}} + x_{\text{encoder1}} \end{aligned} \tag{8}$$

$$x_{\text{Re2}} = \text{Reattention}(\text{LN}(x_{\text{encoder1}})) \tag{9}$$

$$x_{\text{encoder1}} = \text{MLP}(\text{LN}(x_{\text{Re1}} + x)) + x_{\text{Re1}} + x \tag{10}$$

$$x_{\text{Re1}} = \text{Reattention}(\text{LN}(x)) \tag{11}$$

where x is the input of the RCD module, x_{Re1} is the output of the first reattention, x_{encoder1} denotes the output of the first DeepViT encoder, and x_{Re2} is the output of the second reattention.

The RCD module is used to obtain the outputs $y_{L-\text{encoder}}$ and $y_{S-\text{encoder}}$ for the large-scale branch and the small-scale branch, respectively.

The designed RCD module takes the output of the multiscale patch embedding module, i.e., the sequence of tokens containing the class token and patch token information, as input. The RCD module also integrates the class token and patch token information according to the input. The RCD module regenerates attention maps by exchanging information from different attention heads, preserving the diversity of features in different layers. Moreover, through residual connections, information in the intermediate layer is effectively utilized, smoothing the forward and backward propagation of information and reducing the problem of network degradation. The RCD can utilize rich information from different layers.

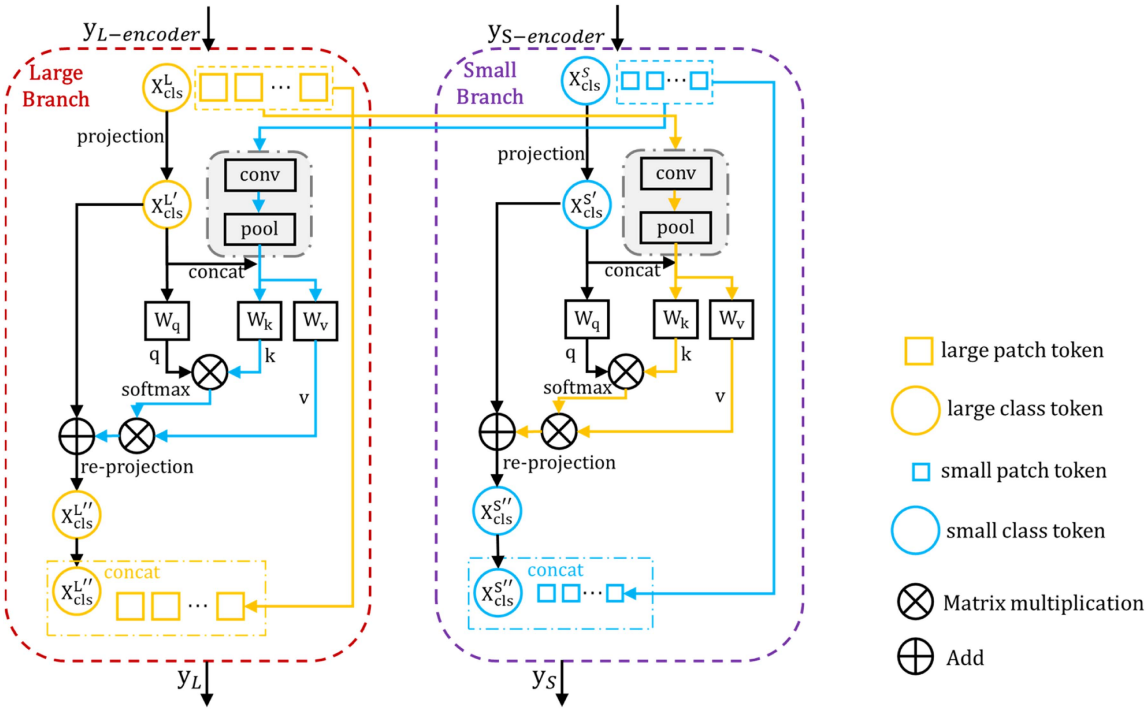


Fig. 5. Illustration of the proposed DBCA module.

C. DBCA Module

Considering that HSIs contain rich information, to better use the complementary information between class tokens and patch tokens and to take into account the multiscale patch information, the DBCA module is proposed to realize the multi-information fusion of HSIs. Its illustration is shown in Fig. 5.

The inputs of the DBCA module are the double-branch outputs of the RCD module, $y_{L-\text{encoder}}$ and $y_{S-\text{encoder}}$. The main interaction core of the DBCA module is to fuse the class token from its own branch with the patch token from another branch (the auxiliary branch) to achieve double-branch information interaction.

Taking the small branch as an example, it mainly consists of three input branches: a small-scale class token branch, a large-scale patch token auxiliary interaction branch, and a small-scale patch token connection branch. The specific interaction processes are shown in (12)–(20).

In the small-scale class token branch, after linear projection, the small-scale class token X_{cls}^{S} becomes $X_{\text{cls}}^{S'}$

$$X_{\text{cls}}^{S\,\prime} = \text{Proj}(X_{\text{cls}}^{S}). \tag{12}$$

In the large-scale patch token auxiliary interaction branch, the large-scale patch token X_{patch}^{L} passes through convolution and pooling operations to obtain x_{pool}. The designed convolution and pooling operations can obtain more nonlinear features and reduce unnecessary information in intermediate processes to reduce computational costs

$$x_{\text{pool}} = \text{Pool}(\text{conv}(X_{\text{patch}}^{L})). \tag{13}$$

Then, x_{pool} and $X_{\text{cls}}^{S'}$ are concatenated, and key(k) and value(v) can be obtained by (15) and (16). The query(q) is obtained through (17) by using the information from $X_{\text{cls}}^{S'}$

$$x_{\text{con}} = \text{concat}\left(x_{\text{pool}}, X_{\text{cls}}^{S\,\prime}\right) \tag{14}$$

$$k = x_{\text{con}} W_k \tag{15}$$

$$v = x_{\text{con}} W_v \tag{16}$$

$$q = X_{\text{cls}}^{S\,\prime} W_q \tag{17}$$

where W_q, W_k, and W_v are the weight parameters.

Next, the attention calculation is performed to obtain the attention result x_{att}, which is subsequently added with $X_{\text{cls}}^{S'}$ and after reverse linear projection, returning to the original dimension to obtain $X_{\text{cls}}^{S''}$, as presented in the following equations:

$$x_{\text{att}} = \left(\text{Softmax}\left(\frac{\text{qk}^{\text{T}}}{\sqrt{\text{d}}}\right)\right)\text{v} \tag{18}$$

$$X_{\text{cls}}{}^{S''} = \text{Reproj}\left(x_{\text{att}} + X_{\text{cls}}^{S\,\prime}\right). \tag{19}$$

In the small-scale patch token connection branch, the small-scale patch token X_{patch}^{S} is concatenated with $X_{\text{cls}}^{S''}$ to obtain the final small-scale branch output y_S, as shown in the following equation:

$$y_S = \text{concat}\left(X_{\text{patch}}^{S}, X_{\text{cls}}{}^{S''}\right). \tag{20}$$

The large branch interaction operation is similar to the above operations; through the large-scale class token branch, the small-scale patch token auxiliary interaction branch, and

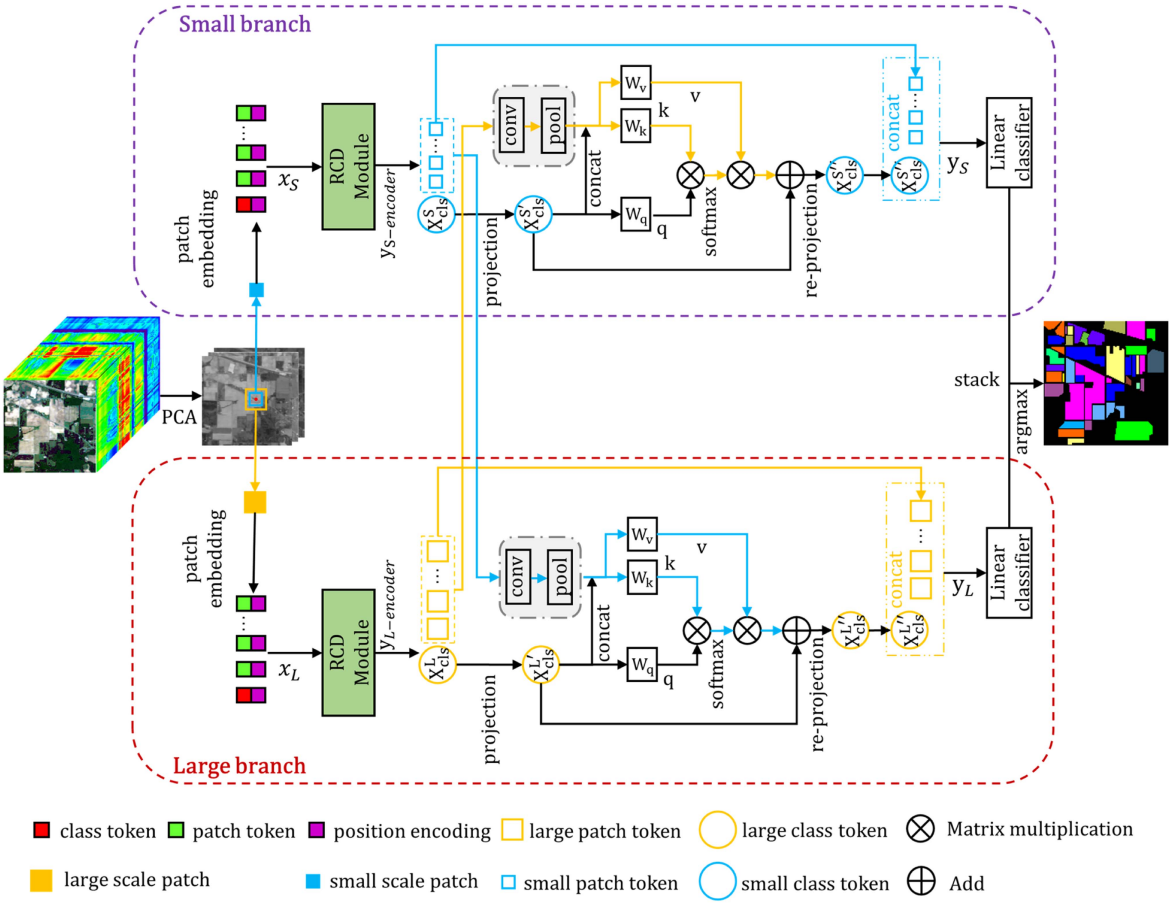


Fig. 6. Flowchart of the proposed CAMFT method.

the large-scale patch token connection branch, the final output of the large-scale branch y_L can be obtained.

The above interaction process of the double branch shows that the DBCA module not only integrates multiscale patch information through two branches, a large-scale branch and a small-scale branch, but also effectively utilizes complementary information between class tokens and patch tokens in the interaction of two branches to obtain more representative features, achieving the multi-information fusion. Moreover, by using the convolution–pooling operation, more nonlinear features are obtained, and the unnecessary information in intermediate processes is reduced, thus reducing computational costs.

In summary, the flowchart of the whole CAMFT method is shown in Fig. 6. It mainly consists of a multiscale patch embedding module for multi-information preprocessing, an RCD module utilizing rich information from different layers, and a DBCA module that obtains more representative multi-information fusion features. First, the HSI is reduced to three dimensions by PCA. In the multiscale patch embedding module, according to the central pixel, patches of two different scales, one at a large scale and one at a small scale, are formed, and two processing branches are built. The output token sequences contain both class tokens and patch tokens. The two processing branches input their respective token sequences into the RCD module and subsequently pass through the DBCA module to obtain representative features via multi-information fusion. Then, the multi-information fusion features are mapped to the sample labeling space through the fully connected layer, after which the two branches are stacked together to achieve a fusion of the classification results. Finally, the classification results are obtained by averaging the stacked data.

The proposed CAMFT method not only considers the interaction information between class tokens and patch tokens but also fully utilizes multiscale patch information while combining the generated multilayer diversity information, ultimately forming more discriminative image features based on multi-information fusion.

IV. Experiments

To validate the effectiveness of the proposed CAMFT method, experiments were conducted on three real hyperspectral datasets and compared with other state-of-the-art HSI classification methods.

A. Datasets

The experiments utilized three public hyperspectral datasets: the Indian Pines (IP), the GF-5 Yancheng (GFYC), and the ZY1-02D Huanghekou (ZYHHK) datasets. For the experiment, 3% of each class was selected as the training set, with a minimum of two samples from each class guaranteed. The remaining samples were used as the test set.

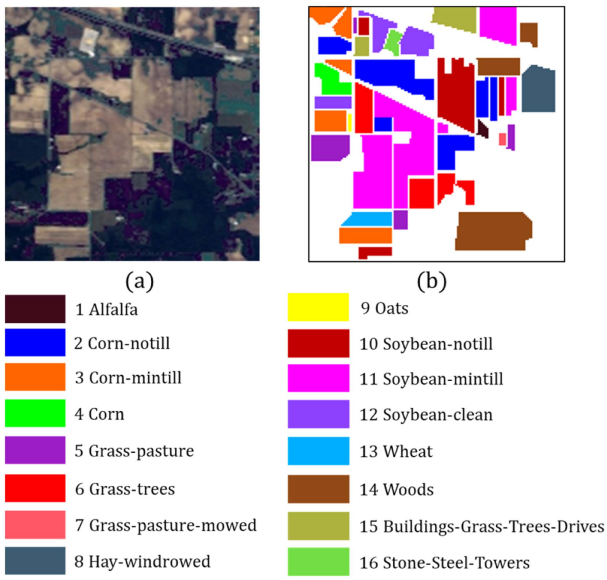


Fig. 7. IP dataset. (a) False-color image. (b) Ground truth map.

TABLE I
SAMPLE INFORMATION FOR THE IP DATASET

No.	Class name	Training samples	Test samples	Total samples
1	Alfalfa	2	44	46
2	Corn-notill	42	1386	1428
3	Corn-mintill	24	806	830
4	Corn	7	230	237
5	Grass-pasture	14	469	483
6	Grass-trees	21	709	730
7	Grass-pasture-mowed	2	26	28
8	Hay-windrowed	14	464	478
9	Oats	2	18	20
10	Soybean-notill	29	943	972
11	Soybean-mintill	73	2382	2455
12	Soybean-clean	17	576	593
13	Wheat	6	199	205
14	Woods	37	1228	1265
15	Buildings-Grass-Trees-Drives	11	375	386
16	Stone-Steel-Towers	2	91	93
	Total	303	9946	10 249

The IP dataset was acquired in 1992 by an AVIRIS sensor over the IP test site in northwestern Indiana with a spatial size of 145 × 145 pixels, a spatial resolution of 20 m/pixel, and 16 land cover classes.[1] A total of 200 spectral bands were retained in the classification experiments. The false-color image and the ground truth map of this dataset are shown in Fig. 7, and the specific sample information is shown in Table I.

The GFYC dataset was acquired by the AHSI sensor aboard on China's GF-5 satellite over the coastal area of Yancheng, Jiangsu, China [58], [59]. The image size is 1175 × 585, with a spatial resolution of 30 m/pixel. A total of 147 bands and 7 common classes are chosen for classification. The false-color image and the ground truth map of this dataset are shown in Fig. 8. The specific sample information is shown in Table II.

[1][Online]. Available: https://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes#Indian_Pines

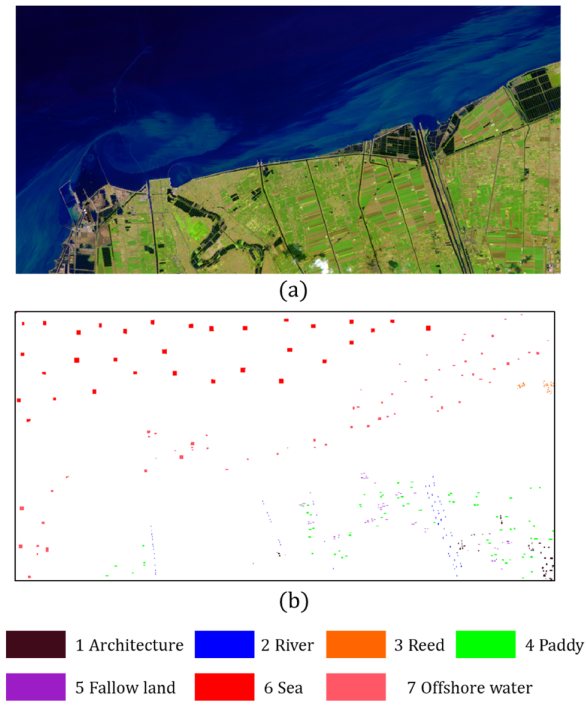


Fig. 8. GFYC dataset. (a) False-color image. (b) Ground truth map.

TABLE II
SAMPLE INFORMATION FOR THE GFYC DATASET

No.	Class name	Training samples	Test samples	Total samples
1	Architecture	10	350	360
2	River	6	211	217
3	Reed	3	129	132
4	Paddy	24	808	832
5	Fallow land	7	227	234
6	Sea	71	2324	2395
7	Offshore water	39	1266	1305
	Total	160	5315	5475

The ZYHHK dataset was acquired by the China ZY1-02D-AHSI sensor over the area around the Yellow River Estuary of China ("Huanghekou" in Chinese) on 29 September 2021 [58], [59]. The image size is 1050 × 1219, with a spatial resolution of 30 m/pixel. The dataset for classification consists of 108 bands and 8 common classes. The false-color image and ground truth map of this dataset are shown in Fig. 9. The specific sample information is shown in Table III.

B. Comparison of Different Classification Methods

To evaluate the performance of the proposed CAMFT method, several state-of-the-art ViT-based HSI classification methods, namely, the SSFTT [42], SwinT [35], SSTN [41], MDvT [49],

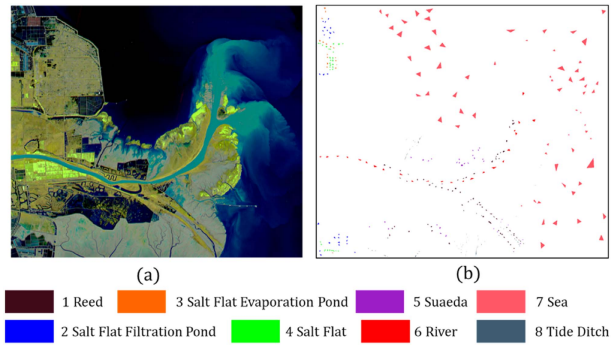


Fig. 9. ZYHHK dataset. (a) False-color image. (b) Ground truth map.

TABLE III
SAMPLE INFORMATION FOR THE ZYHHK DATASET

No.	Class name	Training samples	Test samples	Total samples
1	Reed	31	1026	1057
2	Salt Flat Filtration Pond	15	497	512
3	Salt Flat Evaporation Pond	7	253	260
4	Salt Flat	12	400	412
5	Suaeda	12	406	418
6	River	28	913	941
7	Sea	307	9932	10 239
8	Tide Ditch	7	241	248
	Total	419	13 668	14 087

CS2DT [56], MVAHN [51], IFormer [45], and PD^{2}C [52], were selected for comparison. All the experiments were run for 150 epochs with a learning rate of 0.0005 and a patch size of 13 $\times$ 13 for the comparison methods. The Adam optimizer was used to optimize the network. The classification performance of the networks was measured by three commonly used evaluation metrics [60], [61]: overall accuracy (OA), average accuracy (AA), and Kappa. To ensure fairness, all experiments were repeated ten times and the averaged results were taken.

Figs. 10 –12 show the classification maps for different methods. Taking the IP classification map in Fig. 10 as an example, there are more misclassified regions in the CS2DT and IFormer classification maps. The MVAHN and PD^{2}C methods perform better but still have some obvious misclassified regions. Compared with those of the other methods, the classification map of the proposed CAMFT method has fewer misclassified regions and can yield the optimal classification map. Moreover, according to the GFYC classification map in Fig. 11 and the ZYHHK classification map in Fig. 12, the classification maps of the CAMFT method also have the best performance, with fewer misclassified regions and closer proximity to the ground truth map.

Tables IV–VI list the detailed classification results for the different methods. The data in the tables show that the proposed CAMFT method achieved the highest classification accuracy. Moreover, among the comparison methods, MVAHN and PD^{2}C had better classification results. Table IV lists the classification results for the IP dataset. The proposed CAMFT method achieves the highest classification accuracy, with OA, AA, and Kappa values of 91.61%, 86.7%, and 90.43%, respectively. Among the comparison methods, MVAHN and PD^{2}C perform better, while the CS2DT has the worst performance. The OA, AA, and Kappa of the proposed CAMFT are 3.77%, 1.75%, and 4.32% greater than those of the MVAHN (OA:87.84%, AA: 84.95%, and Kappa: 86.11%, respectively); meanwhile, they achieve 21.56%, 23.1%, and 24.95% greater than those of the CS2DT (OA: 70.05%, AA: 63.6%, and Kappa: 65.48%, respectively). Compared with PD^{2}C (OA: 90.25%, AA: 87.71%, and Kappa: 89.99%, respectively), the proposed CAMFT is 1.36% and 0.44% greater in OA and Kappa.

Similarly, according to the results for the GFYC dataset, as presented in Table V, the proposed CAMFT method achieved 96.65%, 87.98%, and 95.53% for OA, AA, and Kappa, respectively. In addition, the results of the ZYHHK dataset, as shown in Table VI, indicate that the proposed CAMFT method achieved 98.18%, 92.54%, and 96.04% for OA, AA, and Kappa, respectively. On both datasets, the proposed CAMFT method achieved the best classification performance.

Then, based on the classification accuracies for each class in the tables, the proposed CAMFT method can achieve good results in most classes. For the IP dataset, the CAMFT method achieves excellent classification performance for many classes, including classes 3, 10, and 13, with accuracies of 90.31%, 90.95%, and 98.48%, respectively. Parts of classes 3, 10, and 13 are surrounded by a red rectangular box in Fig. 10 and the proposed CAMFT method achieves the best classification result. For the GFYC dataset, the proposed CAMFT method also achieves higher classification accuracy than the other comparison methods for classes 1, 3, 5, 6, and 7. In addition, in the ZYHHK dataset, with accuracies of 98.09%, 96.64%, 89.62%, and 99.71% for classes 1, 2, 3, and 4, respectively, the proposed CAMFT method achieved the highest accuracy among these methods. For a clearer presentation, Fig. 12 also displays a local zoom-in view of the classification maps for parts of classes 2, 3, and 4, demonstrating that the proposed CAMFT method has fewer misclassified pixels in the region and achieves the best classification results.

The reasons why the proposed CAMFT method can achieve optimal classification performance are as follows: First, the RCD module in the CAMFT method integrates the class token and patch token information and regenerates the attention map by exchanging information from different attention heads, preserving the diversity of features in different layers. Moreover, through residual connections, information in the intermediate layer is effectively utilized, smoothing the forward and backward propagation of information and reducing the problem of network degradation. Second, the DBCA module in the CAMFT method not only integrates multiscale patch information through two branches—a large-scale branch and a small-scale branch—but also effectively utilizes complementary information between class tokens and patch tokens in the interaction of two branches to obtain additional representative features. Meanwhile, by using the convolution–pooling operation in the DBCA module, more nonlinear features are obtained, and the unnecessary information in intermediate processes is reduced. Finally, the classification

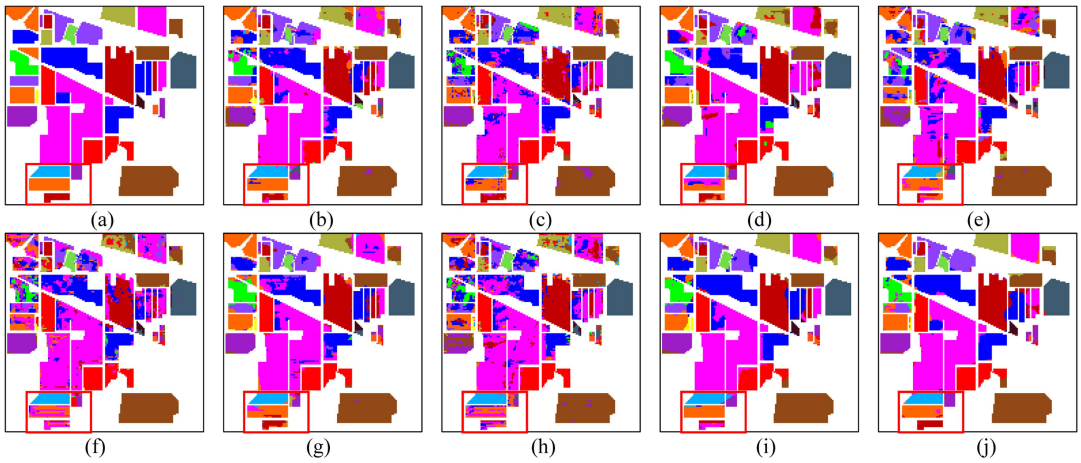


Fig. 10. Classification maps of different methods for the IP dataset. (a) Ground truth map. (b) SSFTT (OA: 85.64%). (c) SwinT (OA: 80.43%). (d) SSTN (OA: 81.77%). (e) MDvT (OA: 74.92%). (f) CS2DT (OA: 73.83%). (g) MVAHN (OA: 89.06%). (h) IFormer (OA: 72.85%). (i) PD^{2}C (OA: 91.69%). (j) Proposed (OA: 93.61%).

TABLE IV
CLASSIFICATION RESULTS OF DIFFERENT METHODS FOR THE IP DATASET

Class	SSFTT	SwinT	SSTN	MDvT	CS2DT	MVAHN	IFormer	PD^{2}C	Proposed
1	56.29±23.27	47.05±10.66	25.46±17.74	50.23±19.32	33.86±14.14	65.91±10.56	47.27±15.24	**94.09±3.96**	77.14±8.6
2	75.41±3.03	63.05±4.72	54.62±2.8	61.24±5.9	57.57±5.52	67.89±5.92	45.06±3.53	86.63±2.66	**87.15±2.96**
3	87.41±3.18	75.12±5.21	74.13±3.54	67.49±8.37	47.93±11.83	89.77±2.51	54.63±11.05	87.98±1.75	**90.31±4.07**
4	85.09±2.82	46.74±6.77	87.09±4.17	40.61±15.46	35.29±7.99	**92.6±3.8**	28.1±8.42	86.87±5.35	86.71±7.57
5	71.82±7.35	68.38±9.1	78.27±4.6	62.94±10.77	71.27±11.77	77.11±3.39	53.82±8.7	82.85±1.61	**84.26±4.22**
6	92.0±3.1	87.38±4.06	**94.85±1.01**	88.5±5.07	89.5±7.32	97.32±0.92	95.81±2.64	89.24±1.85	92.61±2.7
7	72.55±9.26	50.39±12.92	**95.77±3.63**	60.39±17.37	53.46±15.99	89.23±10.15	53.85±15.19	92.31±5.96	60.0±14.81
8	98.55±1.19	98.93±1.25	98.8±0.78	98.69±1.7	98.86±1.15	96.86±6.18	98.17±2.94	**100.0**	98.39±1.84
9	74.14±26.03	26.67±12.37	57.22±24.6	41.11±19.12	23.89±7.05	75.56±24.49	57.78±9.03	**93.33±6.48**	74.38±19.66
10	86.39±1.94	77.07±5.05	86.92±2.12	77.41±6.94	54.17±9.09	86.18±3.61	76.63±2.51	86.84±2.49	**90.95±1.93**
11	91.72±2.23	84.23±4.79	92.37±1.69	84.32±4.79	79.43±3.88	93.83±1.04	83.96±2.75	**97.03±0.38**	95.02±1.5
12	73.27±5.66	50.7±6.24	62.71±7.93	55.7±7.81	38.58±6.87	78.49±6.77	40.81±4.92	77.96±0.92	**82.29±3.27**
13	93.25±5.59	90.66±4.91	98.03±0.8	84.9±5.49	94.39±4.15	98.28±0.85	97.25±2.65	95.21±2.65	**98.48±2.6**
14	92.91±1.8	93.51±3.89	93.99±1.4	92.16±3.18	93.5±2.44	96.15±2.21	90.67±3.18	96.36±0.76	**96.43±1.49**
15	85.28±5.17	89.92±3.92	78.4±3.86	73.84±8.42	52.0±7.63	86.87±4.67	64.0±2.93	93.95±1.54	**95.51±3.3**
16	65.54±10.13	61.12±13.61	75.39±13.12	83.26±9.27	**93.82±5.35**	80.45±14.9	91.91±3.65	86.74±3.21	77.53±9.83
OA(%)	85.19±1.24	78.14±1.84	80.47±0.95	76.66±2.15	70.05±2.72	87.84±1.24	71.89±1.58	90.25±0.33	**91.61±1.07**
AA(%)	81.47±3.33	69.43±1.98	78.37±2.05	70.17±4.16	63.6±3.13	84.95±2.37	67.48±1.6	**87.71±0.55**	86.7±2.47
Kappa(%)	83.47±0.96	74.98±2.03	79.87±1.11	73.31±2.46	65.48±3.31	86.11±1.44	67.61±1.73	89.99±0.38	**90.43±1.22**

The best results are shown in bold.

task is accomplished based on the more representative multi-information fusion features obtained by fusing the classification results of two branches.

C. Parameter Analysis

This section discusses and analyzes several important parameters of the proposed CAMFT method.

1) Effects of Training Sample Percentages: To evaluate the effects of the training sample percentage on the classification performance, in the three datasets, 1%, 2%, 3%, 4%, and 5% of the samples in each class were selected as training samples, respectively. The classification results are presented in Fig. 13. As the percentage of training samples gradually increases, the overall classification accuracy of all methods improves. This is because as the number of training samples increases, the amount of information input into the classification methods also increases, allowing the classification model to better learn the implicit features in the data and improve the classification accuracy.

The red folded line represents the proposed CAMFT method, which consistently achieves the highest accuracy with different percentages of training samples. This advantage is particularly prominent when the training sample size is small, and the proposed CAMFT method still has excellent performance. For example, when the training percentage is only 1% for the IP dataset, the OA of the proposed CAMFT reaches 80.07%, which is 3.21% higher than that of the PD^{2}C (76.86%),

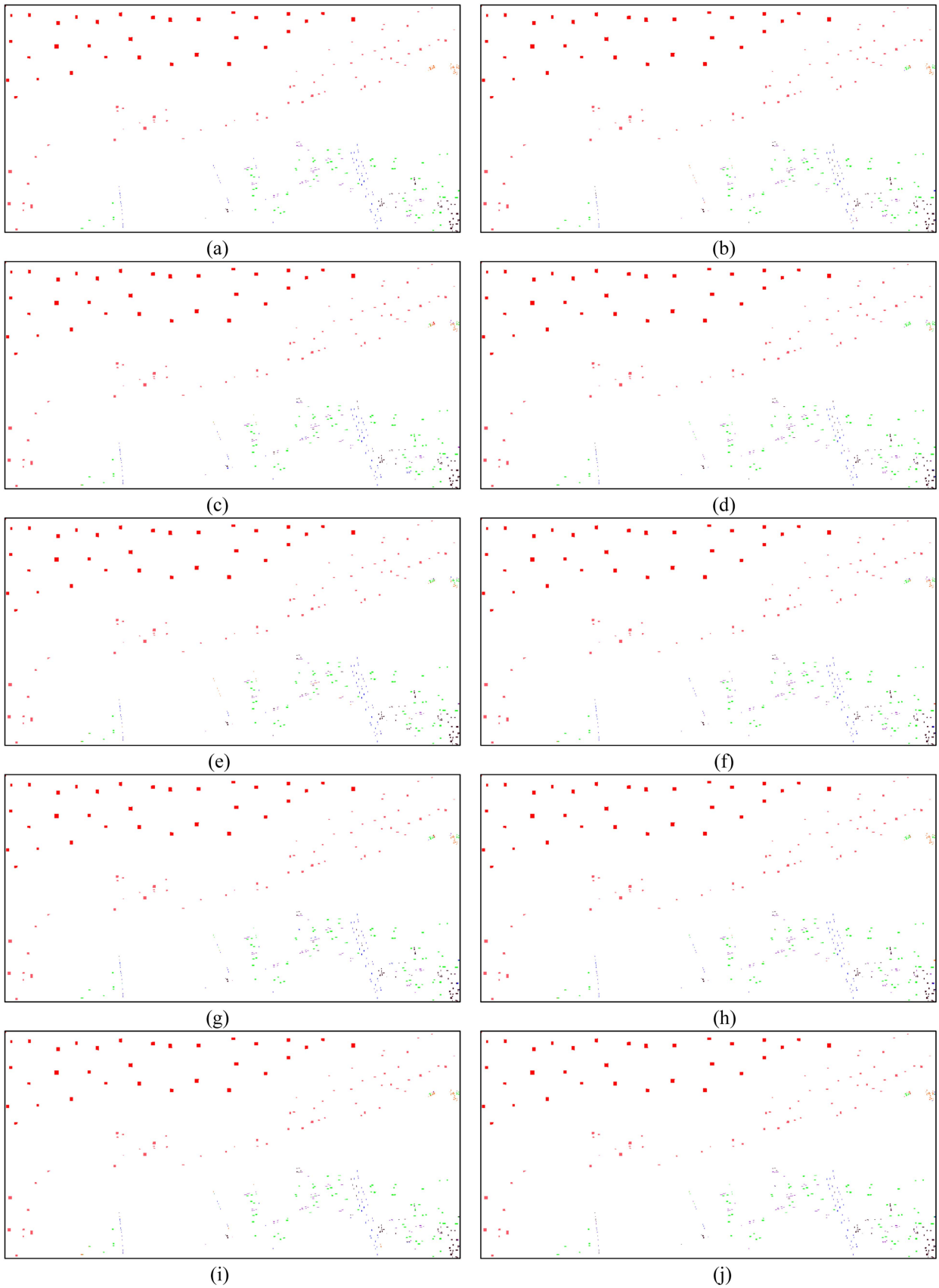


Fig. 11. Classification maps of different methods for the GFYC dataset. (a) Ground truth map. (b) SSFTT (OA: 95.97%). (c) SwinT (OA: 95.47%). (d) SSTN (OA: 95.43%). (e) MDvT (OA: 95.17%). (f) CS2DT (OA: 96.65%). (g) MVAHN (OA: 96.19%). (h) IFormer (OA: 96.69%). (i) PD^{2}C (OA: 95.74%). (j) Proposed (OA: 97.21%).

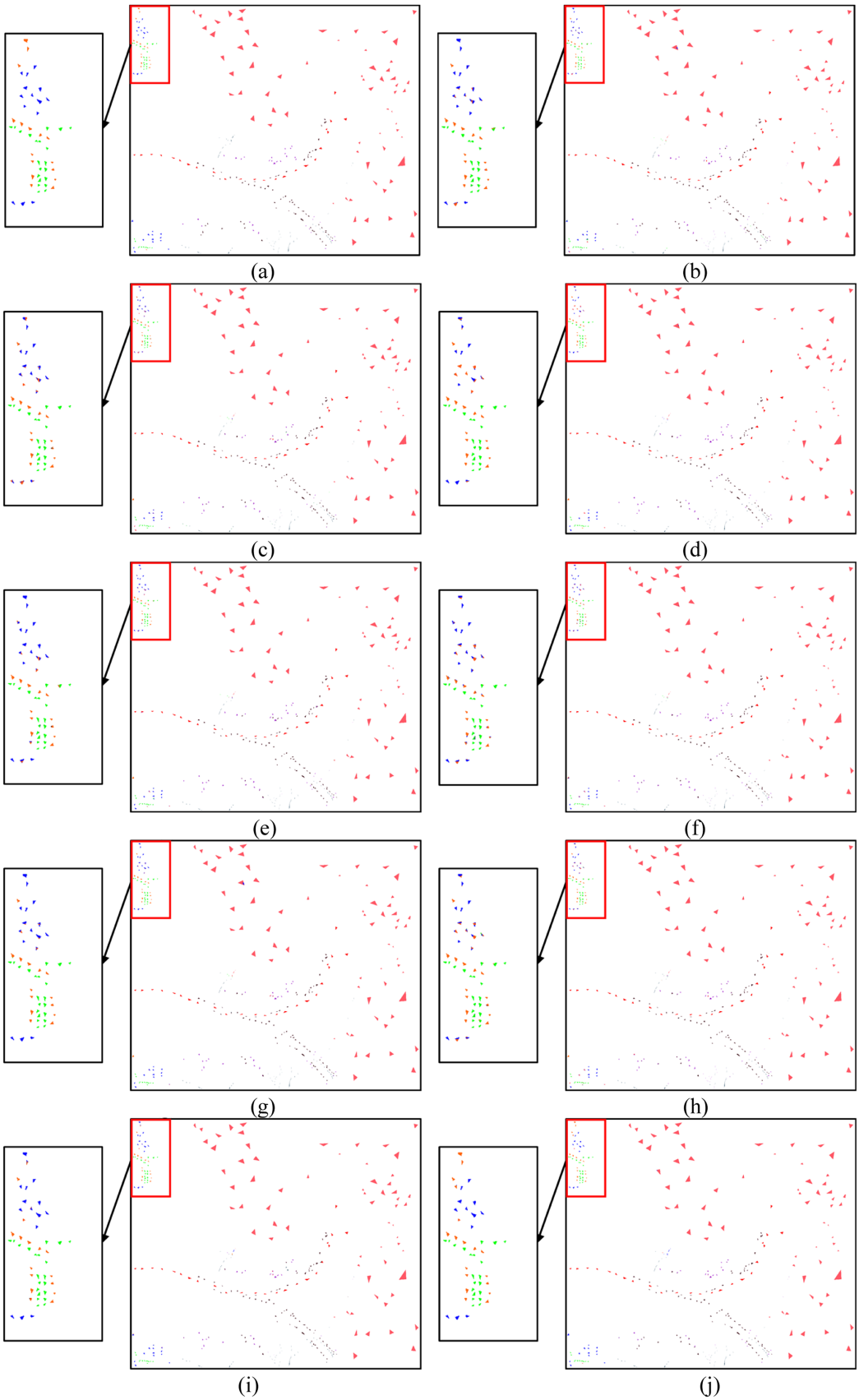


Fig. 12. Classification maps of different methods for the ZYHHK dataset. (a) Ground truth map. (b) SSFTT (OA: 96.94%). (c) SwinT (OA: 96.47%). (d) SSTN (OA: 96.95%). (e) MDvT (OA: 96.9%). (f) CS2DT (OA: 96.88%). (g) MVAHN (OA: 97.31%). (h) IFormer (OA: 96.52%). (i) PD^{2}C (OA: 97.32%). (j) Proposed (OA: 98.72%).

TABLE V
CLASSIFICATION RESULTS OF DIFFERENT METHODS FOR THE GFYC DATASET

Class	SSFTT	SwinT	SSTN	MDvT	CS2DT	MVAHN	IFormer	PD^{2}C	Proposed
1	89.74±8.46	86.03±2.54	88.64±6.2	90.44±4.75	90.96±3.57	86.44±6.43	87.71±2.22	83.13±6.07	**96.47±1.77**
2	62.45±13.51	81.06±5.54	59.17±15.28	75.02±14.49	67.5±18.3	67.97±18.21	**94.04±4.06**	77.88±9.22	77.28±3.38
3	47.01±19.6	28.03±15.8	28.41±13.46	30.53±16.29	46.46±12.32	46.14±9.85	45.04±17.7	56.22±7.95	**57.2±7.96**
4	96.17±5.1	99.17±0.82	**99.36±0.95**	97.79±1.14	97.45±1.64	96.98±2.29	97.75±1.08	96.4±1.52	98.06±0.52
5	76.43±11.8	64.82±13.02	76.28±15.88	59.1±10.87	77.86±12.84	82.18±7.7	85.0±5.86	68.49±15.05	**86.83±8.29**
6	100.0	100.0	100.0	99.93±0.09	100.0	100.0	100.0	99.93±0.1	**100.0**
7	100.0	100.0	100.0	100.0	100.0	99.96±0.11	100.0	100.0	**100.0**
OA(%)	94.97±0.81	94.97±0.36	94.8±0.79	94.59±1.06	95.5±0.67	95.31±0.6	96.15±0.31	95.04±1.03	**96.65±0.46**
AA(%)	81.68±3.73	79.87±2.15	78.84±3.23	78.97±4.45	82.89±2.79	82.81±2.4	87.08±2.21	83.15±2.66	**87.98±1.85**
Kappa(%)	93.01±1.12	92.97±0.51	92.75±1.11	92.48±1.48	93.73±0.93	93.47±0.84	95.34±0.44	93.1±1.42	**95.53±0.64**

The best results are shown in bold.

TABLE VI
CLASSIFICATION RESULTS OF DIFFERENT METHODS FOR THE ZYHHK DATASET

Class	SSFTT	SwinT	SSTN	MDvT	CS2DT	MVAHN	IFormer	PD^{2}C	Proposed
1	88.8±5.14	89.2±3.12	90.14±1.82	89.81±1.49	91.85±3.15	90.01±1.72	92.01±1.24	87.48±4.66	**98.09±2.23**
2	70.24±5.76	69.77±6.83	54.94±12.8	69.98±6.29	63.96±8.88	83.38±5.98	65.76±5.23	92.19±5.18	**96.64±1.09**
3	79.64±1.59	64.19±11.62	78.65±4.48	73.57±3.85	67.85±8.75	82.01±6.38	79.92±7.67	81.81±8.07	**89.62±13.49**
4	95.27±4.49	98.52±1.84	99.61±0.63	90.79±10.39	95.58±7.03	99.65±0.44	91.8±798	97.35±6.42	**99.71±0.73**
5	82.04±6.16	73.76±10.85	81.17±5.83	79.15±8.53	84.0±6.24	**84.74±4.14**	73.92±11.31	73.76±5.61	72.06±6.78
6	99.9±0.2	98.78±2.34	99.74±0.42	99.27±0.9	99.89±0.29	**100.0**	99.96±0.1	99.39±1.08	99.65±0.43
7	99.92±0.09	99.86±0.12	99.87±0.14	**99.94±0.09**	99.93±0.08	99.65±0.19	99.67±0.44	99.92±0.11	99.68±0.19
8	73.82±5.76	70.68±7.3	**85.4±0.94**	78.74±8.19	83.15±4.5	76.81±8.12	84.19±3.55	75.4±19.8	84.92±3.57
OA(%)	96.5±0.38	95.91±0.62	96.29±0.5	96.31±0.38	96.53±0.48	96.79±0.45	96.25±0.56	97.06±0.43	**98.18±0.35**
AA(%)	86.2±1.09	83.1±2.67	86.19±1.98	85.16±1.7	85.77±2.03	89.53±2.01	85.9±2.78	88.42±2.67	**92.54±2.22**
Kappa(%)	92.26±0.84	90.94±1.39	91.71±1.18	91.79±0.86	92.24±1.11	93.84±0.59	91.68±1.28	93.56±0.94	**96.04±0.77**

The best results are shown in bold.

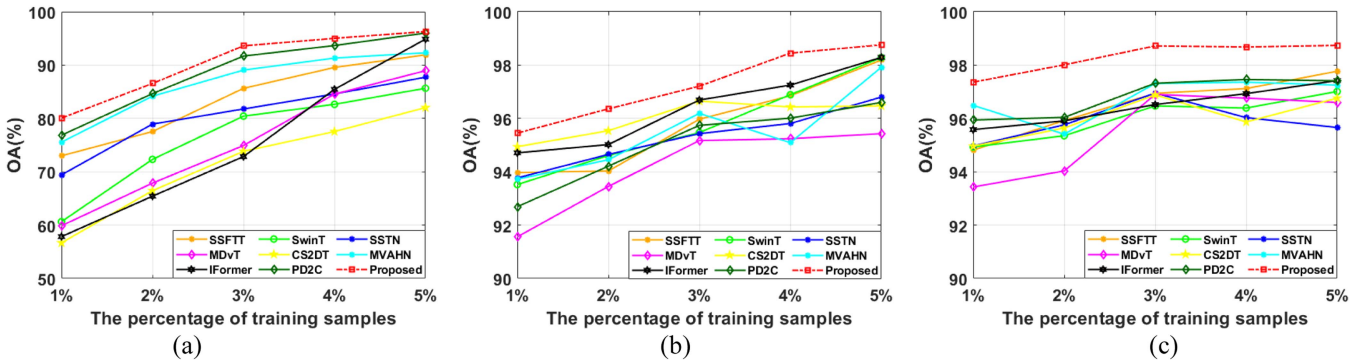


Fig. 13. OAs of different methods under different percentages of training samples. (a) IP dataset. (b) GFYC dataset. (c) ZYHHK dataset.

which has the highest accuracy among the comparison methods, and 23.39% higher than that of the CS2DT method (56.68%).

2) Influence of the Number of DeepViT Encoders: This section discusses the influence of the number of DeepViT encoders in the RCD module. Fig. 14 displays the OA, AA, and Kappa values for the three datasets with varying numbers of DeepViT encoders: 0 (normal ViT encoder), 1, 2, and 3 (residual connections are also added between intermediate layers when the number of DeepViT encoders are 2 and 3). The experimental

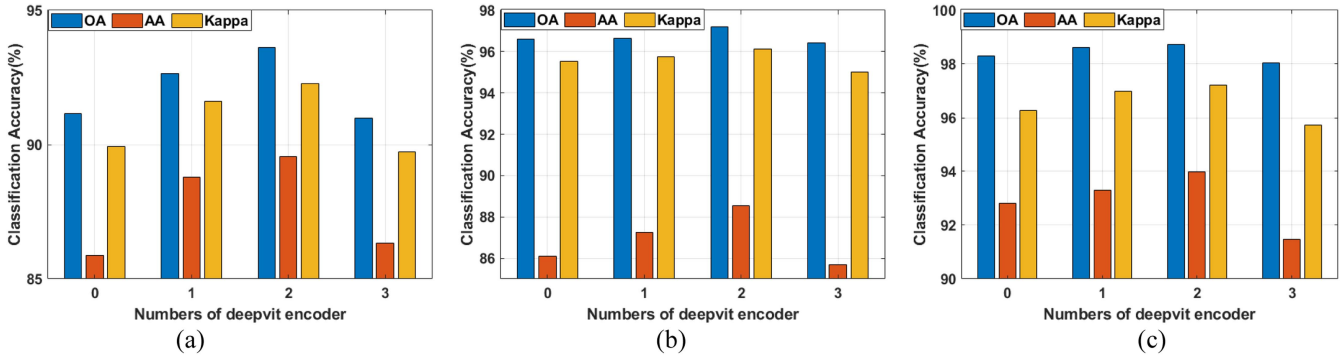


Fig. 14. Classification accuracies for different numbers of DeepViT encoder. (a) IP dataset. (b) GFYC dataset. (c) ZYHHK dataset.

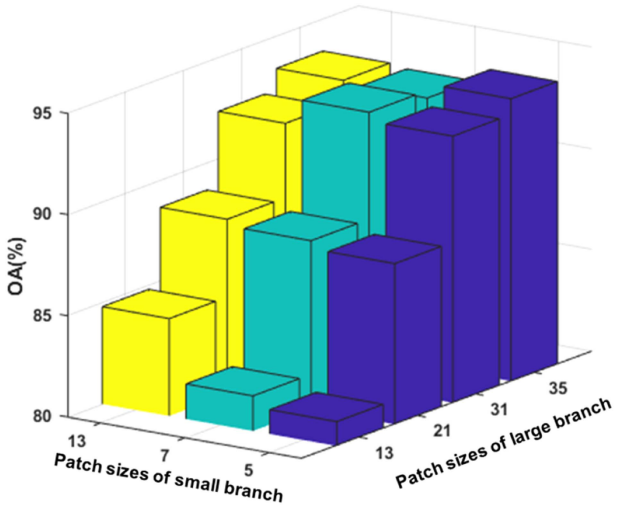


Fig. 15. OAs of different multiscale patch sizes.

results demonstrate that, when the number of DeepViT encoders is 0 (only using the normal ViT encoder), the classification accuracy is relatively low. By adding 1 DeepViT encoder, the classification accuracy significantly improves, indicating the effectiveness of reattention in preserving the diversity of features in different layers. As the number of DeepViT encoders increases, the classification accuracy gradually improves. However, too many encoders can lead to a decrease in classification accuracy due to the overfitting problem. When the number of DeepViT encoders is 2, the method achieves the highest classification accuracy by preserving the diversity of features in different layers, and through residual connections, it effectively utilizes the information in the intermediate layer, smoothing the forward and backward propagation of information and reducing network degradation. Based on these analyses, a proper number of DeepViT encoders can improve the classification accuracy, and the number of DeepViT encoders is ultimately set to 2 in the proposed CAMFT method.

3) Impacts of Multiscale Patch Sizes: This section discusses the impacts of multiscale patch sizes in the proposed CAMFT method, specifically the combination of patch sizes for large-scale branch and small-scale branch. The patch sizes for the large branch were set to 13, 21, 31, and 35, while the patch sizes for the small branch were set to 5, 7, and 13. The experimental results for the IP dataset are shown in Fig. 15. When a multiscale patch is combined with a large and small branch size, it can be seen that when the gap between the patch sizes of the two branches is large, complementary multiscale information is advantageous, and the classification performance is better. However, patches that are too large will also include some pixels that are unrelated to central pixels to reduce classification effects. From Fig. 15, it can be observed that the classification accuracy is higher when the multiscale patch size combination is set to (7, 31) (the patch size of the small branch is 7 and the patch size of the large branch is 31) and (5, 35) because the interaction of large and small branches by these multiscale patch size combinations can achieve better complementarity information. Moreover, considering the actual computational complexity, the proposed CAMFT method sets the multiscale patch size combination to (7, 31).

TABLE VII
ABLATION STUDY ON CONVOLUTION AND POOLING OPERATIONS FOR THE IP DATASET ($\times$ REPRESENTS NO SUCH OPERATION, AND $\surd$ REPRESENTS WITH SUCH OPERATION)

Convolution	Pooling	OA(%)	AA(%)	Kappa(%)
$\times$	$\times$	89.73	85.11	88.3
$\times$	$\surd$	92.04	84.65	90.92
$\surd$	$\times$	86.95	81.54	85.08
$\surd$	$\surd$	**93.61**	**89.55**	**92.27**

The best results are shown in bold.

D. Ablation Study

This section discusses the effectiveness of the DBCA module.

1) Effectiveness of Convolution and Pooling Operations: This experiment aimed to verify the effectiveness of the convolution and pooling operations designed for patch tokens in the DBCA module. The corresponding ablation experimental results for the IP dataset are listed in Table VII. The results show that the classification accuracy is low when both convolution and pooling operations are absent, with an OA of 89.73%. However, when pooling operations are added, the OA improves to 92.04%. Conversely, when there is only convolution and no pooling, the accuracy decreases (OA:

TABLE VIII
ABLATION STUDY ON THE FUSION OF CLASSIFICATION RESULTS FOR THE IP DATASET

Method	OA(%)	AA(%)	Kappa(%)
Only small branch output	93.18	89.26	92.24
Only large branch output	88.62	84.83	87.03
Proposed	**93.61**	**89.55**	**92.27**

The best results are shown in bold.

TABLE IX
ABLATION STUDY ON THE PATCH TOKEN INTERACTION BETWEEN TWO BRANCHES FOR THE IP DATASET

Method	OA(%)	AA(%)	Kappa(%)
Without patch token interactions	92.42	86.24	91.35
Proposed	**93.61**	**89.55**	**92.27**

The best results are shown in bold.

TABLE X
ABLATION STUDY ON THE CLASS TOKEN FOR THE IP DATASET

Method	OA(%)	AA(%)	Kappa(%)
Without class token	86.22	74.13	84.24
Proposed	**93.61**	**89.55**	**92.27**

The best results are shown in bold.

TABLE XI
ABLATION STUDY ON MULTI-INFORMATION FOR THE IP DATASET

Method	OA(%)	AA(%)	Kappa(%)
Only small branch	71.1	63.43	66.77
Only large branch	82.95	74.16	80.54
Proposed	**93.61**	**89.55**	**92.27**

The best results are shown in bold.

86.95%). The proposed CAMFT method, which combines convolution and pooling operations, achieves the highest accuracy, with an OA of 93.61%. The reasons for the above phenomenon are as follows: when the convolution operation is used alone, important information may not be effectively retained from the original features, resulting in information loss. However, by adding a pooling operation and selecting an appropriate pooling size, the information on neighboring positions can be aggregated, leading to an improved classification result while retaining important information. In the DBCA module, a combination of convolution and pooling operations is used, additional nonlinear features are obtained, and unnecessary information from intermediate processes is reduced, thus reducing computational costs.

2) Effectiveness of Classification Result Fusion: This experiment was conducted to verify the effectiveness of fusing the final double-branch classification results with respect to only a single-branch output. The specific results are presented in Table VIII. The results show that, when only the small branch features are output for classification, the OA is 93.18%, while when only the large branch features are output for classification, the OA is 88.62%. By fusing the double-branch classification results, the proposed CAMFT method can achieve an OA of 93.61%. This demonstrates that the fusion stage of the classification results can combine the generated multiscale information of the double branch to improve the classification performance.

3) Effectiveness of the Patch Token Interaction Between Two Branches: This experiment is used to verify the effectiveness of patch token interaction between two branches, compared with no patch token interaction between two branches. The results are shown in Table IX. Table IX reveals that, if there is no patch token interaction between the two branches, the OA decreases from 93.61% to 92.42%, indicating that using the patch token from another auxiliary branch can achieve double-branch information interaction and provide complementary information. Additionally, the multiscale patch information is considered, and the output multi-information fusion features are more representative.

4) Effectiveness of the Class Token: This experiment aimed to verify the effectiveness of the proposed CAMFT method on class tokens. A comparison experiment is designed without the addition of any class token in the whole network, and the attention operation in the DBCA module is then changed to ordinary multihead attention, i.e., q, k, and v are all from the patch token information. Table X displays the experimental results. When the class token is removed, the OA drops to 86.22%, which is 7.39% lower than that of the proposed CAMFT (93.61%). By involving the class token in the entire transformer structure and attention operation, the input image can be comprehended from a global perspective, allowing us to better capture important features. Additionally, the proposed CAMFT method includes cross-branch interactions between the class tokens and the patch tokens, which enhances information complementarity.

5) Effectiveness of Multi-Information: Based on the previous ablation experiment (4), the whole network is modified to have only one branch to verify the effectiveness of the multi-information process. Here, the single branch has no class token and no patch token interaction. As shown in Table XI, the OA is only 71.1% if there is only a small branch. The reason for the poor classification performance is that the small patch size is only 7, which cannot capture enough global context information when processing HSIs. In contrast, when larger branches are available, the size of the initial patch is effectively expanded, resulting in an increase in OA to 82.95%. This indicates that a larger patch size can better capture global information and contextual dependencies. The above classification accuracy is relatively low because only single-scale information is adopted, without utilizing rich multi-information in HSIs. In contrast, based on the rich multi-information, the proposed CAMFT method has the highest OA of 93.61% because it considers large and small branches to combine multiscale patch information while utilizing the complementary information of class tokens and patch tokens, thus enhancing the model's ability to perceive important features in HSIs.

The effectiveness of the proposed DBCA module was verified through the above ablation experiments. The DBCA module not only integrates multiscale patch information through two branches—a large-scale branch and a small-scale branch—but also effectively utilizes complementary information between class tokens and patch tokens in the interaction of two branches to obtain more representative features. Moreover, by using the

TABLE XII
MODEL SIZES OF DIFFERENT METHODS FOR THE IP DATASET

Model	FLOPs(G)	Params(M)
SSFTT	10.05	0.19
SwinT	0.06	0.84
SSTN	0.01	0.02
MDvT	1.36	7.74
CS2DT	0.22	1.98
MVAHN	0.01	0.12
IFormer	0.16	2.03
PD^2C	0.04	4.95
Proposed	1.34	0.87

convolution–pooling operation, more nonlinear features are obtained, and the unnecessary information in intermediate processes is reduced, thus reducing computational costs.

E. Comparison of Model Size

The model size is measured by using FLOPs and parameters (Params). Table XII lists the model sizes of different methods on the IP dataset. It can be seen that the model size of the proposed CAMFT method is relatively large in the comparison methods, but not the largest. The FLOPs of the proposed CAMFT (1.34G) are lower than those of SSFTT (10.05G) and MDvT (1.36G), and the Params of the proposed CAMFT (0.87M) are lower than those of MDvT (7.74M), PD^2C (4.95M), IFormer (2.03M), and CS2DT (1.98M). Therefore, to some extent, the classification performance improvement of the proposed CAMFT method comes at the cost of increasing the model size.

V. CONCLUSION

Existing transformer-based HSI classification methods generally involve limited information utilization and insufficient consideration of the rich information in HSIs. To address the above issues, the CAMFT method was proposed for HSIs; this method mainly includes the multiscale patch embedding, RCD and DBCA modules. The CAMFT method not only considers the interaction information between class tokens and patch tokens but also fully utilizes multiscale patch information, while combining the generated multilayer diversity information, ultimately forming more discriminative image features based on multi-information fusion. Numerous experiments demonstrate that, compared with other state-of-the-art classification methods, the proposed CAMFT method achieves optimal classification performance, especially when with a small training sample size, it still has excellent performance.

The classification performance improvement of the proposed CAMFT method comes at the cost of increasing the model size. Additionally, the multiscale patch size parameters were manually selected. Therefore, in future research, some lightweight networks will be explored and intelligent methods can be considered for multiscale patch size selection.

ACKNOWLEDGMENT

The authors would like to thank Prof. Jiangtao Peng and Prof. Weiwei Sun for providing the GFYC and ZYHHK data, and the editors and reviewers for providing valuable comments and suggestions that significantly helped us improve this article.

REFERENCES

[1] N. Aburaed, M. Q. Alkhatib, S. Marshall, J. Zabalza, and H. Al Ahmad, "A review of spatial enhancement of hyperspectral remote sensing imaging techniques," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 2275–2300, Feb. 2023.
[2] M. Pal, "Deep learning algorithms for hyperspectral remote sensing classifications: An applied review," *Int. J. Remote Sens.*, vol. 45, no. 2, pp. 451–491, 2024.
[3] N. Wambugu et al., "Hyperspectral image classification on insufficient-sample and feature learning using deep neural networks: A review," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 105, 2021, Art. no. 102603.
[4] H. Shirmard, E. Farahbakhsh, R. D. Müller, and R. Chandra, "A review of machine learning in processing remote sensing data for mineral exploration," *Remote Sens. Environ.*, vol. 268, 2022, Art. no. 112750.
[5] V. Dremin et al., "Skin complications of diabetes mellitus revealed by polarized hyperspectral imaging and machine learning," *IEEE Trans. Med. Imag.*, vol. 40, no. 4, pp. 1207–1216, Apr. 2021.
[6] M. Awad, "Sea water chlorophyll-a estimation using hyperspectral images and supervised artificial neural network," *Ecol. Inform.*, vol. 24, pp. 60–68, 2014.
[7] M. Ahmad et al., "Hyperspectral image classification—Traditional to deep models: A survey for future prospects," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 968–999, Feb. 2022.
[8] N. Audebert, B. Le Saux, and S. Lefevre, "Deep learning for classification of hyperspectral data: A comparative review," *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 2, pp. 159–173, Jun. 2019.
[9] B. Hosseiny, M. Mahdianpari, M. Hemati, A. Radman, F. Mohammadimanesh, and J. Chanussot, "Beyond supervised learning in remote sensing: A systematic review of deep learning approaches," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 1035–1052, Jan. 2024.
[10] P. Ranjan and A. Girdhar, "A comprehensive systematic review of deep learning methods for hyperspectral images classification," *Int. J. Remote Sens.*, vol. 43, no. 17, pp. 6221–6306, 2022.
[11] K. R. Hasan, A. B. Tuli, M. A. M. Khan, S. H. Kee, M. A. Samad, and A.-A. Nahid, "Deep-learning-based semantic segmentation for remote sensing: A bibliometric literature review," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 1390–1418, Jan. 2024.
[12] G. Cheng, X. Xie, J. Han, L. Guo, and G.-S. Xia, "Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 3735–3756, Jun. 2020.
[13] L. Jiao and J. Zhao, "A survey on the new generation of deep learning in image processing," *IEEE Access*, vol. 7, pp. 172231–172263, 2019.
[14] H. Wu, Y. Liu, and J. Wang, "Review of text classification methods on deep learning," *Comput., Mater. Continua*, vol. 63, no. 3, pp. 1309–1321, 2020.
[15] L. Huang, B. Jiang, S. Lv, Y. Liu, and Y. Fu, "Deep-learning-based semantic segmentation of remote sensing images: A survey," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 8370–8396, Apr. 2024.
[16] X. Tang, Q. Ma, X. Zhang, F. Liu, J. Ma, and L. Jiao, "Attention consistent network for remote sensing scene classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 2030–2045, Jan. 2021.
[17] Lefei Zhang and L. Zhang, "Artificial intelligence for remote sensing data analysis: A review of challenges and opportunities," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 2, pp. 270–294, Jun. 2022.
[18] L. Lv and L. Zhang, "Advancing data-efficient exploitation for semi-supervised remote sensing images semantic segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, Apr. 2024, Art. no. 5619213.
[19] F. Ullah, I. Ullah, R. U. Khan, S. Khan, K. Khan, and G. Pau, "Conventional to deep ensemble methods for hyperspectral image classification: A comprehensive survey," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 3878–3916, Jan. 2024.

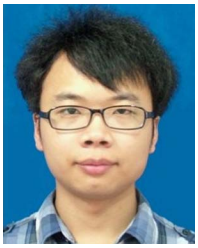
[20] F. Xu, G. Zhang, C. Song, H. Wang, and S. Mei, "Multiscale and cross-level attention learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, Jan. 2023, Art. no. 5501615.
[21] W. Zhao and S. Du, "Spectral–spatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4544–4554, Aug. 2016.
[22] Z. Zhong, J. Li, Z. Luo, and M. Chapman, "Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 847–858, Feb. 2018.
[23] W. Ma, Q. Yang, Y. Wu, W. Zhao, and X. Zhang, "Double-branch multi-attention mechanism network for hyperspectral image classification," *Remote Sens.*, vol. 11, no. 11, 2019, Art. no. 1307.
[24] S. K. Roy, G. Krishna, S. R. Dubey, and B. B. Chaudhuri, "HybridSN: Exploring 3-D–2-D CNN feature hierarchy for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 277–281, Feb. 2020.
[25] M. Zhu, L. Jiao, F. Liu, S. Yang, and J. Wang, "Residual spectral–spatial attention network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 449–462, Jan. 2020.
[26] J. Zhang, Z. Meng, F. Zhao, H. Liu, and Z. Chang, "Convolution transformer mixer for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, Sep. 2022, Art. no. 6014205.
[27] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," *in Proc. 9th Int. Conf. Learn. Representations*, May 2021.
[28] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers and distillation through attention," in *Proc. 38th Int. Conf. Mach. Learn.*, 2021, vol. 139, pp. 10347–10357.
[29] W. Wang et al., "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 548–558.
[30] L. Yuan et al., "Tokens-to-token ViT: Training vision transformers from scratch on ImageNet," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 538–547.
[31] S. Mehta and M. Rastegari, "Mobilevit: Light-weight, general-purpose, and mobile-friendly vision transformer," *in Proc. 10th Int. Conf. Learn. Representations*, Apr. 2022.
[32] D. Zhou et al., "Deepvit: Towards deeper vision transformer," 2021, *arXiv:2103.11886*.
[33] C.-F. R. Chen, Q. Fan, and R. Panda, "CrossViT: Cross-attention multi-scale vision transformer for image classification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 347–356.
[34] H. Fan et al., "Multiscale vision transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 6804–6815.
[35] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 9992–10002.
[36] Z. Peng et al., "Conformer: Local features coupling global representations for visual recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 357–366.
[37] A. Srinivas, T.-Y. Lin, N. Parmar, J. Shlens, P. Abbeel, and A. Vaswani, "Bottleneck transformers for visual recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 16514–16524.
[38] H. Wu et al., "CvT: Introducing convolutions to vision transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 22–31.
[39] D. Hong et al., "SpectralFormer: Rethinking hyperspectral image classification with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Feb. 2022, Art. no. 5518615.
[40] X. He, Y. Chen, and Z. Lin, "Spatial-spectral transformer for hyperspectral image classification," *Remote Sens.*, vol. 13, no. 3, 2021, Art. no. 498.
[41] Z. Zhong, Y. Li, L. Ma, J. Li, and W.-S. Zheng, "Spectral–spatial transformer network for hyperspectral image classification: A factorized architecture search framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Oct. 2021, Art. no. 5514715.
[42] 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, Jan. 2022, Art. no. 5522214.
[43] B. Liu, A. Yu, K. Gao, X. Tan, Y. Sun, and X. Yu, "DSS-TRM: Deep spatial–spectral transformer for hyperspectral image classification," *Eur. J. Remote Sens.*, vol. 55, no. 1, pp. 103–114, 2022.
[44] Y. Wu, J. Feng, G. Bai, Q. Gao, and X. Zhang, "Hyperspectral image classification based on spectrally-enhanced and densely connected transformer model," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2022, pp. 2746–2749.
[45] Q. Ren, B. Tu, S. Liao, and S. Chen, "Hyperspectral image classification with IFormer network feature extraction," *Remote Sens*, vol. 14, no. 19, 2022, Art. no. 4866.
[46] J. Feng, X. Luo, S. Li, Q. Wang, and J. Yin, "Spectral transformer with dynamic spatial sampling and Gaussian positional embedding for hyperspectral image classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2022, pp. 3556–3559.
[47] K. Liu et al., "Mapping coastal wetlands using transformer in transformer deep network on China ZY1-02D hyperspectral satellite images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 3891–3903, May 2022.
[48] Z. Xue, Q. Xu, and M. Zhang, "Local transformer with spatial partition restore for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 4307–4325, May 2022.
[49] X. Zhou, W. Zhou, X. Fu, Y. Hu, and J. Liu, "MDvT: Introducing mobile three-dimensional convolution to a vision transformer for hyperspectral image classification," *Int. J. Digit. Earth*, vol. 16, no. 1, pp. 1469–1490, 2023.
[50] Y. Peng, Y. Liu, B. Tu, and Y. Zhang, "Convolutional transformer-based few-shot learning for cross-domain hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 1335–1349, Jan. 2023.
[51] F. Zhao, J. Zhang, Z. Meng, H. Liu, Z. Chang, and J. Fan, "Multiple vision architectures-based hybrid network for hyperspectral image classification," *Expert Syst. With Appl.*, vol. 234, 2023, Art. no. 121032.
[52] J. Yang, A. Li, J. Qian, J. Qin, and L. Wang, "A hyperspectral image classification method based on pyramid feature extraction with deformable–dilated convolution," *IEEE Geosci. Remote Sens. Lett.*, vol. 21, Jan. 2024, Art. no. 5500105.
[53] J. Gao, X. Ji, G. Chen, and R. Guo, "Main–sub transformer with spectral–spatial separable convolution for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 2747–2762, Jan. 2024.
[54] W. He, W. Huang, S. Liao, Z. Xu, and J. Yan, "CSiT: A multiscale vision transformer for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 9266–9277, Oct. 2022.
[55] Y. Peng, Y. Zhang, B. Tu, Q. Li, and W. Li, "Spatial–spectral transformer with cross-attention for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Sep. 2022, Art. no. 5537415.
[56] H. Xu, Z. Zeng, W. Yao, and J. Lu, "CS2DT: Cross spatial–Spectral dense transformer for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, Oct. 2023, Art. no. 5510105.
[57] C.-F. Chen, Q. Fan, N. Mallinar, T. Sercu, and R. Feris, "Big-little net: An efficient multi-scale feature representation for visual and speech recognition," *in Proc. 7th Int. Conf. Learn. Representations*, New Orleans, LA, USA, May 2019, *arXiv:1807.03848*.
[58] Y. Huang et al., "Cross-scene wetland mapping on hyperspectral remote sensing images using adversarial domain adaptation network," *ISPRS J. Photogramm. Remote Sens.*, vol. 203, pp. 37–54, 2023.
[59] W. Sun and J. Peng, "Cross-scene hyperspectral remote sensing wetland image data," *ISPRS J. Photogrammetry Remote Sens.*, Zenodo, Jul. 2023, doi: 10.5281/zenodo.8105220.
[60] G. M. Foody, "Status of land cover classification accuracy assessment," *Remote Sens. Environ.*, vol. 80, no. 1, pp. 185–201, 2002.
[61] J. A. Richards, "Classifier performance and map accuracy," *Remote Sens. Environ.*, vol. 57, no. 3, pp. 161–166, 1996.

Jinghui Yang received the B.S. degree in electronic information engineering, the M.S. in communication and information systems, and the Ph.D. degree in information and communication engineering from Harbin Engineering University, Harbin, China, in 2010, 2013, and 2016, respectively.

She is currently an Associate Professor and a Doctoral Supervisor with the School of Information Engineering, China University of Geosciences (Beijing), Beijing, China. Her research interests include hyperspectral imagery, remote sensing, pattern recognition, and signal processing.

Anqi Li received the B.S. degree in electronic information engineering in 2022 from the School of Information Engineering, China University of Geosciences (Beijing), Beijing, China, where she is currently pursuing the M.S. degree in communication engineering.

Her research interests include hyperspectral image processing and deep learning.



Jinxi Qian received the B.S. degree in communication engineering and Ph.D. degree in communication and information systems from Harbin Engineering University, Harbin, China, in 2008 and 2013, respectively.

Subsequently, he spent two years as a Postdoctor with the joint laboratory of the Institute of Telecommunication Satellite, China Academy of Space Technology, and Beijing University of Posts and Telecommunications, Beijing, China. He is currently a Professor with the Institute of Telecommunication and Navigation Satellites, China Academy of Space Technology, Beijing. His research interests include image processing, signal processing, wireless communication, and satellite communication.

Jia Qin received the B.S. degree in information engineering from Xihua University, Chengdu, China, in 2022. She is currently working toward the M.S. degree in communication engineering with the School of Information Engineering, China University of Geosciences (Beijing), Beijing, China.

Her research interest covers the classification of small-sample hyperspectral images.

Liguo Wang received the M.S. and Ph.D. degrees in signal and information processing from the Harbin Institute of Technology, Harbin, China, in 2002 and 2005, respectively.

From 2006 to 2008, he was a Postdoctoral Researcher with the College of Information and Communications Engineering, Harbin Engineering University, Harbin, China. He is currently a Professor with the College of Information and Communication Engineering, Dalian Minzu University, Dalian, China. His research interests include remote sensing image processing and machine learning.