Weighted Feature Fusion of Convolutional Neural Network and Graph Attention Network for Hyperspectral Image Classification

Article *in* IEEE Transactions on Image Processing · January 2022

DOI: 10.1109/TIP.2022.3144017

CITATIONS
119

READS
2,788

4 authors, including:

Yanni Dong
China University of Geosciences
37 PUBLICATIONS **779** CITATIONS
SEE PROFILE

Quanwei Liu
China University of Geosciences
2 PUBLICATIONS **133** CITATIONS
SEE PROFILE

Liangpei Zhang
Wuhan University
1,012 PUBLICATIONS **53,075** CITATIONS
SEE PROFILE

Weighted Feature Fusion of Convolutional Neural Network and Graph Attention Network for Hyperspectral Image Classification

Yanni Dong, *Member, IEEE*, Quanwei Liu, Bo Du, *Senior Member, IEEE*, and Liangpei Zhang, *Fellow, IEEE*

***Abstract*— Convolutional Neural Networks (CNN) and Graph Neural Networks (GNN), such as Graph Attention Networks (GAT), are two classic neural network models, which are applied to the processing of grid data and graph data respectively. They have achieved outstanding performance in hyperspectral images (HSIs) classification field, which have attracted great interest. However, CNN has been facing the problem of small samples and GNN has to pay a huge computational cost, which restrict the performance of the two models. In this paper, we propose Weighted Feature Fusion of Convolutional Neural Network and Graph Attention Network (WFCG) for HSI classification, by using the characteristics of superpixel-based GAT and pixel-based CNN, which proved to be complementary. We first establish GAT with the help of superpixel-based encoder and decoder modules. Then we combined the attention mechanism to construct CNN. Finally, the features are weighted fusion with the characteristics of two neural network models. Rigorous experiments on three real-world HSI data sets show WFCG can fully explore the high-dimensional feature of HSI, and obtain competitive results compared to other state-of-the art methods.**

***Index Terms*— Hyperspectral image classification, convolutional neural network, graph attention network, weighted feature fusion, attention mechanism.**

I. Introduction

HYPERSPECTRAL sensor can capture the spectral information and spatial information of the observed material simultaneously to form the hyperspectral images (HSIs) [1]. Usually, HSI are composed of hundreds of continuous spectral bands, which can be combined into data cubes to provide very detailed spectral information. In addition, with the development of hyperspectral technology and sensor platform, more detailed spatial information can also be obtained by hyperspectral sensors [2]. Rich spectral and spatial information make HSI widely used, such as environmental monitoring mineral investigation, agricultural evaluation, military reconnaissance, etc [3]–[7].

The HSI classification has generated considerable attention due to its importance, which refers to the determination of the category of ground objects represented by each pixel in HSI [8]. In the early stage, most of the classic machine learning-based methods are supervised learning methods such as k-nearest neighbors [9], logistic regression [10] and random forest [11], which could achieve satisfactory results in an ideal situation. However, traditional classification methods rely on experts to manually design features, which are usually shallow, resulting in the further limitation of the accuracy [12]. In contrast, deep learning (DL) [13] using neural networks can automatically extract hierarchical and nonlinear features with end-to-end learning. It has achieved great success in many fields such as driverless cars, target recognition, and machine translation [14]–[17].

In HSI classification, the most widely used deep neural networks include Deep Belief Neural Networks (DBN) [18], Auto-encoder (AE) [19], Recurrent Neural Networks (RNN) [20] and Convolutional Neural Networks (CNN) [21], [22]. Among them, CNN has the characteristics of local connection and weight sharing, which can significantly reduce the amount of parameters. Moreover, by dividing patches, it is also possible to capture spectral information and spatial information at the same time [23]. CNN-based methods are playing an increasingly important role in the field of HSI classification. The first spectral–spatial CNN method was proposed in [24], which achieves higher accuracy than traditional machine learning methods in the field of HSI classification. The end-to-end spectral-spatial residual network (SSRN) [25] could take raw 3D cubes as input data which avoids complex feature engineering on HSI. A hybrid spectral CNN (HybridSN) [26] was proposed, which is a spectral-spatial 3D-CNN followed by spatial 2D-CNN to further learns more abstract-level spatial representation. In addition, more and more DL methods are applied to HSI classification, such as residual network and the densely connected CNN [27], [28].

After continuous exploration of the CNN-based models, it has made remarkable achievements in the field of HSI

Manuscript received August 31, 2021; revised December 9, 2021; accepted January 7, 2022. Date of publication January 25, 2022; date of current version February 1, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 62171417 and Grant 41871243 and in part by the Science and Technology Major Project of Hubei Province (Next-Generation AI Technologies) under Grant 2019AEA170. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Leyuan Fang. *(Corresponding author: Bo Du.)*

Yanni Dong and Quanwei Liu are with the Hubei Subsurface Multi-scale Imaging Key Laboratory, Institute of Geophysics and Geomatics, China University of Geosciences, Wuhan 430074, China (e-mail: dongyanni@cug.edu.cn; liuquanwei@cug.edu.cn).

Bo Du is with the National Engineering Research Center for Multimedia Software, School of Computer Science, Institute of Artificial Intelligence, Wuhan University, Wuhan 430072, China, and also with the Hubei Key Laboratory of Multimedia and Network Communication Engineering, Wuhan University, Wuhan 430072, China (e-mail: dubo@whu.edu.cn).

Liangpei Zhang is with the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing, Wuhan University, Wuhan 430072, China (e-mail: zlp62@whu.edu.cn).

Digital Object Identifier 10.1109/TIP.2022.3144017

classification, but there are still two major challenging problems to be solved as follows: 1) As we all know, the acquisition of labeled samples is extremely resource intensive [29], [30]. For surpervised CNN-based method, the problem of small samples is protrudent, tending to occur overfitting phenomenon. 2) CNN, which is designed for Euclidean data, is used to process a regular spatial structure, so it cannot easily capture the internal connections of different land cover in HSI [31].

In order to solve the above problems, Graph Neural Networks (GNN) [32] have quickly attracted attention because of their wide applicability and powerful performance. GNN can perform DL of non-Euclidean data, that is, it can perform end-to-end representation learning of node feature information and structural information at the same time. HSI data can be converted into graph data through superpixel-based methods, then GNN methods can be used to efficiently model the spectral–spatial contextual information. In this way, the number of labels are implicitly expanded, which alleviates the problem of small samples to a certain extent. In [33], Graph Convolution Networks (GCN) based on superpixel segmentation is applied to HSI for the first time. It uses multi-order neighbor nodes to construct an adjacency matrix so that GCN can capture multi-scale spatial information. Then, a method to automatically learn the graph structure during training was proposed [34], which can promote the node feature learning and make the graph more adaptive to HSI content.

In GCN, the weights between nodes are fixed and cannot be changed, which limits the expressive ability of the network. In order to dynamically change the weight between nodes, a new GNN model named graph attention networks (GAT) [35] is proposed, which enables specifying different weights to different nodes in a neighborhood, without depending on knowing the graph structure upfront or requiring any kind of costly matrix operation. Using k-nearest neighbors to select neighbor nodes to construct the adjacency matrix, GAT can calculate the weight of each different node, which are used for HSI classification [36]. To alleviate a huge computational cost, similar to CNN-based HSI classification, the strategy of dividing subgraphs was used to construct adjacency matrix [37]. Pyramid structure is usually useful for feature extraction [38]. The spectral pyramid GAT [39] uses 3D-CNN to extract multi-scale spectral information and then applies graph attention mechanism to explicitly extract high-order spatial feature, which significantly improve the classification accuracy.

Both CNN and GNN are DL methods capable of capturing deep features. What are their characteristics in HSI classification tasks? Can we design a fusion strategy to integrate them and learn from each other to improve the ability of HSI classification? A well informed answer to this question (detailed answers in Section II) can help us understand the capability and limitations of GCN and CNN in a principled way, guiding us to further improve the ability of HSI classification.

This paper based on these two networks, i.e. CNN and GAT, designs two simple classifiers to explore their characteristics, and then proposes a weighted feature fusion of CNN and GAT networks (WFCG) to combine their characteristics. Technically, first, we use two 1×1 convolution layers to process original HSI to perform spectral feature compression and denoising. Then, the output spectral features are passed into two branches respectively, namely branch one and branch two. In the branch one, in order to obtain stable superpixel-level features, we use multi-head GAT and normalization layer to extract superpixel-level features, and the connection between them is realized by a graph encoder and a decoder. In the branch two, the processed pixel-level spectral feature is passed into the CNN combined with the attention modules to capture long-range deep feature. Finally, the features obtained by two branches are weighted fusion. The classification label $\mathbf{Y}$ can be obtained after classification using the softmax function. The main contributions of this paper are as follows:

(1) In order to further improve the performance of HSI, we analyze the classification performance of superpixel-based GAT and compare it with the classic CNN of pixel classification. The results showed that they are complementary in different training proportion datasets.

(2) We propose a novel hybrid DL framework through the weighted feature fusion of CNN and GAT. It uses graph encoder and decoder modules to connect grid and nodes, which can better combine the features of CNN and GAT for HSI classification.

(3) In order to ensure the stability of the network output, we first pass the GAT feature through the fully connected layer and the normalization layer. Moreover, in order to better capture long-range information and high-level feature, we combine the attention mechanism to design the branch of the convolutional network.

(4) Our extensive experiments on three well-known HSI data sets (Indian Pines, University of Pavia, WHU-Hi-HongHu) clearly show that WFCG outperforms the state-of-the-art CNN and GNN, which is prefect for challenging classification tasks.

The rest of the paper is organized as follows. In Section II, we design two simple weak classifiers to evaluate the characteristics of GAT and CNN, and give an intuitive explanation. In Section III, we propose WFCG and give a detailed introduction to each part. We will conduct detailed experiments and analysis on our proposed model in Section IV. Summarize in the last part, the Section V.

II. Model Performance Based on GNN and CNN: An Experimental Investigation

In this section, we designed two weak classifiers based on the superpixel-based GAT and CNN, and used two classic hyperspectral datasets (Indian pines and University of Pavia, which are introduced in detail in Section IV) perform classification experiments. By changing the ratio of training samples, we can clearly compare the classification performance and evaluate the characteristics of the two weak classifiers. Then the characteristics of the two models can be discovered. We first design a network based on GAT, which is only composed of two 1×1 convolutional layers and GAT. Then, we designed a semi-supervised CNN which consists of two 1×1 convolutional layers and two 3×3 convolutional layers.

For the Indian Pines dataset, which only has 10249 of the total number of labels and has a serious problem of unbalanced

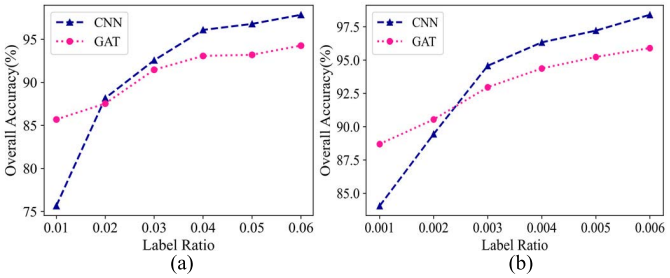

Fig. 1. Comparison of classification results of two different models. (a) Indian Pines. (b) University of Pavia.

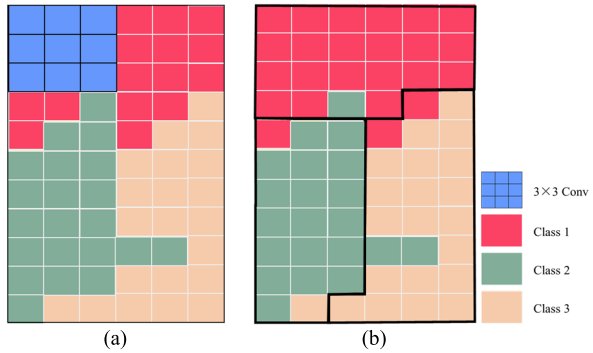

Fig. 2. Intuitively understand the two network processing modes. Each small box represents a pixel, and red, green and orange pixels represent different land covers. (a) Pixel-level feature extraction diagram of CNN, in which the blue box represents the processing of HSI by the 3 × 3 convolution kernel. (b) Superpixel-level feature extraction diagram of GNN, in which irregular thick black solid line frame represents superpixels.

samples, we change the ratio of training samples from 1% to 6%. As shown in Fig. 1(a), we can observe that when the size of training samples changes, GAT can maintain a fairly high accuracy in the case of fewer samples, and as the size of training samples increases, the accuracy of GAT increases slowly. However, the accuracy of CNN decreases rapidly with decreasing the size of training samples. For the University of Pavia dataset, which only contains 9 categories and 42,776 of the total number of labels, the ratio of the training samples is set from 0.1% to 0.6%. The experimental conclusions of University of Pavia dataset are the similar as the ones of Indian pines dataset, as shown in Fig. 1(b). When the number of labels is only 0.1%, GAT can achieve 88.69% accuracy, while CNN has only 84.04% accuracy. With the increase of the number of training samples, GAT grows slowly and tends to be saturated, while the accuracy of CNN increases rapidly.

This phenomenon indicates that the sensitivity of the CNN network and the GAT network to the sample size is different. As shown in Fig. 2(a), CNN is used for pixel-level feature extraction, which can be extracted separately by the convolution kernel. CNN can extract fine feature, but this may result in the need for more samples for learning to achieve higher accuracy.

As shown in Fig. 2(b), the label of the superpixel is determined by the pixel-based category of which contains the most labels. When the size of the training samples is small, due to the division of superpixels, some unknown labels are determined by the labels of the superpixels, and then the number of samples is implicitly increased, result in improving the classification accuracy. When the size of training samples increases, the selected labels are more likely to be concentrated in one superpixel. There is a certain error in the superpixel segmentation methods itself, which limits the further increase in accuracy. Based on the above reasons, we are motivated to design a fusion mechanism that can give full play to the characteristics of CNN and GNN, making up for their respective shortcomings and improving the accuracy of hyperspectral classification. However, the biggest obstacle in reality is that the feature distribution extracted by GNN is different from CNN. Direct application of CNN and GNN feature addition or concatenation may not get the best classification performance. Therefore, we proposed the WFCG algorithm.

III. Proposed Method

The proposed WFCG framework for the HSI classification is summarized by the flowchart in Fig. 3. We denote HSI as $\mathbf{X} \in \mathbb{R}^{H\times W\times B}$. H, W, and B are denoted as the height, width, the number of bands, respectively. The original HSI is first transformed by spectral feature. Then, the superpixel-level and pixel-level feature are extracted by branch one and branch two, respectively. These two features were weighted and fused. Finally, the label $\mathbf{Y}$ of each pixel is obtained through the softmax layer. The model is mainly divided into eight modules. The main modules of the model will be described in detail below.

A. Convolutional Layers

CNN is a special type of DNN, which is inspired by neuroscience. The architecture of CNN is different from other DL models. Instead of using a fully connected approach, CNN uses local connections to extract the contextual feature information. In addition, instead of assigning weights to each input separately, CNN uses shared weights to greatly reduce the amount of parameters. Based on these characteristics, CNN tends to provide better generalization when facing computer vision problems.

Inspired by Network-in-Network [40], in the initial stage of the network we use two 1×1 convolutional layers. Due to original HSI contains noise and redundant information, they are mainly used as cross-channel information exchange to remove useless spectral information to improve discrimination ability. Then, they are used as dimension reduction modules to remove computational cost. This allows for not only increasing the depth, but also the width of our networks without significant performance penalty. Specifically, by inputting the HSI feature $\mathbf{X}_{h,w,b}$ into the spectral convolutional layer, we have

$$\mathbf{X}^{*}_{h,w,m} = \sigma\Big(\sum_{i,j,b} \mathbf{K}_{i,j,b,m}\mathbf{X}_{h,w,b} + \mathbf{B}_{i,j,b,m}\Big), \tag{1}$$

where $\mathbf{X}^{*}_{h,w,m}$ denotes output feature map. h, w, m represents length, width and channels respectively. $\mathbf{K}_{i,j,b,m}$ denotes the convolution kernel of the input feature map in row i, column j and channel b, m is the number of convolution kernels. $\mathbf{B}_{i,j,b,m}$ represents the bias of the convolution, and $\sigma(\cdot)$ represents the leakyReLU activation function.

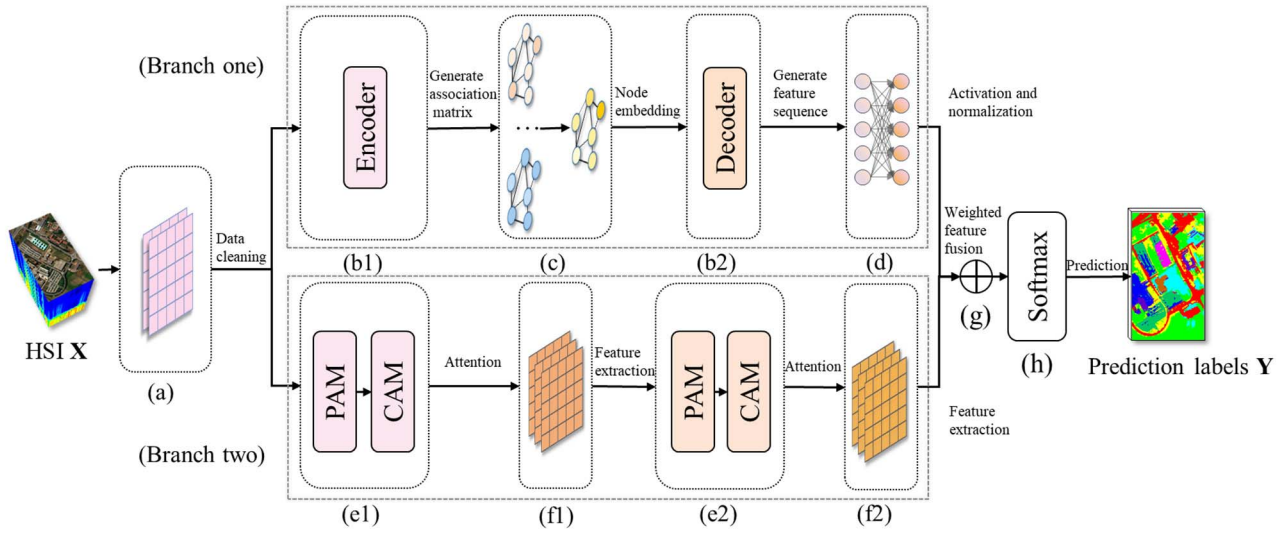

Fig. 3. The framework of WFCG model. The model is mainly divided into eight parts. (a) Spectral convolutional layers. (b1) and (b2) Data conversion module. (c) Graph attention module. (d) Non-linear feature transformation module. (e1) and (e2) Position attention module and channel attention module. (f1) and (f2) Depthwise separable convolutional layers. (g) Weighted feature fusion module. (h) Softmax layer.

1×1 convolution can only extract the features in the spectral domain, while HSI contains rich spatial information. 2D-CNN can be used to extract the contextual spatial features. However, due to the numerous parameters in the convolutional kernels and the limited training samples in the task, 2D-CNN may be easily overfitting. Thus, we use a lightweight depthwise separable convolution [41] to build 2D-CNN, which can greatly reduce the parameters and enhance the robustness. Depthwise convolution with one filter per input channel can be written as

$$\hat{\mathbf{X}}^{*}_{h,w,b} = \sigma\Big(\sum_{i,j,b} \hat{\mathbf{K}}_{i,j,b}\mathbf{X}_{h,w,b} + \hat{\mathbf{B}}_{i,j,b}\Big), \tag{2}$$

where $\hat{\mathbf{X}}^{*}_{h,w,b}$ denotes the output feature map. $\hat{\mathbf{K}}_{i,j,b}$ denotes convolution kernel. $\hat{\mathbf{B}}_{i,j,b}$ denotes bias.

B. Feature Conversion and Superpixel-Level Feature Extraction

GNN only accepts graph data as input data, but the feature map generated by the module (a) is arranged in a standard rectangular grid. Although we can treat image pixels as the nodes in a graph, GNN requires input adjacency matrix, which will make the calculation cost unbearable. To address this problem, we adopt Simple Linear Iterative Clustering (SLIC) [42] to group pixels into perceptually meaningful superpixels. SLIC segmentation adopts k-means clustering approach to generate superpixels as graph nodes. Graph attention enables neural networks to learn useful graph representations by selectively attending to different nodes.

Due to the different number of pixels contained in superpixels, we cannot integrate the above segmentation method directly into the WFCG framework. Inspired [34], we apply a data conversion to allow the feature to be propagated between the pixel and superpixel, as shown in Fig. 4. Concretely, let $\mathbf{O} \in \mathbb{R}^{HW \times Z}$ be the association matrix between pixels and

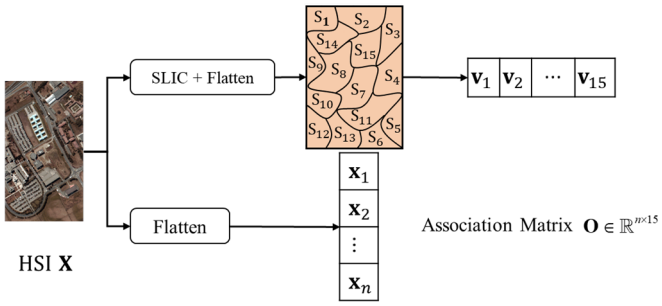

Fig. 4. The flowchart of the pixel and superpixel data conversion module, where $\mathbf{X}_i$ denotes the ith pixel in the flattened HSI, and $\mathbf{V}_j$ denotes the average radiance of the pixels contained in the superpixels S_j.

superpixels, where Z denotes the number of superpixels, then we have

$$\mathbf{O}_{i,j} = \begin{cases} 1, & \text{if } \bar{\mathbf{X}}_i \in S_j \\ 0, & \text{if } \bar{\mathbf{X}}_i \notin S_j \end{cases} \quad \bar{\mathbf{X}} = Flatten(\mathbf{X}), \tag{3}$$

where $Flatten(\cdot)$ denotes the HSI data flattened by the spatial dimension. $\mathbf{O}_{i,j}$ denotes the value of $\mathbf{O}$ at location (i, j). $\bar{\mathbf{X}}_i$ denotes the i-th pixel in $\bar{\mathbf{X}}$. Then, we can use the following formula to achieve feature conversion as

$$\mathbf{V} = Encoder(\mathbf{X}; \mathbf{O}) = \bar{\mathbf{O}}^{T} Flatten(\mathbf{X}), \tag{4}$$

$$\tilde{\mathbf{X}} = Decoder(\mathbf{V}; \mathbf{O}) = Reshape(\mathbf{OV}), \tag{5}$$

where $\bar{\mathbf{O}}$ denotes the normalized $\mathbf{O}$ by column. $\mathbf{V}$ denotes the nodes composed of superpixels, and $Reshape(\cdot)$ denotes restoring the spatial dimension of the flattened data. $\tilde{\mathbf{X}}$ denotes the feature converted back to grid. After SLIC segmentation, the feature can be regarded as an undirected graph as $G = (V, E)$, where V and E denote the nodes and edges, respectively. Here the feature of the node is the average value of the pixel feature in the superpixel.

After encode the latent representation of the pixels into the superpixels, we apply attention module to get node

embeddings, representing the current node by aggregating neighbor information, so it can learn spatial feature. Specifically, in order to obtain sufficient expressive power, we first use the weight matrix $\mathbf{W} \in \mathbb{R}^{B\times C}$ to perform a linear transformation on the input nodes feature $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_Z]$, where C represents the number of channels of $\mathbf{v}$. Then the shared attention mechanism a is used to calculate the attention coefficient as

$$e_{ij} = a(\mathbf{W}\mathbf{v}_i, \mathbf{W}\mathbf{v}_j). \tag{6}$$

This coefficient can represent the importance of node j relative to node i. In order to capture the boundary information more accurately, we use the first-order attention mechanism, that is, only the node j connected to node i is calculated. Next we use the softmax function to normalize the attention coefficient into weight information as

$$\alpha_{ij} = softmax(e_{ij}) = \frac{\exp(e_{ij})}{\sum\limits_{k\in N_i}\exp(e_{ik})}, \tag{7}$$

where N_i denotes the neighborhood of node i. Finally, we use the Leaky ReLU activation function to encapsulate this process into a single-layer feedforward neural network. Then we have

$$\alpha_{ij} = \frac{\exp(\text{leaky ReLU}(a^T[\mathbf{W}\mathbf{v}_i||\mathbf{W}\mathbf{v}_j]))}{\sum\limits_{k\in N_i}\exp(\text{leaky ReLU}(a^T[\mathbf{W}\mathbf{v}_i||\mathbf{W}\mathbf{v}_k]))}, \tag{8}$$

where $||$ is the concatenation operation. a^T is the transposition of a, which denotes the learnable parameters. As a result, the node embeddings can be expressed as

$$\mathbf{v}_i' = \sigma\Big(\sum_{j\in N_i}\alpha_{ij}\mathbf{W}\mathbf{v}_j\Big). \tag{9}$$

In order to enable node embeddings to stably represent the node i, we apply the multi-head attention mechanism in the first attention layer, that is, execute Eq.(9) K times independently and then concatenate the obtained node embeddings, resulting in the following output node representation as

$$\mathbf{v}_i' = ||_{k=1}^{K}\sigma\Big(\sum_{j\in N_i}\alpha_{ij}^k\mathbf{W}^k\mathbf{v}_j\Big). \tag{10}$$

After we get the node embeddings, we use the decoder module to convert the node feature into grid feature. Only the multi-head attention mechanism cannot guarantee stable output, while gradient explosion and gradient disappearance will occur in the process of model training [43]. To make the final input more stable, we apply layer normalization as

$$\tilde{\mathbf{X}}^* = \frac{\tilde{\mathbf{X}} - E[\tilde{\mathbf{X}}]}{\sqrt{Var[\tilde{\mathbf{X}}]}}\times\gamma + \beta, \tag{11}$$

where the expectation $E(\tilde{\mathbf{X}})$ and variance $Var[\tilde{\mathbf{X}}]$ are computed over the decoder grid feature. γ and β are learnable parameters.

In summary, we use the encoder module to convert grid feature into node feature, so that the GNN can be smoothly integrated into the convolutional network framework. With the help of graph attention module generated attention coefficient α_{ij} dynamically just depending on the local neighborhood and rearranges the neighbors by their importance, which makes the model more flexible to specific input sample. Then the decoder module is used to perform the inverse transformation of node feature to complete the pixel-level classification task. In order to keep the final output stable, we apply a normalization layer to normalize the output feature. Consequently, we can obtain stable feature output through branch one.

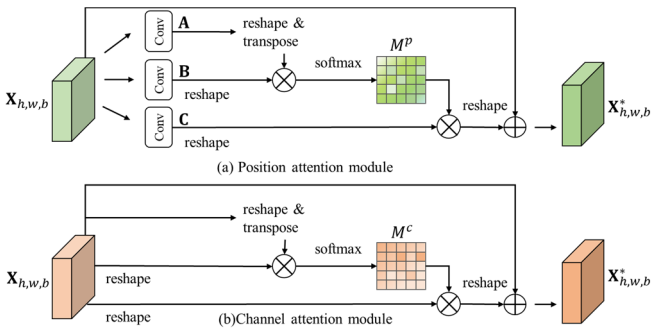

Fig. 5. The details of (a) position attention module, and (b) channel attention module. Conv denotes a convolutional layer. M^P and M^C denotes attention weight matrix $\otimes$ is matrix multiplication and $\oplus$ is broadcast element-wise addition.

C. Position Attention Module and Channel Attention Module

The attention mechanism is a processing mechanism of the brain for input signals, which allows humans to use limited resources to process more information [44]. Inspired by this phenomenon, in order to better exploit the correlation between the hyperspectral pixels and channels, we introduce the self-attention mechanism [45] to build the attention module. The position attention module could encodes a wider range of contextual information into local feature, thus enhancing their representation capability, and the channel attention module emphasize interdependent feature maps and improve the feature representation of specific semantics.

1) Position Attention Module: As illustrated in Fig. 5(a), given a local feature $\mathbf{X}_{h,w,b}$, we first use the convolution to implement a nonlinear transformation to generate two new feature maps, i.e., $\mathbf{A}$ and $\mathbf{B}$, where $\{\mathbf{A}, \mathbf{B}\} \in \mathbb{R}^{H\times W\times C}$. Their dimensions are reshaped by $\mathbb{R}^{N\times C}$, where N is the total number of input local feature pixels. Then, we perform a matrix multiplication between the transpose of $\mathbf{A}$ and $\mathbf{B}$, applying a softmax layer to calculate the position attention map $\mathbf{M}^P \in \mathbb{R}^{N\times N}$ as

$$m_{ji}^p = \frac{\exp(A_i\cdot B_j)}{\sum\limits_{i=1}^{N}\exp(A_i\cdot B_j)}, \tag{12}$$

where m_{ji}^p measures the i-th position impacted on j-th position. The position attention map is a matrix of which the shape is the same as the number of input feature maps and the value is in range of (0, 1). The size of this value represents the importance of this position. In the same way, we get the feature map $\mathbf{C}$ and perform a matrix multiplication between it and spatial attention map, reshaping the results to $\mathbb{R}^{H\times W\times C}$.

Finally, in order to better extract the attention information without losing the original information, we multiply it by a learnable scale parameter α [46] and perform an element-wise sum operation with input feature $\mathbf{X}_{h,w,b}$ to obtain the final output $\mathbf{X}^*_{h,w,b}$ as follows:

$$X^*_{h,w,b,j} = \alpha \sum_{i=1}^{N} (m^p_{ji} C_i) + X_{h,w,b,j}. \tag{13}$$

2) Channel Attention Module: In contrast to the position attention, the channel-wise attention refines the weight of feature maps. The structure of channel attention module is illustrated in Fig. 5(b). We directly calculate the channel attention map $\mathbf{M}^C \in \mathbb{R}^{C\times C}$ from the original feature $\mathbf{X}_{h,w,b}$ as

$$m^c_{ji} = \frac{exp(X_{h,w,b,i} \cdot X_{h,w,b,i})}{\sum_{i=1}^{N} exp(X_{h,w,b,i} \cdot X_{h,w,b,i})}. \tag{14}$$

In addition, similar to the position attention module, we perform a matrix multiplication between the spatial attention map and input feature $\mathbf{X}^*$, reshaping the results to $\mathbb{R}^{H\times W\times C}$. Finally, we multiply it by a learnable scale parameter β and perform an element-wise sum operation with input feature $\mathbf{X}^*$ to obtain the final output $\mathbf{Z} \in \mathbb{R}^{H\times W\times C}$. To be summarized, the channel attention is computed as

$$X^*_{h,w,b,j} = \beta \sum_{i=1}^{N} (m^c_{ji} X_{h,w,b,j}) + X_{h,w,b,j}. \tag{15}$$

Noted that each attention layer is added to the original input feature through the residual connection, we can highlight the important information while retaining the original information. Therefore, in order to obtain higher-level abstract information, we do not employ the original parallel scheme, but a serial scheme. Through these two effective and flexible attention layers, we have expanded the expressive capabilities of CNN so that the network can not only adaptively learn targets at different scales, but also benefit from long-range dependencies for classification tasks.

D. Weighted Feature Fusion and Classification

Due to the construction of superpixels, the number of labels is implicitly expanded, so that our model could maintain high accuracy even with a small amount of labels. The CNN branch embedded in the attention layer has powerful expression capabilities, adaptively transforming information and capturing accurate boundary information in different dimensions, positions, and scales.

Due to the influence of the different neural network models of the two branches, the feature distributions of the two branches are different. We assign different weights η to the two branches to perform different degrees of scaling so that the model can be better integrated, as follows:

$$\mathbf{F} = \eta \cdot \mathbf{C} + (1 - \eta) \cdot \mathbf{G}, \tag{16}$$

where $\mathbf{G}$, $\mathbf{C}$ and $\mathbf{F}$ denote the branch one, branch two and the final fused feature map, respectively. To train the network, the loss function can be denoted via the cross-entropy, which can be written as

$$L(\mathbf{Y}, \mathbf{P}) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} y_{i,c} \log(p_{i,c}), \tag{17}$$

where $\mathbf{Y}$ denotes ground truth and $\mathbf{P}$ denotes the predicted value of each pixel, and $y_{i,c}$ denotes the c-th element of the label $\mathbf{Y}$. $p_{i,c}$ denotes the probability of the pixel i belonging to the c-th class, which could be calculated by the softmax function. C and N refer to the total number of categories and samples of the training datasets, respectively.

TABLE I

THE LAND COVER CATEGORIES AND THE DATA SET DIVISION FOR EACH CATEGORY OF INDIAN PINES DATASET

No	Class	Train	Validation	Test
1	Alfalfa	1	1	44
2	Corn-notill	15	15	1398
3	Corn-min	9	9	812
4	Corn	3	3	231
5	Grass-pasture	5	5	473
6	Grass-tress	8	8	714
7	Grass-pasture-mowed	1	1	26
8	Hay-windrowed	5	5	468
9	Oats	1	1	18
10	Soybean-notill	10	10	952
11	Soybean-mintill	25	25	2405
12	Soybean-clearn	6	6	581
13	Wheat	3	3	199
14	Woods	13	13	1239
15	Buildings-Grass-Trees	4	4	378
16	Stone-Stell-Towers	1	1	91
	Total	110	110	9619

IV. EXPERIMENTAL RESULTS

This section summarizes our data sets, experimental settings, results, and a brief analysis of the proposed WFCG.

A. Data Description

In order to comprehensively evaluate the performance of the model, three well-known hyperspectral data sets were used for comparative experiments, i.e., the Indian Pines dataset, the University of Pavia dataset, and the WHU-Hi-HongHu dataset.

1) The Indian Pines Dataset: The Indian Pines dataset was collected by Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor in 1992, which is one of the earliest dataset used for HSI classification [47]. It uses an image with a spatial resolution of 20m $\times$ 20m, covering 145×145 pixels. The wavelength range is 0.4-2.5μm, including 220 continuous bands. After removing 20 water absorption and noisy bands, 200 bands are reserved. About half of the data (10,249 pixels from a total of 21,025) are labeled into 16 different classes. For details of each category and data set division, see Table I.

2) The University of Pavia Dataset: The University of Pavia dataset is captured by using the Reflective Optics System Imaging Spectrometer (ROSIS) sensor over the University of Pavia, Italy, in 2001 [12]. The size of this dataset is 610×340, containing 207,400 pixels in total. The resolution is 1.3 meters,

TABLE II

THE LAND COVER CATEGORIES AND THE DATA SET DIVISION FOR EACH CATEGORY OF UNIVERSITY OF UNIVERSITY OF PAVIA DATASET

	Class	Train	Validation	Test
1	Asphalt	7	7	6617
2	Meadows	19	19	18611
3	Gravel	3	3	2093
4	Trees	4	4	3056
5	Metal Sheets	2	2	1341
6	Bare Soil	6	6	5017
7	Bitumen	2	2	1326
8	Bricks	4	4	3674
	Shadow	1	1	945
	Total	48	48	42680

TABLE III

THE LAND COVER CATEGORIES AND THE DATA SET DIVISION FOR EACH CATEGORY OF WHU-HI-HONGHU DATASET

No	Class	Train	Validation	Test
1	Chinese cabbage	4	4	3312
2	Garlic sprout	2	2	1605
3	Rape	22	22	21777
4	Road	2	2	1788
5	Carrot	40	40	39289
6	White radish	17	17	16942
7	Lactuca sativa	5	5	4044
8	Tuber mustard	11	11	10284
9	Bare soil	13	13	12132
10	Cotton	7	7	6533
11	Film covered lettuce	9	9	8569
12	Cabbage	19	19	18477
13	Brassica parachinensis	8	8	7340
14	Pakchoi	2	2	998
15	Romaine lettuce	8	8	7246
16	Red roof	4	4	3002
17	Celtuce	4	4	3074
18	Brassica chinensis	5	5	4191
19	Small brassica chinensis	3	3	2609
	Total	185	185	173212

and the wavelength range is from 0.43 to 0.86 μm, including 115 continuous bands. After removing the disturbing bands, the remaining 103 bands are used for research. The available ground-truth information comprises about 20% of the pixels (42,776 of 207,400), labeled into 9 different classes. For details of each category and data set division, see Table II.

3) The WHU-Hi-HongHu Dataset: The Wuhan University Hyperspectral Image (WHU-Hi) dataset was acquired in HongHu city, China, in 2017, by a unmanned aerial vehicle (UAV) platform [48]. The experimental area is a complex agricultural scene with many types of crops. We intercepted the region with a size of 410×478 pixels, which is typical of the regions affected by land fragmentation. Several different varieties of one crop are grown in this area, such as Chinese cabbage/cabbage and brassica chinensis/small brassica chinensis, which have highly similar spectral curves. The UAV flew at an altitude of 100m, so it has a very high spatial resolution, about 0.043 m. There are 270 bands with wavelengths ranging from 0.4 to 1 nm. The available ground-truth information comprises about 90% of the pixels (173,582 of 195,980) labeled into 19 different classes. For details of each category and data set division, see Table III.

B. Experimental Settings

The proposed algorithm is implemented via Python 3.8.5 and Pytorch1.7.0. The hardware used for training is an i7-10700K CPU and a NVIDIA GeForce RTX 3090 GPU.

Several recent state-of-the-art HSI classification methods are used for comparison, including machine learning-based method, i.e., support vector machine (SVM); CNN-based methods, i.e., 1D-CNN [49], Joint Spatial–Spectral Attention Network (JSAN) [50], and 3D-2D SN spectral CNN (HybirdSN) [26]; GNN-based methods, i.e., Graph Convolution Network (GCN) [32], Multi-scale Dynamic Graph Convolutional Network (MDGCN) [33], and CNN-Enhanced Graph Convolutional Network (CEGCN) [34]. For fair comparison, all experiments are performed in the same environment using the hyperparameters in the original paper.

For the Indian Pines dataset, different types of the data have relatively similar spectral curves, and the sample categories are extremely unbalanced. For example, Soybeans-min occupies 2455 pixels. Oats only occupies 20 pixels, which makes classification difficult. We choose 1% of the labeled pixels in each land-cover category are randomly chosen as training samples and validation samples, the rest as testing samples. For the University of Pavia dataset and the WHU-Hi-HongHu dataset, because they have more labels, 0.1% the labels are selected as the training set and the validation set. In order to prevent the training set and the validation set from being empty, we select samples by rounding up. To ensure a fair comparison, all the methods used the same ground-truth data for each dataset. Training set, validation set and test set for each land-cover category is shown in Tables I-III, respectively.

Furthermore, in order to quantitatively and qualitatively evaluate quantify the classification performance of WFCG, six quantitative evaluation indicators were used in this experiment: per-class accuracy, the overall accuracy (OA), average accuracy (AA), kappa coefficient (kappa), training time and testing time are employed. Visualization of experimental results is also provided. All experimental results are the average of five independent runs.

For the proposed method, there are two main tuning parameters, i.e., kernel size k and fusion coefficient η, which control the receptive field size of CNN and the feature scaling ratio of the two branches, respectively. For the kernel size, considering the trade-off between the receptive field size and the computational cost, we set the first convolution layer kernel size and the second to (3, 3), (3, 5), (5, 3), (5, 5), respectively. As can be seen in Table IV, the results are relatively stable with regard to the variation of kernel size. Thus, this inspired us to set kernel size as (3, 5) in the experiments. For the fusion coefficient, we set the range of $\eta = 0.01$, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, and 0.6 to investigate the effect of parameters η. As shown in Fig. 6, the three data sets showed the same trend, and the best OA was achieved when η was equal to 0.05. Inspired this, the fusion coefficient of WFCG is fixed to 0.05 in all experiments. Table V details the configuration of

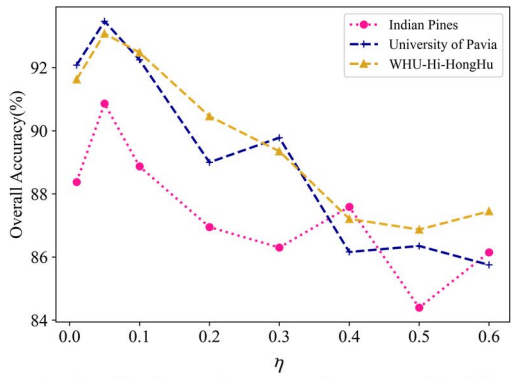

Fig. 6. The classification performance of WFCG with different fusion coefficients η on each data set.

TABLE IV

CLASSIFICATION PERFORMANCE OF WFCG WITH DIFFERENT KERNEL SIZE k ON EACH DATASET. (3, 3) MEANS THAT THE CONVOLUTION KERNELS OF THE FIRST AND SECOND CONVOLUTION LAYERS ARE 3×3

	(3,3)	(3,5)	(5,3)	(5,5)
Indian Pines	87.94	90.86	85.68	88.98
University of Pavia	91.34	93.47	91.45	92.33
WHU-Hi-HongHu	91.49	91.39	91.33	91.29

TABLE V

THE NETWORK OF WFCG FOR LAYER CONFIGURATIONS. (A), (C), (F1), (D), (F2), (G) REFER TO THE MODULES IN FIG. 3, RESPECTIVELY. 1×1@128 REFERS TO THE USE OF 1×1 CONVOLUTION KERNEL, THE NUMBER OF CHANNELS IS 128. 30, 3 REFERS TO THE NUMBER OF HIDDEN LAYER NEURONS AND THE NUMBER OF GAT HEADS, RESPECTIVELY. 0.05 REFERS TO THE FUSION COEFFICIENT

	Kernel Size and Fusion Coefficient	Branch One Hidden Units	Branch Two Kernel Size
(a)	1×1@128 1×1@128	-	-
(c)/(f1)	-	30, 3	3×3@128
(d)/(f2)	-	64	5×5@64
(g)	0.05	-	-

our WFCG for the layerwise network architecture. In addition, we use Adam optimizer to train the model with a learning rate 0.001. The l_2-norm regularization is set to 0.0001 to stabilize the network training and reduce overfitting. The number of superpixels for each dataset is set to 1/100 of the number of pixels. The training epoch is set to 300.

C. Experimental Results

The classification accuracies of different methods on each data set are detailed in Tables VI-VIII, and the classification maps obtained by these methods are illustrated in Figs. 7-9.

1) The Indian Pines Dataset: As shown in Table VI, due to the lack of training samples and extreme imbalance of samples, the comparison algorithms perform poorly on the Indian Pines dataset, but the OA of the proposed WFCG still achieved 90.86%, while the AA and kappa of which also exceed other comparison algorithms. It can be observed that SVM and 1D-CNN have achieved relatively poor results because they may not make full use of the training samples. Although JSSAN and HybridSN have strong expression ability [51], the accuracies of which are relatively low with small training samples. For example, in the case of class 7, their overall accuracies are only 30.97% and 35.24%, respectively. GCN-based methods generally can achieve higher accuracy in the case of small samples. However, due to the impact of superpixel segmentation accuracy, the classification accuracies of GCN-based methods are also difficult to be further improved. CEGCN adopts the concatenate feature fusion method, which may cause difficulty in feature fusion and result in unsatisfactory classification results. In contrast, WFCG can fully explore the feature of the samples and has strong expression ability, which can be stable to obtain higher classification accuracy. In the terms of time efficiency, WFCG is competitive compared to other comparison algorithms. Furthermore, from the classification results in Fig. 7, smoother visual effect and fewer misclassifications can be observed in the classification map of the proposed WFCG when compared with other competitors. For example, the two categories Corn-notill and Buildings-Grass-Trees, SVM methods present the characteristic "salt and pepper" noise in the obtained classifications, which is reduced by the CNN methods. GCN methods can further obtain smoother features, but at the cost of misclassification. The method we proposed obtains a smooth classification map while ensuring the correct classification.

2) The University of Pavia Dataset: As shown in Table VII, for the University of Pavia data set, it is worth noting that the classification accuracy of SVM is higher than that of HybridSN, which indicates that the unique advantages of the SVM algorithm based on support vectors and kernel techniques. The OA of MDGCN is higher than that of CEGCN, but AA is lower. Combined with the classification result figure shown in Fig. 8, this may be because MDGCN has the advantage in capturing large targets, while CEGCN can achieve better results on fine-grained targets. The proposed WFCG method has the advantages of GAT while integrating the effects of CNN, and the results show that the best effectiveness. Although the training time of the proposed algorithm is little higher than some comparison algorithms, the WFCG is still a competitive model by considering its stable performance and fast test time. The classification maps obtained by these methods are illustrated in Fig. 8. It can be observed that WFCG results in a significantly reduced misclassification rate.

3) The WHU-Hi-HongHu Dataset: As shown in Table VIII, SVM, 1D-CNN and HybridSN can obtain lower classification accuracies, which show the stability of classical machine-learning-based methods and the weakness of DL-based methods in dealing with small samples. In addition, JSSAN can achieve classification performance comparable to that based on GCN, indicating that the use of a multi-scale fusion attention mechanism may fully mine the information of the training samples and improve the classification accuracy. For the GCN-based methods, note that MDGCN is not used for comparison, as they are difficult to scale to this large data set (due to video memory consumption and time efficiency). CEGCN is not as good as the simple GCN model, indicating it may not integrate CNN and GCN well. The proposed WFCG method

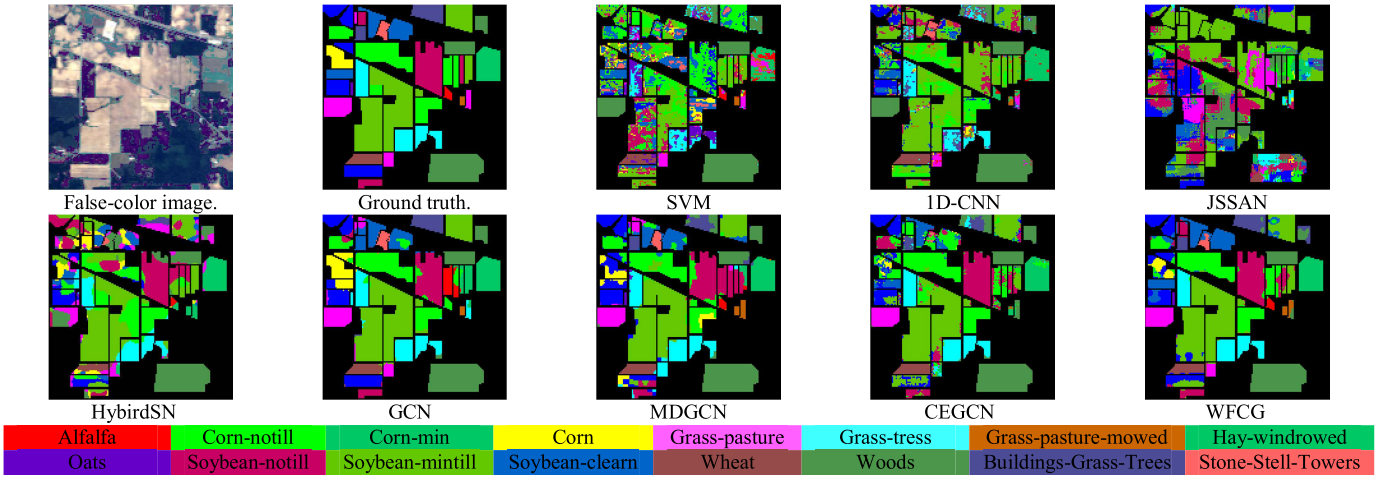

Fig. 7. False-color image, Ground truth and classification maps obtained by different methods on Indian Pines dataset.

TABLE VI

COMPARISON RESULTS OBTAINED BY VARIOUS METHODS ON INDIAN PINS DATASET, INCLUDING PER-CLASS ACCURACY, OA, AA, KAPPA, TRAIN TIME AND TEST TIME

	SVM	1D-CNN	JSSAN	HybirdSN	GCN	MDGCN	CEGCN	WFCG
class 1	15.91	16.02	52.46	75.00	**100.00**	90.91	80.00	91.57
class 2	42.92	57.83	77.98	46.72	81.80	71.73	70.16	**90.52**
class 3	34.36	55.58	58.34	68.73	83.21	**93.66**	70.05	85.59
class 4	37.23	38.49	**85.76**	73.88	76.29	49.03	96.54	79.53
class 5	60.68	45.29	81.14	76.44	93.33	66.23	90.43	**98.02**
class 6	81.79	73.00	83.68	85.73	**96.05**	90.45	79.12	94.01
class 7	50.00	17.44	50.42	59.24	77.37	**100.00**	67.22	58.89
class 8	95.94	83.29	90.96	94.95	**98.03**	96.69	90.87	95.71
class 9	27.78	6.11	30.97	35.24	54.76	**77.78**	70.00	73.91
class 10	31.83	62.07	78.60	58.45	76.86	69.75	79.11	**92.27**
class 11	70.64	54.61	84.52	72.28	91.03	89.40	70.81	**92.35**
class 12	39.59	37.24	77.42	49.33	80.53	66.87	72.50	**81.51**
class 13	87.44	72.07	68.62	81.23	98.44	**100.00**	96.20	93.15
class 14	81.36	75.40	92.53	79.91	97.59	**99.28**	85.83	97.50
class 15	22.22	32.06	89.59	58.68	81.83	70.83	**92.30**	86.23
class 16	65.93	**96.89**	64.88	58.86	75.29	89.01	79.22	95.37
OA(%)	58.51	59.37	79.44	66.12	87.24	83.24	77.00	**90.86**
AA(%)	52.85	51.46	72.99	67.17	85.15	82.60	80.65	**87.88**
kappa×100	52.20	52.75	76.57	61.13	85.44	88.52	72.70	**89.58**
train time(s)	0.02	7.79	50.44	25.08	4.22	245.32	6.40	17.83
test time(s)	0.69	2.75	5.41	4.27	0.69	2.68	0.40	0.38

TABLE VII

COMPARISON RESULTS OBTAINED BY VARIOUS METHODS ON UNIVERSITY OF PAVIA DATASET, INCLUDING PER-CLASS ACCURACY, OA, AA, KAPPA, TRAIN TIME AND TEST TIME

	SVM	1D-CNN	JSANN	HybirdSN	GCN	MDGCN	CEGCN	WFCG
class 1	68.66	74.91	58.84	45.44	81.84	**92.45**	85.51	91.67
class 2	88.57	78.49	89.49	84.97	95.40	**98.98**	95.94	97.52
class 3	33.01	28.56	54.23	53.07	70.86	71.18	78.48	**78.91**
class 4	53.27	72.96	50.49	43.04	76.05	71.21	93.77	**95.21**
class 5	37.96	94.85	80.27	82.77	94.56	**98.03**	94.96	92.31
class 6	46.48	43.71	84.48	73.22	89.40	98.74	**98.80**	97.14
class 7	75.49	17.68	65.32	37.73	67.98	**94.99**	80.28	88.95
class 8	79.29	59.55	61.43	46.02	65.02	**86.67**	77.84	85.54
class 9	99.68	99.95	27.36	0.00	95.93	70.70	**99.74**	97.51
OA(%)	72.73	71.54	75.70	66.60	85.09	92.75	91.44	**93.47**
AA(%)	64.71	63.41	63.55	51.81	81.90	86.99	89.48	**91.64**
kappa×100	63.18	64.95	67.25	54.27	80.28	85.85	88.51	**91.37**
train time(s)	0.00	3.42	28.06	12.07	23.58	634.89	37.35	190.91
test time(s)	2.94	11.13	22.19	17.52	4.23	17.87	6.26	6.12

can achieve the best results in OA, AA, and kappa at a certain time cost. We can also see from Fig. 9 that WFCG can get a cleaner classification map than other comparison methods, especially for the Celtuce and Carrot.

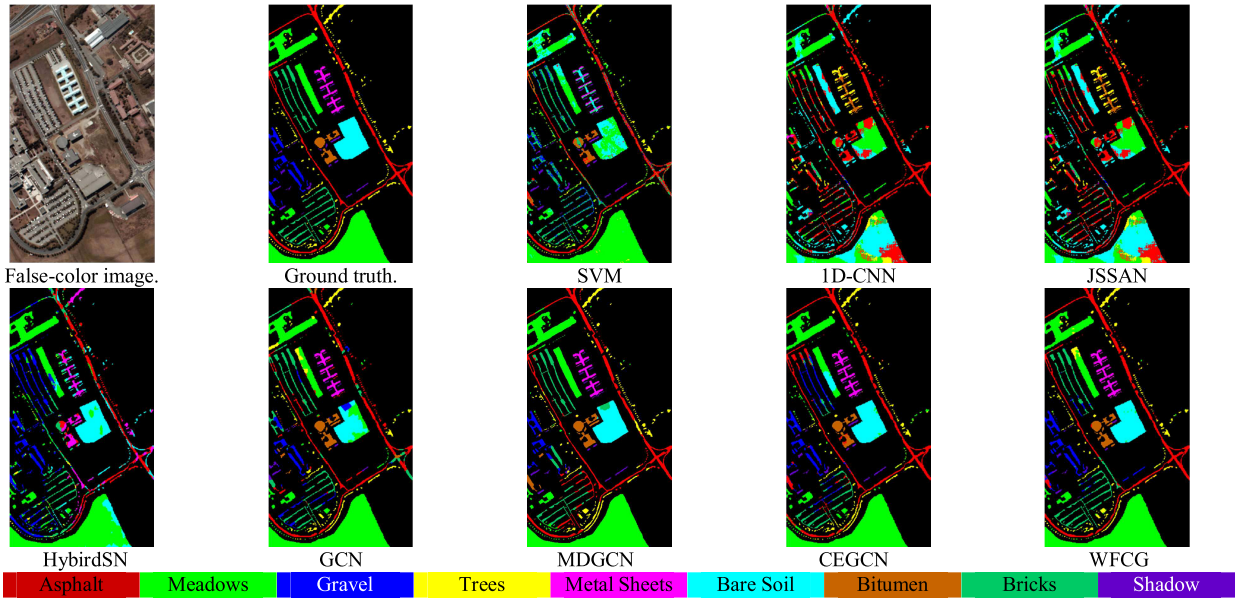

Fig. 8. False-color image, Ground truth and classification maps obtained by different methods on Pavia University dataset.

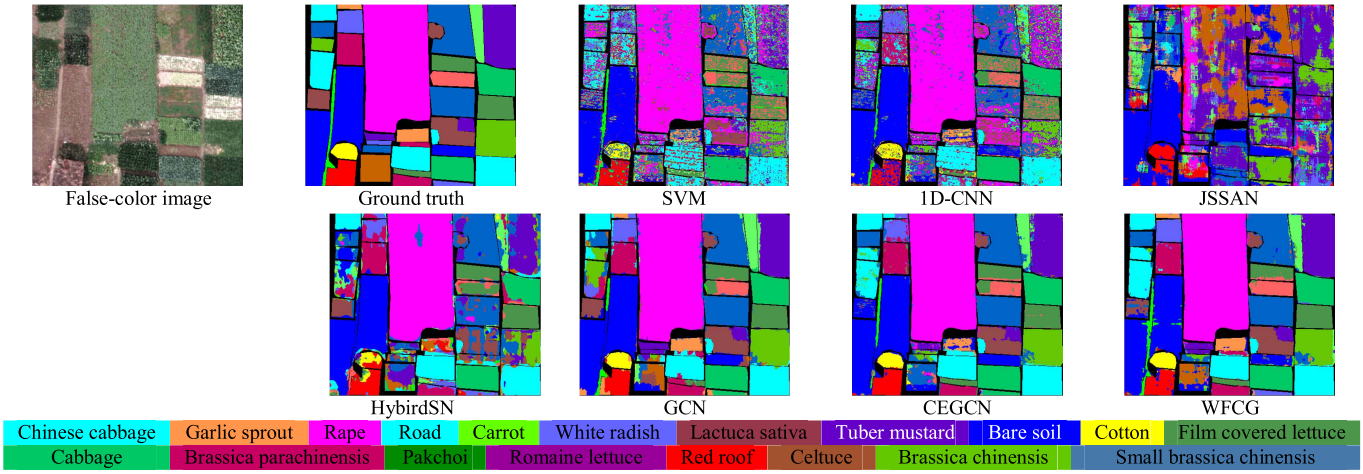

Fig. 9. False-color image, Ground truth and classification maps obtained by different methods on WHU-Hi-HongHu dataset.

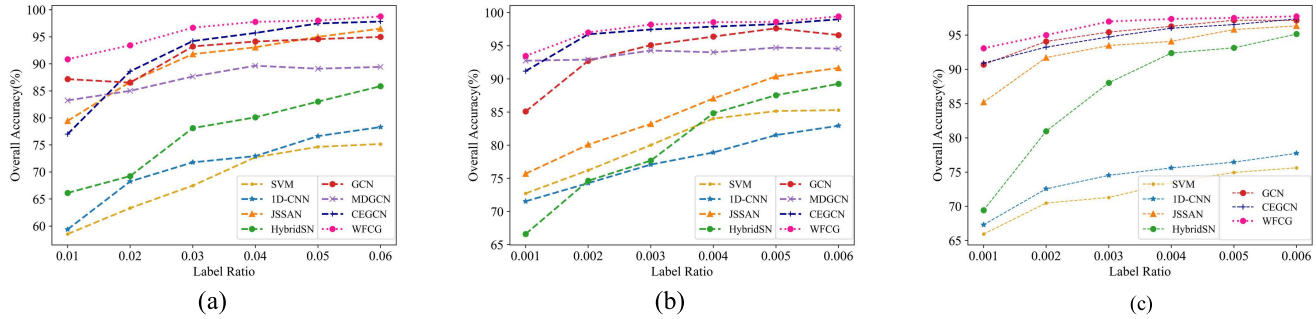

Fig. 10. The classification performance of each method with different training set ratios on three datasets. (a) India Pines. (b) University of Pavia. (c) WHU-Hi-HongHu.

D. Influence of Label Ratio

We investigate the classification accuracies of the proposed WFCG method and the other competitors under different number of labeled samples. The ratio of training samples for the Indian Pines dataset is set from 1% to 6% per class, and the ratios of training samples are both set from 0.1% to 0.6% per class for the University of Pavia dataset and the WHU-Hi-HongHu dataset. It is clear that the proposed WFCG algorithm obtains the best classification accuracy for three data sets, and the OA increases monotonically as the training samples increase, as shown in Fig. 10. Meanwhile, we observe that GCN has a higher OA when there are fewer training samples,

TABLE VIII
COMPARISON RESULTS OBTAINED BY VARIOUS METHODS ON WHU-HI-HONGHU DATASET, INCLUDING PER-CLASS ACCURACY, OA, AA, KAPPA, TRAIN TIME AND TEST TIME

	SVM	1D-CNN	JSANN	HybirdSN	GCN	MDGCN	CEGCN	WFCG
class 1	83.12	73.91	86.17	35.30	90.06	-	**91.08**	90.71
class 2	71.78	51.95	35.63	0.00	70.49	-	**89.61**	67.72
class 3	88.87	85.10	85.02	79.92	**91.83**	-	89.34	91.36
class 4	47.82	60.29	**92.36**	34.30	89.12	-	89.58	91.15
class 5	92.31	78.21	95.89	95.15	**99.07**	-	98.00	98.87
class 6	64.93	64.37	85.15	69.21	90.19	-	89.53	**90.33**
class 7	10.68	14.59	67.70	11.57	82.91	-	88.69	**94.36**
class 8	82.44	84.69	92.23	74.18	97.59	-	96.40	**98.87**
class 9	42.10	53.70	81.99	51.75	91.19	-	**95.06**	94.95
class 10	23.05	43.06	83.83	45.79	84.32	-	92.74	**94.23**
class 11	43.52	41.01	70.91	33.32	80.75	-	**85.79**	82.69
class 12	63.59	62.86	79.68	59.94	87.64	-	79.55	**93.82**
class 13	43.01	53.21	81.46	58.50	91.40	-	97.40	**96.30**
class 14	65.23	56.30	74.45	41.37	79.21	-	**90.97**	90.59
class 15	55.80	73.34	90.60	66.65	92.59		94.61	**96.22**
class 16	47.10	50.99	76.41	35.51	**88.48**	-	77.20	87.28
class 17	25.99	34.65	82.59	71.15	80.27	-	**84.41**	82.69
class 18	19.59	34.21	82.67	43.53	76.67	-	**92.15**	86.22
class 19	36.57	57.51	74.52	4.76	76.03	-	82.08	**86.63**
OA(%)	65.95	67.31	85.20	69.43	90.69	-	90.91	**93.08**
AA(%)	53.03	56.52	79.96	48.00	86.31	-	89.69	**90.26**
kappa×100	0.00	62.97	83.40	65.42	89.57	-	89.78	**92.25**
train time(s)	0.02	6.63	109.16	46.75	25.87	-	37.80	166.06
test time(s)	18.27	56.22	135.84	78.09	5.01	-	6.72	6.45

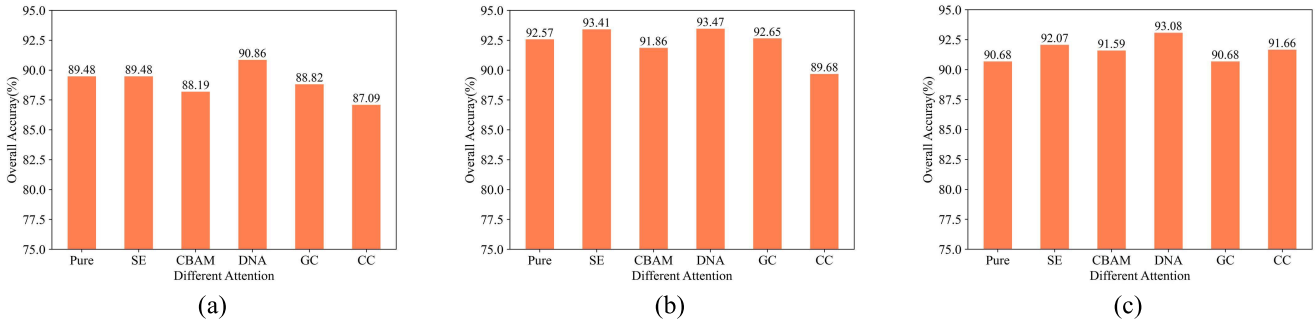

Fig. 11. The results of embedding with different attention modules on three datasets. (a) Indian Pines; (b) University of Pavia. (c) WHU-Hi-HongHu.

but it quickly reaches saturation as the number of training samples increases, while the CNN-based methods can maintain a high growth trend. When the label ratio exceeds 0.05%, JSSAN can achieve better results than GCN. This once again shows that the two models of GCN-based and CNN-based have different characteristics, suggesting that we can improve the performance of model by adopting an appropriate fusion method. The CEGCN is a method based on feature fusion, OA of which increases significantly with the increase of label ratio, indicating that its fusion method may be not suitable for small samples. The proposed WFCG algorithm outperforms the other methods, especially with small size of training samples. These results demonstrate that the reuse of different levels of feature and the design of the model structure can improve the effectiveness and stability for HSI classification.

E. Attention Mechanism

The attention mechanism can re-encode the input features, which will not significantly increase the amount of parameters, to improve the performance of the model. Respectively, we compare the changes in model accuracy when there is no attention mechanism and when the Squeeze-and-Excitation Networks (SE) [52], Convolutional Block Attention Module (CBAM) [53], Dual Attention Network (DNA) [45], global context block (GC) [54], and Criss-Cross Attention modules (CC) [55] are embedded. The experimental configuration is the same as Section III. And the corresponding results are summarized in Fig. 11.

With different attention modules, WFCG still performs well, which proves its robustness and effectiveness. By observing the results on the three data sets, we can obtain similar results. In practice, we chose the DNA model as the final attention module to embed in our framework. It is worth noting that we embed the spatial attention module and channel attention module serially into the module instead of using the parallel scheme of the original paper, which will further improve the ability of model to extract complex feature.

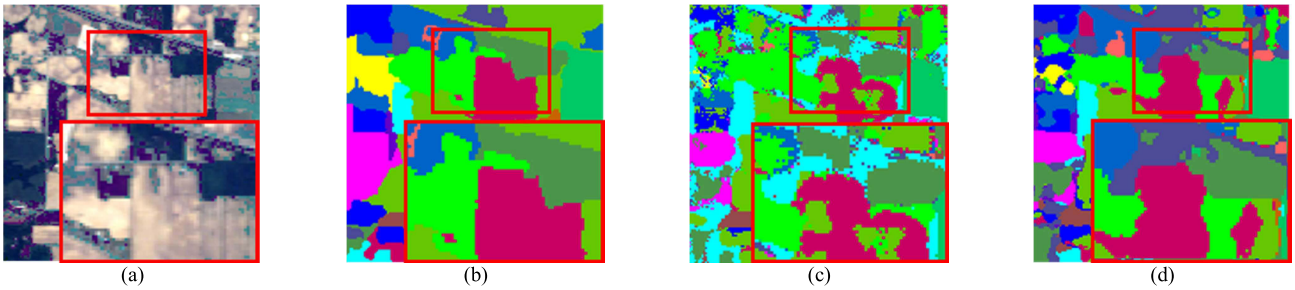

Fig. 12. Ablation visualization of WFCG model on Indian Pines dataset. (a) False-color image; (b) Visualization of all areas by branch one; (c) Visualization of all areas by branch two; (d) Visualization of all areas by WFCG.

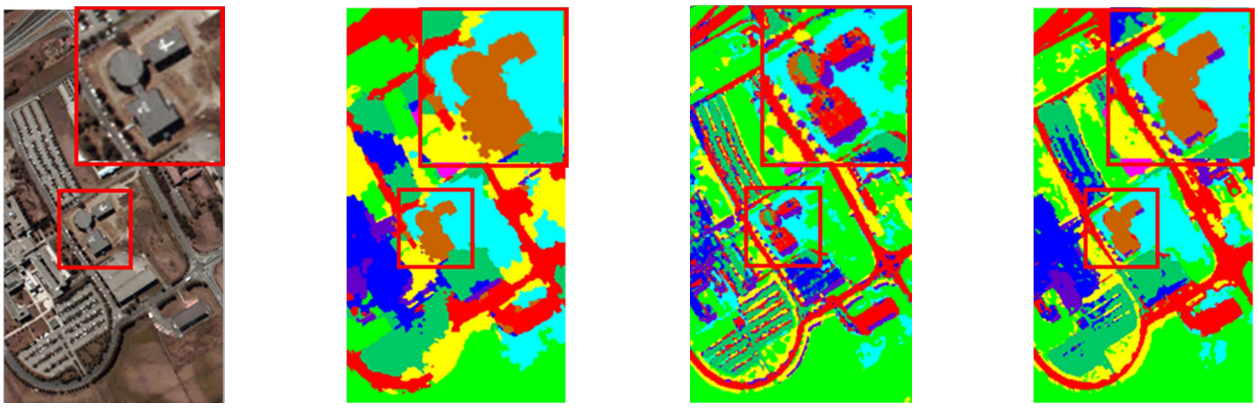

Fig. 13. Ablation visualization of WFCG model on University of Pavia dataset. (a) False-color image; (b) Visualization of all areas by branch one; (c) Visualization of all areas by branch two; (d) Visualization of all areas by WFCG.

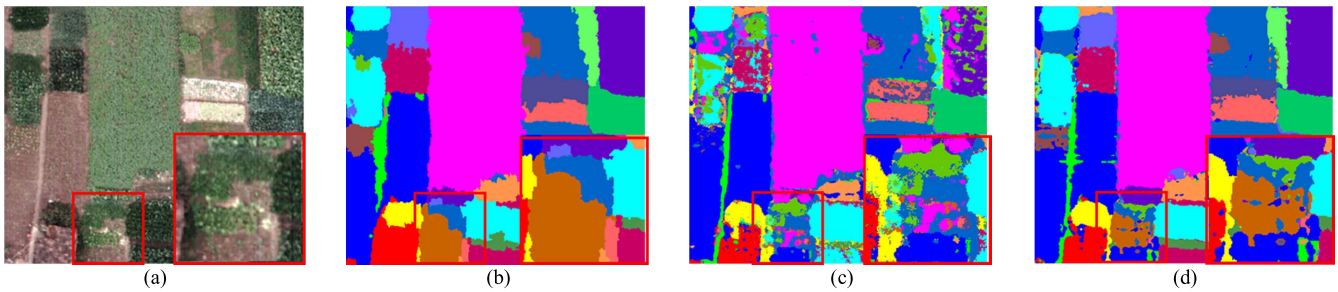

Fig. 14. Ablation visualization of WFCG model on WHU-Hi-HongHu dataset. (a) False-color image; (b) Visualization of all areas by branch one; (c) Visualization of all areas by branch two; (d) Visualization of all areas by WFCG.

F. Visualization and Analysis of Weighted Feature Fusion

The proposed WFCG algorithm consists of two very important branches, which correspond to two weak classifiers, i.e., CNN and GAT. In order to study the characteristics and fusion effects of the two weak classifiers, we visualize their classification results respectively. Figs. 12-14 show the classification maps of all regions of three data sets. Let us pay attention to the area in the red box, and we can observe that the classification maps generated by the superpixel-based GAT are smoother, but this also causes misclassification in some small areas. The pixel-based branch of CNN pays more attention to the edge information and can correctly classify some small areas, while produces more salt and pepper noise. However, after fusing them, the probability maps become more consistent with the ground truth. These results above indicate that WFCG tends to rely on the smoothed feature of GAT in large-object area, while it is more likely to use the sophisticated feature of the CNN in small-object regions,. By feature fusion, the proposed WFCG algorithm can achieve satisfying classification results on the entire HSI.

V. Conclusion

In this paper, we use an intuitive experiment to verify the characteristics of superpixel-based GNN and CNN models for HSI classification. The results show that they are complementary when the size of training samples changes. A new framework weighted feature fusion of convolutional neural network and graph attention network is proposed. In the proposed approach, each specific feature as well as the cross information between different features can be exploited to improve the discrimination capability. Our experimental results, conducted in three well-known HSI data sets, indicate that the pixel-level and superpixel-level feature exhibit great complementarity, which can be used to obtain higher classification results compared with other approaches proposed recently.

One focus of the future work will be to reduce the number of hyperparameters and perform feature fusion in an adaptive manner. In addition, we will optimize the model structure and make full use of the multi-scale and high-level abstract feature of HSI to enhance the model classification accuracy and computational efficiency with small size of training samples.

REFERENCES

[1] M. E. Paoletti, J. M. Haut, J. Plaza, and A. Plaza, "Deep learning classifiers for hyperspectral imaging: A review," *ISPRS J. Photogramm. Remote Sens.*, vol. 158, pp. 279–317, Dec. 2019.

[2] S. Liu, Q. Shi, and L. Zhang, "Few-shot hyperspectral image classification with unknown classes using multitask deep learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 6, pp. 5085–5102, Jun. 2021.

[3] S. Sabbah, "Remote sensing of gases by hyperspectral imaging: System performance and measurements," *Opt. Eng.*, vol. 51, no. 11, Jul. 2012, Art. no. 111717.

[4] D. Krupnik and S. Khan, "Close-range, ground-based hyperspectral imaging for mining applications at various scales: Review and case studies," *Earth-Sci. Rev.*, vol. 198, Nov. 2019, Art. no. 102952.

[5] J. Fan, N. Zhou, J. Peng, and Y. Gao, "Hierarchical learning of tree classifiers for large-scale plant species identification," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4172–4184, Nov. 2015.

[6] M. Shimoni, R. Haelterman, and C. Perneel, "Hypersectral imaging for military and security applications: Combining myriad processing and sensing techniques," *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 2, pp. 101–117, Jun. 2019.

[7] J. Xie, L. Fang, B. Zhang, J. Chanussot, and S. Li, "Super resolution guided deep network for land cover classification from remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, early access, Oct. 15, 2021, doi: 10.1109/TGRS.2021.3120891.

[8] Y. Dong, T. Liang, Y. Zhang, and B. Du, "Spectral–spatial weighted kernel manifold embedded distribution alignment for remote sensing image classification," *IEEE Trans. Cybern.*, vol. 51, no. 6, pp. 3185–3197, Jun. 2021.

[9] M.-L. Zhang and Z.-H. Zhou, "ML-KNN: A lazy learning approach to multi-label learning," *Pattern Recognit.*, vol. 40, no. 7, pp. 2038–2048, Jul. 2007.

[10] W. Xue, L. Zhang, X. Mou, and A. C. Bovik, "Gradient magnitude similarity deviation: A highly efficient perceptual image quality index," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 684–695, Feb. 2014.

[11] M. Belgiu and L. Drăguţ, "Random forest in remote sensing: A review of applications and future directions," *ISPRS J. Photogramm. Remote Sens.*, vol. 114, pp. 24–31, Apr. 2016.

[12] Z. Wang, B. Du, and Y. Guo, "Domain adaptation with neural embedding matching," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 7, pp. 2387–2397, Jul. 2019.

[13] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[14] Y. Dong, W. Shi, B. Du, X. Hu, and L. Zhang, "Asymmetric weighted logistic metric learning for hyperspectral target detection," *IEEE Trans. Cybern.*, early access, May 26, 2021, doi: 10.1109/TCYB.2021.3070909.

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE CVPR*, Jun. 2016, pp. 779–788.

[16] Y. Dong, B. Du, L. Zhang, and L. Zhang, "Dimensionality reduction and classification of hyperspectral images using ensemble discriminative local metric learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2509–2524, May 2017.

[17] J. Yue, L. Fang, H. Rahmani, and P. Ghamisi, "Self-supervised learning with adaptive distillation for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–13, 2021.

[18] A. Mughees and L. Tao, "Multiple deep-belief-network-based spectral-spatial classification of hyperspectral images," *Tsinghua Sci. Technol.*, vol. 24, no. 2, pp. 183–194, Apr. 2019.

[19] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2094–2107, Jun. 2014.

[20] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

[21] M. E. Paoletti, J. M. Haut, J. Plaza, and A. Plaza, "A new deep convolutional neural network for fast hyperspectral image classification," *ISPRS J. Photogramm. Remote Sens.*, vol. 145, pp. 120–147, Nov. 2018.

[22] 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.

[23] N. Wang, X. Gao, and J. Li, "Random sampling for fast face sketch synthesis," *Pattern Recognit.*, vol. 76, pp. 215–227, Apr. 2018.

[24] J. Yue, W. Zhao, S. Mao, and H. Liu, "Spectral–spatial classification of hyperspectral images using deep convolutional neural networks," *Remote Sens. Lett.*, vol. 6, no. 6, pp. 468–477, May 2015.

[25] 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.

[26] 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.

[27] L. Mou, P. Ghamisi, and X. X. Zhu, "Unsupervised spectral–spatial feature learning via deep residual conv–deconv network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 1, pp. 391–406, Jan. 2018.

[28] M. Paoletti, J. Haut, J. Plaza, and A. Plaza, "Deep&dense convolutional neural network for hyperspectral image classification," *Remote Sens.*, vol. 10, no. 9, p. 1454, Sep. 2018.

[29] Y. Gu, Q. Wang, H. Wang, D. You, and Y. Zhang, "Multiple kernel learning via low-rank nonnegative matrix factorization for classification of hyperspectral imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2739–2751, Jun. 2015.

[30] Q. S. U. Haq, L. Tao, F. Sun, and S. Yang, "A fast and robust sparse approach for hyperspectral data classification using a few labeled samples," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 6, pp. 2287–2302, Jun. 2012.

[31] J. B. Estrach, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and deep locally connected networks on graphs," in *Proc. ICL*, Apr. 2014, pp. 1–14.

[32] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," Sep. 2016, *arXiv:1609.02907*.

[33] S. Wan, C. Gong, P. Zhong, B. Du, L. Zhang, and J. Yang, "Multiscale dynamic graph convolutional network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3162–3177, May 2019.

[34] Q. Liu, L. Xiao, J. Yang, and Z. Wei, "CNN-enhanced graph convolutional network with pixel- and superpixel-level feature fusion for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 10, pp. 8657–8671, Oct. 2021.

[35] P. Veličkovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *Proc. ICL*, 2018, p. 12.

[36] A. Sha, B. Wang, X. Wu, and L. Zhang, "Semisupervised classification for hyperspectral images using graph attention networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 1, pp. 157–161, Jan. 2021.

[37] Z. Zhao, H. Wang, and X. Yu, "Spectral–spatial graph attention network for semisupervised hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2021.

[38] D. Han, J. Kim, and J. Kim, "Deep pyramidal residual networks," in *Proc. IEEE CVPR*, Jul. 2017, pp. 5927–5935.

[39] T. Wang, G. Wang, K. E. Tan, and D. Tan, "Spectral pyramid graph attention network for hyperspectral image classification," Jan. 2020, *arXiv:2001.07108*.

[40] M. Lin, Q. Chen, and S. Yan, "Network in network," Mar. 2014, *arXiv:1312.4400*.

[41] A. G. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," Apr. 2017, *arXiv:1704.04861*.

[42] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[43] B. Hanin, "Which neural net architectures give rise to exploding and vanishing gradients?" Oct. 2018, *arXiv:1801.03744*.

[44] N. He, L. Fang, and A. Plaza, "Hybrid first and second order attention UNet for building segmentation in remote sensing images," *Sci. China Inf. Sci.*, vol. 63, no. 4, pp. 1–12, Apr. 2020.

[45] J. Fu *et al.*, "Dual attention network for scene segmentation," in *Proc. IEEE CVPR*, Jun. 2019, pp. 3146–3154.

[46] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," in *Proc. ICML*, 2019, pp. 7354–7363.

[47] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 6232–6251, Oct. 2016.
[48] Y. Zhong, X. Hu, C. Luo, X. Wang, J. Zhao, and L. Zhang, "WHU-Hi: UAV-borne hyperspectral with high spatial resolution (H^2) benchmark datasets and classifier for precise crop identification based on deep convolutional neural network with CRF," *Remote Sens. Environ.*, vol. 250, Dec. 2020, Art. no. 112012.
[49] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *J. Sensors*, vol. 2015, pp. 1–12, Jan. 2015.
[50] L. Li, J. Yin, X. Jia, S. Li, and B. Han, "Joint spatial–spectral attention network for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 10, pp. 1816–1820, Oct. 2021.
[51] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[52] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE CVPR*, Jun. 2018, pp. 7132–7141.
[53] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. IEEE ECCV*, Sep. 2018, pp. 3–19.
[54] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, "GCNet: Non-local networks meet squeeze-excitation networks and beyond," in *Proc. IEEE ICCV*, Oct. 2019, pp. 1–10.
[55] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. IEEE ICCV*, Oct. 2019, pp. 603–612.

Yanni Dong (Member, IEEE) received the B.S. degree in sciences and techniques of remote sensing from Wuhan University, Wuhan, China, in 2012, and the Ph.D. degree in photogrammetry and remote sensing from the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, in 2017.

She is currently an Associate Professor with the Institute of Geophysics and Geomatics, China University of Geosciences, Wuhan. She was a Hong Kong Scholar with the Department of Land Surveying and Geo-Informatics, The Hong Kong Polytechnic University, Hong Kong. Her current research interests include hyperspectral image processing, pattern recognition, and machine learning.

Dr. Dong serves as a Reviewer of more than 30 international journals, including the IEEE TRANSACTIONS ON CYBERNETICS, IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON MULTIMEDIA, *PR*, and IEEE GEOSCIENCE AND REMOTE SENSING LETTERS. She regularly serves as a PC Member for IJCAI and AAAI.

Quanwei Liu received the B.S. degree in exploration technology and engineering from the Henan University of Engineering, Zhengzhou, China, in 2020. He is currently a Graduate Student with the Institute of Geophysics and Geomatics, China University of Geosciences, Wuhan, China.

His research interests include hyperspectral images processing, machine learning, and graph neural networks.

Bo Du (Senior Member, IEEE) received the Ph.D. degree in photogrammetry and remote sensing from the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China, in 2010.

He is currently a Professor with the School of Computer Science and Institute of Artificial Intelligence, Wuhan University. He is also the Director of the National Engineering Research Center for Multimedia Software, Wuhan University. He has more than 80 research papers published in the IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON CYBERNETICS, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, IEEE JOURNAL OF SELECTED TOPICS IN EARTH OBSERVATIONS AND APPLIED REMOTE SENSING, and IEEE GEOSCIENCE AND REMOTE SENSING LETTERS. His 13 ESI hot papers or highly cited papers. His major research interests include pattern recognition, hyperspectral image processing, and signal processing.

Dr. Du regularly serves as a Senior PC Member for IJCAI and AAAI. He served as the Area Chair for ICPR. He won the Highly Cited Researcher 2019 by the Web of Science Group. He also won the International Joint Conferences on Artificial Intelligence (IJCAI) Distinguished Paper Prize, the IEEE Data Fusion Contest Champion, and the IEEE Workshop on Hyperspectral Image and Signal Processing Best paper Award in 2018. He serves as an Associate Editor for *Neural Networks*, *Pattern Recognition*, and *Neurocomputing*. He also serves as a Reviewer for 20 Science Citation Index (SCI) magazines, including IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON CYBERNETICS, IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE JOURNAL OF SELECTED TOPICS IN EARTH OBSERVATIONS AND APPLIED REMOTE SENSING, and IEEE GEOSCIENCE AND REMOTE SENSING LETTERS.

Liangpei Zhang (Fellow, IEEE) received the B.S. degree in physics from Hunan Normal University, Changsha, China, in 1982, the M.S. degree in optics from the Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, China, in 1988, and the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 1998.

He is currently a Chair Professor with the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing (LIESMARS), Wuhan University. He was a Principal Scientist of the China State Key Basic Research Project (2011–2016) appointed by the Ministry of National Science and Technology of China to lead the remote sensing program in China. He has published more than 700 research papers and five books. He is the Institute for Scientific Information (ISI) highly cited author. He is the holder of 30 patents. His research interests include hyperspectral remote sensing, high-resolution remote sensing, image processing, and artificial intelligence.

Dr. Zhang is a fellow of the Institution of Engineering and Technology (IET). He was a recipient of the 2010 Best Paper Boeing Award, the 2013 Best Paper ERDAS Award from the American Society of Photogrammetry and Remote Sensing (ASPRS), and the 2016 Best Paper Theoretical Innovation Award from the International Society for Optics and Photonics (SPIE). His research teams won the top three prizes of the IEEE GRSS 2014 Data Fusion Contest and his students have been selected as the winners or finalists of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS) student paper contest in recent years. He is the Founding Chair of the IEEE Geoscience and Remote Sensing Society (GRSS) Wuhan Chapter. He also serves as an associate editor or an editor for more than ten international journals. He serves as an Associate Editor for the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING.