Neural Networks 168 (2023) 105–122

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

CEGAT: A CNN and enhanced-GAT based on key sample selection strategy for hyperspectral image classification

Cuiping Shi [a,*], Haiyang Wu [a], Liguo Wang [b]

[a] College of Communication and Electronic Engineering, Qiqihar University, Qiqihar 161000, China
[b] College of Information and Communication Engineering, Dalian Nationalities University, Dalian 116000, China

ARTICLE INFO

Article history:
Received 13 February 2023
Received in revised form 16 May 2023
Accepted 31 August 2023
Available online 17 September 2023

Keywords:
Deep learning
Convolution neural network
Graph convolution network
Hyperspectral image classification

ABSTRACT

In recent years, the application of convolutional neural networks (CNNs) and graph convolutional networks (GCNs) in hyperspectral image classification (HSIC) has achieved remarkable results. However, the limited label samples are still a major challenge when using CNN and GCN to classify hyperspectral images. In order to alleviate this problem, a double branch fusion network of CNN and enhanced graph attention network (CEGAT) based on key sample selection strategy is proposed. First, a linear discrimination of spectral inter-class slices (LD_SICS) module is designed to eliminate spectral redundancy of HSIs. Then, a spatial spectral correlation attention (SSCA) module is proposed, which can extract and assign attention weight to the spatial and spectral correlation features. On the graph attention (GAT) branch, the HSI is segmented into some super pixels as input to reduce the amount of network parameters. In addition, an enhanced graph attention (EGAT) module is constructed to enhance the relationship between nodes. Finally, a key sample selection (KSS) strategy is proposed to enable the network to achieve better classification performance with few labeled samples. Compared with other state-of-the-art methods, CEGAT has better classification performance under limited label samples.

1. Introduction

Different from ordinary images, hyperspectral images (HSIs) not only have rich spatial texture information, but also each pixel of HSIs contains hundreds of continuous spectral bands (Wang, Yan, Mu and Huang, 2020; Zhang & Zhang, 2022). This provides a strong guarantee for HSIs to be able to classify at the pixel level. In recent years, HSIs have been widely used in medical, national defense, agriculture and other fields (Wan, Fan, & Jin, 2021; Wang, Yang and Zhang, 2020; Xing et al., 2019; Yu, Kong, Wang, Liu, & Liu, 2020; Zhang, Song, Du, & Zhang, 2021). All these applications need to be based on the accurate classification of HSIs.

In order to achieve accurate classification of HSIs, a number of excellent methods have emerged. In the early days, some methods based on machine learning were proposed. For example, sparse representation classification (SRC) (Chen, Nasrabadi, & Tran, 2010) and kernel support vector machine (KSVM) (Kuo, Ho, Li, Hung, & Taur, 2014). The former pays more attention to sparse feature representation learning, while the latter pays more attention to classifier design. They are all methods based on pixel classification, but only focus on spectral information, ignoring the importance of spatial information, so that the performance of hyperspectral image classification is limited. In addition, traditional machine learning relies on manual features and parameter experience (Yaling, Xiwu, Hui, & Yurong, 2021), which makes such classification methods less robust and generalization. In recent years, the excellent performance of deep learning (DL) in hyperspectral image classification (HSIC) task has attracted extensive attention (Li et al., 2019). Deep learning avoids the complicated manual feature process of machine learning, and can extract the features of samples for classification by training a part of samples. Among them, the performances of convolutional neural networks (CNNs) and graph convolutional networks (GCNs) (Ahmad et al., 2022) are particularly impressive.

In the past few decades, CNNs have been one of the most widely used methods in HSIC. For example, in order to make full use of the spatial spectral information of HSIs, a spectral spatial residual network (SSRN) was proposed (Zhong, Li, Luo, & Chapman, 2018). SSRN uses 3-D convolution kernel (Ying, Zhang, & Qiang, 2017) as a feature extraction tool, which can adapt to the three-dimensional data structure of HSIs and directly extract spatial spectral features. In addition, SSRN uses residual structure to connect the network, which can expand the depth of the network. Subsequently, a fast dense spectral spatial convolution (FDSSC) was proposed (Wang, Dou, Jiang, & Sun, 2018) to further extract the deep information of HSIs. Similar to SSRN, FDSSC

* Corresponding author.
E-mail addresses: shicuiping@qqhru.edu.cn (C. Shi), 2021910323@qqhru.edu.cn (H. Wu), wangliguo@hrbeu.edu.cn (L. Wang).

https://doi.org/10.1016/j.neunet.2023.08.059
0893-6080/

also uses 3-D convolution kernel to directly extract the spatial spectral information of HSIs. The difference is that FDSSC uses a dense connection structure and can reuse the same feature. However, this method requires a large number of calculations and parameters, which is not friendly to conventional computing resources and is not easy to be widely used in practice. Therefore, a stepwise quantization method is proposed (Mei, Chen, Zhang, Li and Plaza, 2022). Although this method is slightly insufficient in classification accuracy, it can save a lot of calculation costs and is more conducive to wide application. In addition, the small sample problem is also a major challenge in CNN. The method proposed in Zhang et al. (2023) has brought some inspiration to alleviate this problem. In recent years, an attention mechanism has been developed based on human vision (Haut, Paoletti, Plaza, Plaza, & Li, 2019; Mei et al., 2019; Sun, Zheng, Lu, & Wu, 2020; Woo, Park, Lee, & Kweon, 2018; Zheng, Wang, Du and Lu, 2022). The attention mechanism can extract features with more discriminating ability, avoiding the interference of irrelevant features on classification. In Ma, Yang, Wu, Zhao, and Zhang (2019), a double branch multi attention (DBMA) network was proposed. In order to avoid the interaction between spectral and spatial features, DBMA uses double branch structure to extract spatial spectral features respectively. In addition, different attention is used on the two branches. In order to further improve the accuracy of HSIC, a double branch dual attention (DBDA) method was proposed (Li, Zheng, Duan, Yang, & Wang, 2020). Compared with DBMA, DBDA achieves higher classification accuracy with fewer parameters. However, in CNN based attention networks, the size of convolutional kernel is extremely sensitive to spatial rotation. Therefore, a rotation invariant attention network (RIA) was proposed (Zheng, Sun, Lu and Xie, 2022). RIA can extract rotation invariant spatial spectral features, which alleviates the impact of convolution size on classification results. In order to explore the pre and post dependence of spectral bands, in Mei, Li, Liu, Cai and Du (2022), a bidirectional long short-term memory (Bi-LSTM) network based on attention is proposed, which further improves the accuracy of HSIC. Although CNN based methods have achieved some gratifying results, there are still some bottlenecks in the classification of hyperspectral images. Including:

(1) Using regular convolution kernels to extract the features of HSIs with complex ground distribution tends to weaken the context dependency, thereby limiting the classification performance of the network.
(2) CNNs can only process Euclidean space data effectively, ignoring the internal correlation between ground objects in HSIs.
(3) The classification performance of CNNs for HSIs still depends on a large number of labeled training samples.

In recent years, with the rise of GCNs, these problems have been alleviated to some extent. GCNs can directly handle the graph structure, and can better capture the internal relationship between adjacent objects of HSIs. For example, in Mou, Lu, Li, and Zhu (2020), a non local graph convolution (nonlocal GCN) was proposed. This method directly processes the entire hyperspectral image using graph convolution to capture non local features of the hyperspectral image and model the internal relationships between ground objects. However, the method based on GCNs also has some shortcomings. First of all, directly using each pixel of the whole image as a graph node requires a lot of calculations. Secondly, the weight between adjacent nodes in conventional GCNs is fixed, which greatly limits the classification performance of the network. In order to avoid using each pixel of the HSI as the input node directly, resulting in excessive network computation, some superpixel based GCN methods have been proposed. For example, a multiscale dynamic graph convolution (MDGCN) was proposed in Wan et al. (2020). It uses simple linear iterative clustering (SLIC) to segment HSI into superpixels and use them as input nodes for GCN. Specifically, in Wan et al. (2020), multi-scale SLIC were used to segment hyperspectral images to obtain multi-scale feature information of HSI. In addition, some graph attentional network (GAT) based methods have been proposed to dynamically update node features. In Wang, Wang, Tan and Tan (2020), a spectral pyramid graph attention network is proposed. spectral pyramid have been used to encode contextual information of spectra and have achieved good classification results. In Sha, Wang, Wu, and Zhang (2021) and Zhao, Wang, and Yu (2021), GAT is used for semi-supervised classification. And they are all classified by assigning different weights to spatial spectral features through attention. Subsequently, in order to further improve the classification accuracy of HSIs. Some methods utilize a hybrid framework of GCN and CNN to extract pixel level and superpixel features. A CNN enhanced GCN (CEGCN) is proposed (Liu, Xiao, Yang, & Wei, 2021). First, HSI is segmented and encoded into a graph structure through simple linear iterative clustering (SLIC), and then GCN is used to capture the super pixel level features of HSI, and combined with the pixel level features extracted by CNN. Under the premise of ensuring the classification performance, the network parameters are greatly reduced. Subsequently, the weighted feature fusion of convolutional neural network and graph attentional network (WFCG) is proposed (Dong, Liu, Du, & Zhang, 2022). Different weights are given to nodes through graph attentional network (GAT) and updated through training. In addition, considering the interference of redundant pixels on GCN classification performance, a spatial pooling graph convolutional network (SPGCN) was proposed in Zhang et al. (2022). SPGCN has developed a spatial pooling method and used GCN to extract the spatial topological features of HSI, and SPGCN has been proved to be superior to some CNNs based methods.

Although these methods have alleviated some problems of CNNs and GCNs, there are still some challenges. Among them, the problem of small sample is still prominent. With limited label samples, the precision of HSIC will still be greatly limited. Therefore, in this paper, a fusion network of CNN and enhanced GAT (CEGAT) is proposed. Specifically, in order to avoid the interference of redundant information, a linear discrimination of spectral inter class slices (LD_SICS) module is first designed to eliminate the spectral redundant information and map the feature information to the low dimensional space, which is conducive to subsequent feature extraction. Then, a dual branch network framework integrating GAT and CNN was proposed. Among them, an enhanced graph attention network (EGAT) module is constructed to enhance the representation ability of GAT. In addition, in order to extract pixel level spatial spectral features with more discriminative ability and reduce the computational complexity of the network, a spatial spectral correlation attention (SSCA) module is proposed. In order to make full use of the feature information of HSI, this paper adopts the structure of super pixel coding graph and uses CNN to extract pixel level spatial spectral features. Finally, a key sample selection (KSS) strategy is proposed, which can effectively improve the accuracy of classification.

The main contributions of this paper include the following three parts:

(1) To enhance the attention capacity of the GAT, an EGAT module was proposed. Through the enhancement of GAT, the network can capture the node features with more discriminating ability in HSI, which improves the feature expression ability of the network.
(2) In order to capture the pixel level spatial spectral features of HSIs, a SSCA module is designed. SSCA module can

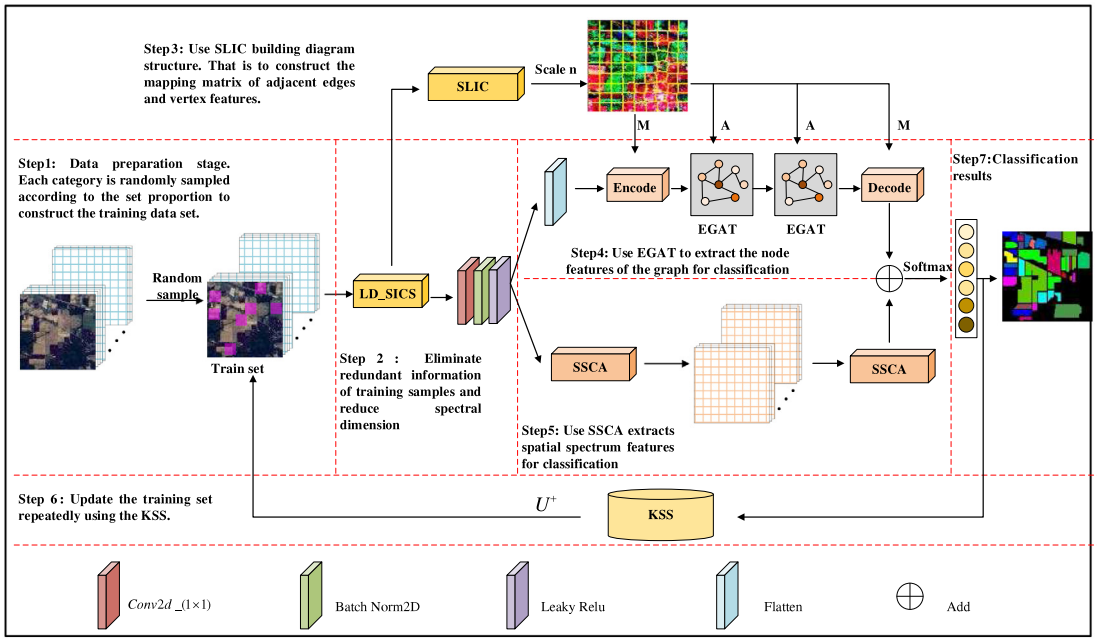


Fig. 1. The overall framework of CEGAT. (Where, M is the mapping matrix of the construction node. A is the adjacent edge matrix. U^+ means adding samples selected by KSS to the training set.)

reduce the information loss during feature extraction, and effectively capture important features with more discriminating ability. In addition, the SSCA module can reduce the computational complexity of the network to a certain extent.

(3) A KSS strategy is proposed. By iteratively selecting unlabeled samples that are more difficult to distinguish from the sample pool, KSS adds them to the training, thus improving the discriminative ability of the model. This effectively alleviates the sensitivity of the network to the small sample problem.

The rest of this paper is arranged as follows: In Section 2, the CEGAT is discussed in detail. Section 3 first introduces the datasets used in the experiment and the experimental parameter settings, then conducts some ablation experiments with the proposed module, and finally verifies the effectiveness of CEGAT. In Section 4, the conclusions are given.

2. Methodology

In order to mitigate the impact of small samples on the classification performance of HSIs, an effective CEGAT is proposed in this paper. First, the HSI data set is divided into training set, verification set and testing set, which can be recorded as $x_t \in \mathbb{R}^{h\times w\times b}$, $x_y \in \mathbb{R}^{h\times w\times b}$ and $x_{te} \in \mathbb{R}^{h\times w\times b}$ in turn. Where h, w and b represent the height, width and number of spectral bands, respectively.

2.1. The overall structure of CEGAT

The overall structure of CEGAT is shown in Fig. 1. The whole process can be divided into seven parts. First, the original HSI is divided into training set, verification set and testing set by random selecting sampling. In particular, the samples in the training set were randomly selected from each category with predetermined ratio. Next, a LD_SICS module is designed to eliminate the spectral redundancy information of training samples, and to project the spectral information into a low dimensional space more conducive to discrimination. This process can be expressed as

$$X_{in} = LDA\left(sice2_{-1}\left(C.V\left(Norm\left(sice1_{-1}\left(T\left(x_t\right)\right)\right)\right)\right)\right)\ x_t \in \mathbb{R}^{h\times w\times b}$$
$$X_{in} \in \mathbb{R}^{h\times w\times b'} \quad (1)$$

where, $T(\cdot)$ means flattening the spatial dimension of the training set. $sice1_{-1}(\cdot)$ is to slice the continuous spectral bands, and $sice2_{-1}(\cdot)$ is to slice the coefficient of variation of spectral bands. $Norm(\cdot)$ is the normalized function. $C.V(\cdot)$ represents the operation of calculating the coefficient of variation. $LDA(\cdot)$ is a linear discriminate function. X_{in} represents the data after removing the spectral redundancy information. LD_SICS is based on our previous work (Shi, Wu, & Wang, 2023) using linear discriminant analysis (LDA). Specifically, LD_ SICS first slices the spectral bands, then calculates their coefficients of variation and compares them, and uses slices to preserve discriminative spectral bands (i.e., spectral bands with high coefficient of variation). Finally, the linear discriminate function is used to project the spectral information into a lower dimensional space that is easier to distinguish.

GAT directly acts on the graph. If the whole HSI is directly used as the input, the network will generate a large number of parameters, thus reducing the convergence speed of the network. Therefore, in the step 3, X_{in} is divided into different super pixel blocks using SLIC. In addition, in order to establish a close relationship between the super pixel level features extracted by GAT and the pixel level features extracted by CNN, an encoding method in Liu et al. (2021) is adopted to construct graph structure $G = (X_V, A)$, where X_V is the node of the graph and $A \in \mathbb{R}^{N\times N}$ is the adjacency matrix. First, the vertex features of the graph are constructed through a mapping matrix $M \in \mathbb{R}^{(h\times w)\times N}$. This process can be represented as

$$X_V = M \times Flatten\left(LR\left(B\left(F\left(X_{in}\right)\right)\right)\right) \quad X_V \in \mathbb{R}^{N\times C} \quad (2)$$

where, $F(\cdot)$ is the two-dimensional convolution kernel of 1×1, and its channel number is C. $B(\cdot)$ is the batch normalization function. $LR(\cdot)$ is the activation function LeakyRelu. $Flatten(\cdot)$ represents the operation of flattening the spatial dimension. In

particular, the construction process of mapping matrix $M \in \mathbb{R}^{(h\times w)\times N}$ can be expressed as

$$M_{(i,j)} = \begin{cases} 1, Flatten\left(LR\left(B\left(F\left(X_{in}\right)\right)\right)\right)_i \in H_j \\ 0, \text{ other} \end{cases} \tag{3}$$

where, $M_{(i,j)}$ represents the element in row i and column j of M. $Flatten\left(LR\left(B\left(F\left(X_{in}\right)\right)\right)\right)_i$ represents the feature value of the ith pixel, and H_j represents the jth super pixel. In addition, the encoding of the adjacency matrix $A \in \mathbb{R}^{N\times N}$ can be expressed as

$$A_{(i,j)} = \begin{cases} 1, H_j \text{ and } H_i \text{ are adjancent} \\ 0, \text{ other} \end{cases} \tag{4}$$

where, $A_{(i,j)}$ represents the element in row i and column j of A. H_i is the ith super pixel. After coding, go to step 4. EGAT captures image features of the super pixel level and performs the inverse transform through the decoder, which is more conducive to feature fusion at the final stage. In the step 5, pixel-level spatial spectral features are extracted by the CNN branch. Specifically, on the CNN branch, an SSCA module is proposed. By using the SSCA module, the information loss during the feature extraction process can be reduced, and important features with higher recognition ability can be effectively captured. In addition, the computational complexity of the network can be reduced to a certain extent. And then the features extracted from the two branches are fused. Then, the early stop mechanism is used to judge whether the model is pre convergent, and the step 6 is carried out when the model is pre convergent. In the step 6, the KSS strategy is proposed to select unlabeled samples that are difficult to distinguish in the validation set and add them to the training set for training again. Finally, classify in step 7.

2.2. EGAT module

Conventional GCNs deal with graph structure in a way that it is difficult to assign different weights to nodes, which greatly limits the representation ability of the network. To solve this problem, this paper proposes an EGAT module, which structure is shown in Fig. 2. EGAT captures the important features of the graph by enhancing the discriminate ability of features, so as to enhance the representation ability of the network. Specifically, the vector set of node features can be expressed as

$$X_V = \left\{\vec{X}_{V1}, \vec{X}_{V2}, \ldots, \vec{X}_{VN}\right\} \quad \vec{X}_{Vi} \in \mathbb{R}^C \tag{5}$$

In order to improve the representation ability of features, each node vector is linearly transformed and the attention correlation coefficient between nodes is calculated. This process can be expressed as

$$X_{eij} = Atten\left(W\vec{X}_{Vi}, W\vec{X}_{Vj}\right) \tag{6}$$

where, X_{eij} is the obtained attention correlation coefficient, which represents the correlation between node $\vec{X}_{Vi}$ and $\vec{X}_{Vj}$. W represents a linear transformation operation. $A(\cdot)$ indicates a shared attention mechanism. In particular, in order to avoid breaking the structural features of the graph and reduce the computational complexity, only the first order attention is focused here (i.e. attention between first-order adjacent nodes). In order to enhance the ability of distinguishing features, two groups of X_{eij} are calculated and the values of attention coefficient are mapped to 0–1. Then, the two groups of attention coefficients are multiplied to get X'_{eij}. After summation by column, divide by the number of features each node C to obtain the normalization coefficient ϖ_0 after attention enhancement. Finally, we divide X'_{eij} by ϖ_0. This process can be expressed as

$$X'_{eij} = sigmoid1\left(X_{eij}\right) \times sigmoid2\left(X_{eij}\right) \tag{7}$$

$$X''_{eij} = \frac{X'_{eij}}{SUM_{-1}\left(X'_{eij}\right)/C} \tag{8}$$

In Eq. (7), $sigmoid1\left(X_{eij}\right)$ and $sigmoid1\left(X_{eij}\right)$ represent two groups of normalized attention coefficients respectively. In Eq. (8), $SUM_{-1}(\cdot)$ represents the operation of summing by column. X''_{eij} is the final enhanced attention coefficient. Then the attention weights of different nodes are obtained through *soft* max. The whole process can be expressed as

$$X_{att} = softamx\left(X''_{eij}\right) = \frac{\exp\left(X''_{eij}\right)}{\sum_{k\in N_i}\exp\left(X''_{eik}\right)} \tag{9}$$

where N_i is the neighbor set of the ith node. X_{att} represents the final enhanced attention weight. In order to output attention stably, a multi head mechanism is added to EGAT. Specifically, the attention weight X_{att} obtained is embedded into the node features, and then the process is repeated n times, and then the results obtained n times are connected to obtain the n header EGAT. This process can be expressed as

$$X_{EGAT} = \|_1^n \sigma\left(X_{att} W X_V\right) \tag{10}$$

where, $\|$ is the connection operation. $\sigma(\cdot)$ is a nonlinear layer function, and X_{EGAT} represents the output result of EGAT.

In the proposed CEGAT network, two layers of EGAT are used to extract super pixel level features. On the whole, EGAT can capture more important node features and suppress the interference of redundant features by strengthening the conventional GAT and using the correlation of first-order adjacent nodes without destroying the structural features of the graph.

2.3. SSCA module

There is a lot of spectral redundancy information in HSIs. Although the proposed CEGAT has passed the LD_ SICS eliminates most of the spectral redundancy, but there will still be some redundant information in the process of feature extraction. This problem can be further alleviated by extracting the spatial spectral correlation features of HSIs. Inspired by Zhong, Li, Ma, Li, and Zheng (2022), a novel SSCA module was designed. As shown in Fig. 3, first represent the input data as $X'_{in} \in \mathbb{R}^{C\times H\times W}$ and flatten its spatial dimension. In addition, in order to reduce information loss, SSCA directly uses linear transformation to build higher level features. The calculation process of linear transformation can be expressed as

$$K = \left[W_K\left(reshape\left(X'_{in}\right)\right)\right]^{\mathrm{T}} X'_{in} \in \mathbb{R}^{C\times H\times W} \quad K \in \mathbb{R}^{S\times C} \tag{11}$$

$$Q = W_Q\left(reshape\left(X'_{in}\right)\right) X'_{in} \in \mathbb{R}^{C\times H\times W} \quad Q \in \mathbb{R}^{C\times S} \tag{12}$$

$$V = W_V\left(reshape\left(X'_{in}\right)\right) X'_{in} \in \mathbb{R}^{C\times H\times W} \quad V \in \mathbb{R}^{C\times S} \tag{13}$$

where, $S = H \times W$. $W_K(\cdot)$, $W_Q(\cdot)$ and $W_V(\cdot)$ are three linear layer functions respectively. Subsequently, in order to reduce the computational complexity of the network, and establish the spatial and spectral correlation. We use multi-scale pyramid pooling (He, Zhang, Ren, & Sun, 2015) to deal with $K \in \mathbb{R}^{S\times C}$ and $V \in \mathbb{R}^{C\times S}$. This process can be expressed as

$$K' = \left[AAP_1(K) \parallel AAP_2(K) \parallel AAP_4(K) \parallel AAP_8(K)\right] \tag{14}$$

$$V' = \left[AAP_1(V) \parallel AAP_2(V) \parallel AAP_4(V) \parallel AAP_8(V)\right] \tag{15}$$

Here, $AAP_1(\cdot)$, $AAP_2(\cdot)$, $AAP_4(\cdot)$, and $AAP_8(\cdot)$ represent adaptive average pooling with size 1, 2, 4, and 8, respectively. Then, the correlation between K' and Q is used to calculate the attention mask of spatial spectral correlation. The process is

$$SSA = Soft\max_{-1}\left(K' \times Q\right) \quad SSA \in \mathbb{R}^{S\times 15} \tag{16}$$

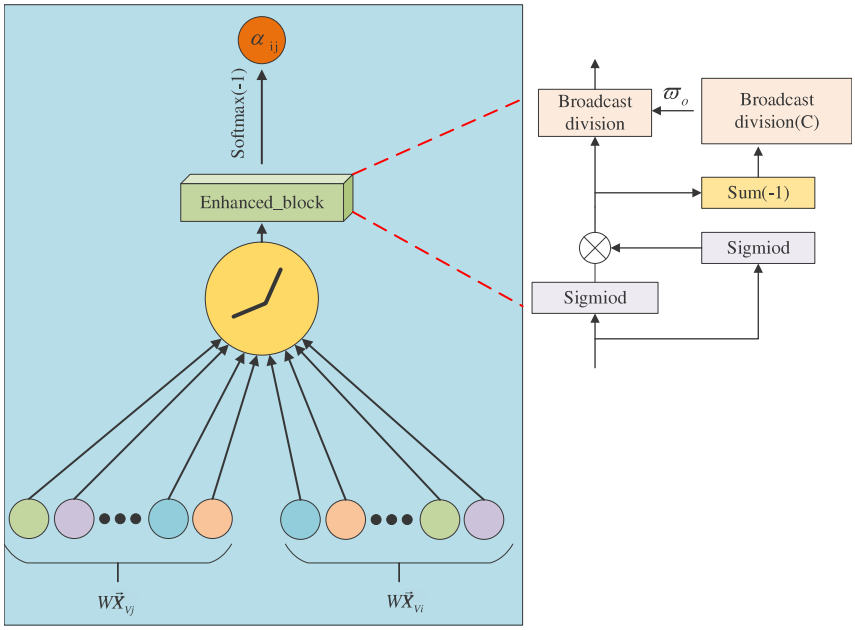


Fig. 2. EGAT module. (Where ⟋ represents the activation function. ⊗ represents the multiplication operation.)

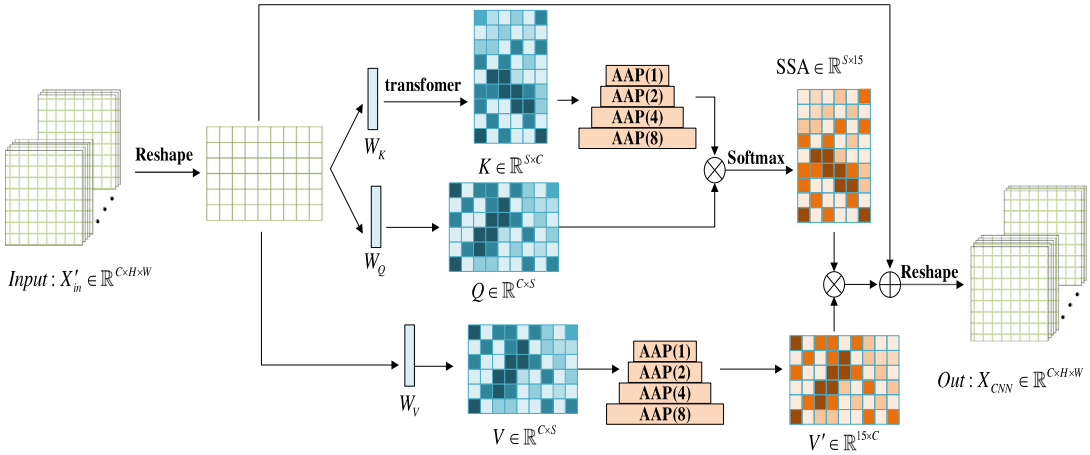


Fig. 3. SSCA module. (AAP (1), AAP (2), AAP (4) and AAP (8) represent the adaptive average pooling of sizes 1, 2, 4 and 8, respectively.)

where, $Soft\ \max_{-1}(\cdot)$ represents Softmax operations by column. Finally, the attention mask of spatial spectral correlation is used to weight V' to obtain the attention feature map. In addition, residual connection is introduced in SSCA to prevent network over fitting. The calculation of SSCA module can be simplified as

$$X_{CNN} = V' \times SSA + reshape\left(X'_{in}\right) \tag{17}$$

In a word, the spatial and spectral information can be correlated through the multi-scale pooling of SSCA module, and the information loss in the feature extraction process is reduced. In addition, compared with the method in Zhong et al. (2022) which directly uses $K \in \mathbb{R}^{S\times C}$ and $Q \in \mathbb{R}^{C\times S}$ dot product to calculate the attention mask, the SSCA proposed in this paper can effectively reduce the computational complexity of the network. Specifically, the complexity of SSCA can be expressed as

$$O = o_1\left(S \times C^2 \times 15\right) + o_2\left(S \times C \times 15^2\right) \tag{18}$$

The computational complexity of directly multiplying $K \in \mathbb{R}^{S\times C}$ and $Q \in \mathbb{R}^{C\times S}$ points can be expressed as

$$O' = o_1\left(S^2 \times C^2\right) + o_2\left(S^3 \times C\right) \tag{19}$$

where, o_1 and o_2 are the computational complexity of the attention mask and the computational complexity of the attention feature map, respectively. Obviously, the computational complexity of the attention mask is reduced by $S/15$ times through SSCA. And the computational complexity of attention feature map is reduced by $S^2/15^2$ times.

2.4. KSS module

The influence of small samples has always been the main bottleneck limiting the improvement of hyperspectral image classification performance. Because the feature information that can be extracted by the network is limited under the limited label training samples, which will directly affect the final classification accuracy. In addition, each sample contains different amount of discriminate feature information. Therefore, the training set consisting of randomly selected samples will also have a large difference in the amount of feature information. This will make the classification accuracy of the network through training fluctuate greatly, which affects the stability of the network. In order

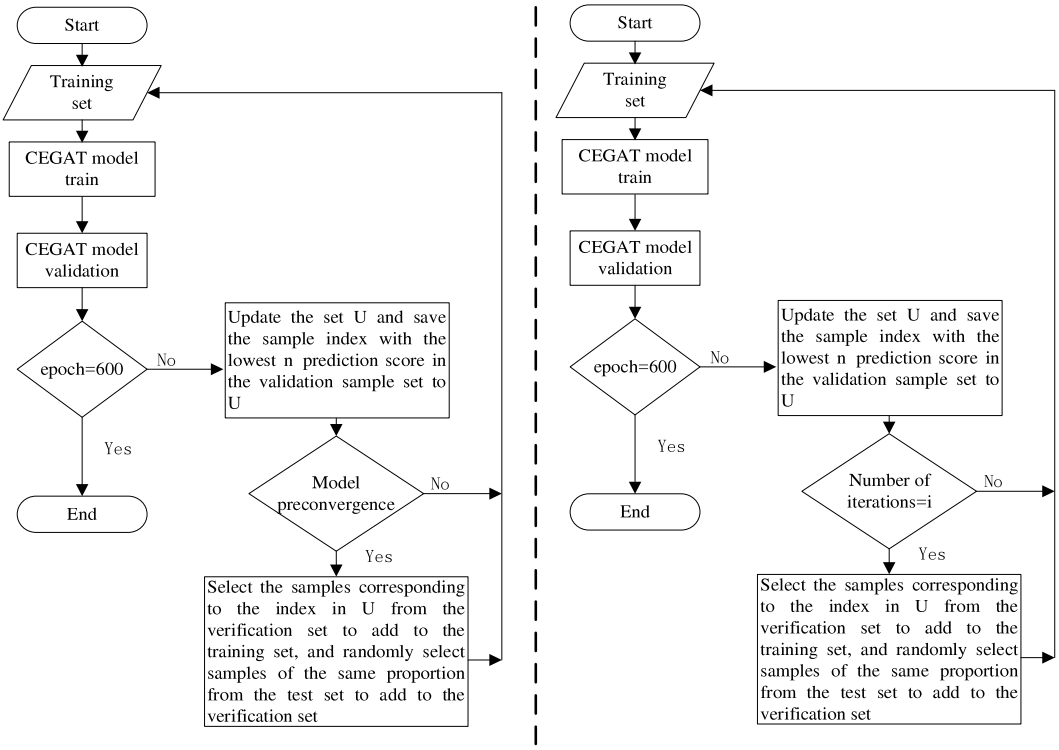


Fig. 4. The process of the KSS strategy.

to alleviate this problem, a KSS strategy is proposed in this paper. The processing of the KSS strategy is shown in Fig. 4.

First, take an epoch as an example, as shown in the left part of the dotted line in Fig. 4. The training set $x_t \in \mathbb{R}^{h\times w\times b}$ is trained by the proposed CEGAT, and the model parameters are saved. Then the saved model parameters are used by the network to verify set $x_y \in \mathbb{R}^{h\times w\times b}$. After the verification, the network will judge whether the training is over (i.e. epoch = 600). If the training is not completed, the index of 10 more discriminative key samples in the validation samples are selected by CEGAT and saved to set U. Where, U is the set of sample indexes selected by the KSS strategy from the validation set. In particular, the 10 key samples selected here refer to the 10 samples with the largest verification loss. (In this article, these samples with high validation losses are considered key samples. Samples with high validation losses have some features that cannot be recognized by the current network. This also means that these samples have greater uncertainty and therefore contain more feature information.) Then, the network is judged whether it has pre converged according to the preset early stop criterion. In particular, the pre-convergence here means that the network reaches the early stop condition. In addition, the pre-set early stop criterion is to verify that the loss value does not decrease for 10 consecutive times. If the network does not pre-converge, CEGAT will continue to execute the next epoch. Otherwise, CEGAT will first select the unlabeled samples corresponding to the index in U from $x_y \in \mathbb{R}^{h\times w\times b}$ and add them to the training set. Then select the same number of unlabeled samples from $x_{te} \in \mathbb{R}^{h\times w\times b}$ to add to the verification set. Finally, continue to execute the next epoch. In particular, after the network model has been judged to have had a pre convergence, it will not be judged whether it has had pre convergence. As shown in the right part of the dotted line in Fig. 4, after updating the set U, CEGAT will determine whether the number of iterations of the KSS has reached or exceeded i. If the number of iterations has reached i, the next epoch will be executed until the end of the training. Otherwise, the selection process of unlabeled samples is continued.

In conclusion, the KSS strategy can effectively mitigate the impact of small samples on network stability. In addition, the KSS strategy can greatly enrich the network features that can be extracted by selecting unlabeled samples containing more feature information for training, thus greatly improving the network accuracy. This provides a new way to solve the problem of small samples in hyperspectral image classification. In particular, the "unlabeled samples" here do not refer to samples without labels in the true sense. The "unlabeled sample" here means that we did not use its real label during training.

3. Experimentation and analysis

In order to verify the effectiveness of the proposed CEGAT network, this paper conducts experiments on five open datasets. All experiments were conducted in the same experimental environment. Specifically, the experimental environment used in this article is equipped with NVIDIA GeForce RTX 3090 GPU, and the compilation software used is PyCharm. In addition, we use python 1.10.3 and python 3.8.13. In order to avoid the accident of the experiment, all experiments were repeated for 20 times, and the average result of the 20 repeated experiments was taken as the final result.

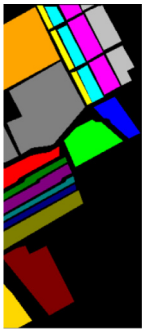
3.1. HSI datasets

This paper verifies the effectiveness of CEGAT network on five open datasets. The five data sets are Indian Pines (IN), Salinas Valley (SV), Pavia University (UP), WHU-Hi-LongKou (LK) and WHU-Hi-HanChuan (HC), as shown in Figs. 5–9. Among them, IN has 200 consecutive spectral bands and includes 16 classes. The number of pixels covered by the ground object is 10 249. SV also contains 16 categories, and these ground objects cover 54 129 pixels. At the same time, SV has 204 consecutive bands. The number of ground object categories and spectral bands included in UP is slightly lower than SV and IN, with only 9 classes and


Ground truth

No.	Legend	Name	Sample	Original train set/ Final train set	Original validation set/ Final validation set	Original test set/ Final test set
1		Alfafa	46	1(1%)/-	-/-	-/-
2		Corn–n	1428	15(1%)/-	-/-	-/-
3		Corn–m	830	9(1%)/-	-/-	-/-
4		Corn	237	3(1%)/-	-/-	-/-
5		Grass–p	483	5(1%)/-	-/-	-/-
6		Grass–t	730	8(1%)/-	-/-	-/-
7		Grass–p–m	28	1(1%)/-	-/-	-/-
8		Hay-w	478	5(1%)/-	-/-	-/-
9		Oats	20	1(1%)/-	-/-	-/-
10		Soybean-n	972	10(1%)/-	-/-	-/-
11		Soybean-m	2455	25(1%)/-	-/-	-/-
12		Soybean-c	593	6(1%)/-	-/-	-/-
13		Wheat	205	3(1%)/-	-/-	-/-
14		Woods	1265	13(1%)/-	-/-	-/-
15		Buildings-G-T	386	4(1%)/-	-/-	-/-
16		Stone-S-T	93	1(1%)/-	-/-	-/-
TOTAL			10249	110(1%)/310(3%)	307(3%)/307(3%)	9832(96%)/9632(94%)

Fig. 5. Description of IN dataset.


Ground truth

No.	Legend	Name	Sample	Original train set/ Final train set	Original validation set/ Final validation set	Original test set/Final test set
1		weeds_1	2009	3(0.1%)/-	-/-	-/-
2		weeds_2	3726	4(0.1%)/-	-/-	-/-
3		Fallow	1976	2(0.1%)/-	-/-	-/-
4		Fallow-r-p	1394	2(0.1%)/-	-/-	-/-
5		Fallow-s	2678	3(0.1%)/-	-/-	-/-
6		Stubble	3959	4(0.1%)/-	-/-	-/-
7		Celery	3579	4(0.1%)/-	-/-	-/-
8		Grapes-u	11271	12(0.1%)/-	-/-	-/-
9		Soil-v-y-d	6203	7(0.1%)/-	-/-	-/-
10		C-s-g-weeds	3278	4(0.1%)/-	-/-	-/-
11		L-r-4wk	1068	2(0.1%)/-	-/-	-/-
12		L-r-5wk	1927	2(0.1%)/-	-/-	-/-
13		L-r-6wk	916	1(0.1%)/-	-/-	-/-
14		L-r-7wk	1070	2(0.1%)/-	-/-	-/-
15		VIN-yard-u	7268	8(0.1%)/-	-/-	-/-
16		VIN-yard-v-t	1807	2(0.1%)/-	-/-	-/-
TOTAL			54129	62(0.1%)/262(0.48%)	260(0.48%)/260(0.48%)	53807(99.42%)/ 53607(99.04%)

Fig. 6. Description of SV dataset.

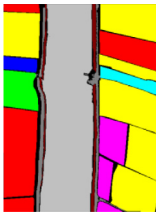

Ground truth

No.	Legend	Name	Sample	Original train set/ Final train set	Original validation set/ Final validation set	Original test set/Final test set
1		Asphalt	6631	7(0.1%)/-	-/-	-/-
2		Meadows	18649	19(0.1%)/-	-/-	-/-
3		Gravel	2099	3(0.1%)/-	-/-	-/-
4		Trees	3064	4(0.1%)/-	-/-	-/-
5		Painted m s	1345	2(0.1%)/-	-/-	-/-
6		B Soil	5029	6(0.1%)/-	-/-	-/-
7		Bitumen	1330	2(0.1%)/-	-/-	-/-
8		S-B Bricks	3682	4(0.1%)/-	-/-	-/-
9		Shadows	947	1(0.1%)/-	-/-	-/-
TOTAL			42776	48(0.1%)/248(0.57%)	243(0.57%)/243(0.57%)	42485(99.33%)/ 42285(98.86%)

Fig. 7. Description of UP dataset.

104 bands. The number of pixels covered by the ground objects is 42 776. LK and HC are data sets collected by Wuhan University. LK has 270 continuous spectral bands, including 9 categories, and its ground objects cover 204 842 pixels. HC has a slightly higher number of bands than LK, and it contains 274 bands. HC also includes 16 classes, and the number of pixels covered by ground objects is up to 257 530.

Figs. 5–9 shows the dataset partition of the proposed method. Different from conventional HSIC methods, the training set, validation set and test set in this paper are dynamically changed during the training process, and this change is not limited to a specific category but is random. This is because the KSS is adopted in this paper, and KSS selects samples based on the importance of samples, not according to category. Therefore, the number of each category in the final training set, validation set and test set of the method proposed in this paper is not necessarily the same in each training process. In particular, the original training set, validation set, and test set in the figure refer to the distribution of the dataset before training begins. The final training set, verification


Ground truth

No.	Legend	Name	Sample	Original train set/ Final train set	Original validation set/ Final validation set	Original test set/Final test set
1		Corn	34511	35(0.1%)/-	-/-	-/-
2		Cotton	8374	9(0.1%)/-	-/-	-/-
3		Sesame	3031	4(0.1%)/-	-/-	-/-
4		Broad-leaf soybean	63212	64(0.1%)/-	-/-	-/-
5		Narrow-leaf soybean	4151	5(0.1%)/-	-/-	-/-
6		Rice	11854	12(0.1%)/-	-/-	-/-
7		Water	67056	68(0.1%)/-	-/-	-/-
8		Roads and houses	7124	8(0.1%)/-	-/-	-/-
9		Mixed weed	5229	6(0.1%)/-	-/-	-/-
TOTAL			204542	211(0.1%)/411(0.2%)	409(0.2%)/409(0.2%)	203922(99.70%)/ 203722(99.60%)

Fig. 8. Description of LK dataset.


Ground truth

No.	Legend	Name	Sample	Original train set/ Final train set	Original validation set/ Final validation set	Original test set/Final test set
1		Strawberry	44735	45(0.1%)/-	-/-	-/-
2		Cowpea	22753	23(0.1%)/-	-/-	-/-
3		Soybean	10287	11(0.1%)/-	-/-	-/-
4		Sorghum	5353	6(0.1%)/-	-/-	-/-
5		Water spinach	1200	2(0.1%)/-	-/-	-/-
6		Watermelon	4533	5(0.1%)/-	-/-	-/-
7		Greens	5903	6(0.1%)/-	-/-	-/-
8		Trees	17978	18(0.1%)/-	-/-	-/-
9		Grass	9469	10(0.1%)/-	-/-	-/-
10		Red roof	10516	11(0.1%)/-	-/-	-/-
11		Gray roof	16911	17(0.1%)/-	-/-	-/-
12		Plastic	3679	4(0.1%)/-	-/-	-/-
13		Bare soil	9116	10(0.1%)/-	-/-	-/-
14		Road	18560	19(0.1%)/-	-/-	-/-
15		Bright object	1136	2(0.1%)/-	-/-	-/-
16		Water	75401	76(0.1%)/-	-/-	-/-
TOTAL			257530	265(0.1%)/462(0.18%)	463(0.18%)/463(0.18%)	256802(99.72%)/ 256602(99.64%)

Fig. 9. Description of HC dataset.

set and test set refer to the distribution of the data set after KSS selects samples. '–' indicates that the sample size is unknown.

3.2. Parameter settings

For fair comparison, the hyperparameter settings of all comparison methods are the same as those in the reference. For the proposed CEGAT, we use an learning rate of 0.001 and sets the epoch to 600. In addition, for different data sets, different sizes of super pixel segmentation scales are set. Tables 1–5 show the impact of different superpixel segmentation scales on classification results on different datasets. Specifically, as can be seen from Table 1, there is no significant difference between the classification results of the segmentation scale with 50 and the classification results of the segmentation scale with 100. However, for the segmentation scale with 100, the floating point operations (FLOPs) of the network have a greater advantage compared with that of the segmentation scale with 50. In addition, on the IN dataset, when the segmentation scale is greater than 100, although the FLOPs of the network continue to decrease, the classification accuracy also continues to decline. In this paper, the scale of superpixel segmentation on IN data set is set to 100. For other datasets, the scale of superpixel segmentation is different from that of the IN dataset. This is because the resolution of other datasets is much higher than IN, which increases the computational complexity of the network. Therefore, the segmentation scale is set relatively larger on other datasets. Finally, the scale of superpixel segmentation on SV, UP, LK, and HC is set to 500. The reason is that when the segmentation scale is 500, it has better classification accuracy and lower network computational complexity compared to other segmentation scales.

Finally, in order to further improve the classification performance, the number of iterations of the KSS strategy is explored. As shown in Fig. 10, six experiments were conducted on each dataset. Specifically, the number of iterations of the KSS strategy is set to {5, 10, 15, 20, 25, 30}. The overall accuracy (OA), average accuracy (AA) and Kappa coefficient (Kappa) of the six groups of experiments are compared respectively. From the experimental results, it can be found that there is a similar performance on all data sets. That is, as the number of iterations of the KSS increases, the three evaluation indicators show an upward trend and gradually converge. Among them, the four data sets UP, SV, LK and HC begin to converge when the number of iterations is about 20, while IN begins to converge when the number of iterations is about 25. This is because the total number of samples in the IN dataset is small, but the number of categories is relatively large, which makes the amount of feature information available for classification less. Therefore, more iteration times are required for IN data set to improve classification performance. In order to facilitate subsequent experiments, the number of iterations of the KSS strategy is set to 20.

3.3. Performance analysis of proposed modules

3.3.1. Ablation experiments

In this part, some ablation experiments were performed on the modules proposed in this paper on five data sets to verify their effectiveness. Take OA as the evaluation index. In addition, all experiments used the same number of training samples on the same dataset. Specifically, on the IN dataset, 3% of the samples are used for training (when using KSS, the training samples consist of 1% of the original training samples and 2% of the samples

Table 1
Effects of different superpixel segmentation scales on classification on the IN dataset.

IN					
Scale	50	100	200	300	400
OA	98.41(0.0034)	98.39(0.0041)	98.07(0.0025)	97.97(0.0059)	97.04(0.0047)
AA	98.33(0.0061)	98.93(0.0124)	98.22(0.0036)	98.03(0.0068)	97.94(0.0043)
KAPPA	98.16(0.0038)	98.16(0.0047)	97.81(0.0029)	97.69(0.0067)	96.64(0.0053)
FLOPs	1280.31M	1154.15M	1128.81M	1125.41M	1120.87M

Table 2
Effects of different superpixel segmentation scales on classification on the SV dataset.

SV					
Scale	300	400	500	600	700
OA	98.37(0.0075)	97.86(0.0051)	99.18(0.0029)	98.76(0.0041)	98.28(0.0038)
AA	99.09(0.0044)	98.71(0.0034)	99.47(0.0016)	99.25(0.0014)	99.06(0.0022)
KAPPA	98.19(0.0084)	97.62(0.0056)	99.08(0.0033)	98.62(0.0045)	98.08(0.0043)
FLOPs	6031.76M	5975.23M	5951.37M	5936.84M	5928.06M

Table 3
Effects of different superpixel segmentation scales on classification on the UP dataset.

UP					
Scale	300	400	500	600	700
OA	98.95(0.0078)	99.07(0.0039)	99.20(0.0035)	99.11(0.0037)	98.96(0.0042)
AA	98.19(0.0169)	98.60(0.0068)	98.83(0.0035)	98.60(0.0073)	98.21(0.0098)
KAPPA	98.62(0.0104)	98.76(0.0053)	98.90(0.0024)	98.82(0.0049)	98.62(0.0055)
FLOPs	11 147.26M	10 947.47M	10 881.72M	10 838.82M	10 812.11M

Table 4
Effects of different superpixel segmentation scales on classification on the LK dataset.

LK					
Scale	300	400	500	600	700
OA	99.65(0.0006)	99.63(0.0005)	99.55(0.0007)	99.67(0.0005)	99.63(0.0006)
AA	98.95(0.0022)	98.88(0.0022)	98.70(0.0025)	99.04(0.0016)	98.90(0.0027)
KAPPA	99.54(0.0008)	99.51(0.0007)	99.41(0.0009)	99.57(0.0007)	99.51(0.0008)
FLOPs	11 854.06M	11 624.06M	11 554.34M	11 515.10M	11 472.11M

Table 5
Effects of different superpixel segmentation scales on classification on the HC dataset.

HC					
Scale	300	400	500	600	700
OA	97.10(0.0045)	96.89(0.0053)	96.96(0.0036)	96.80(0.0040)	96.85(0.0036)
AA	93.08(0.0125)	92.92(0.0143)	93.10(0.0101)	92.89(0.0133)	93.06(0.0110)
KAPPA	96.61(0.0052)	96.56(0.0046)	96.44(0.0043)	96.25(0.0047)	96.43(0.0041)
FLOPs	21 476.94M	20 839.38M	20 652.87M	20 535.09M	20 441.13M

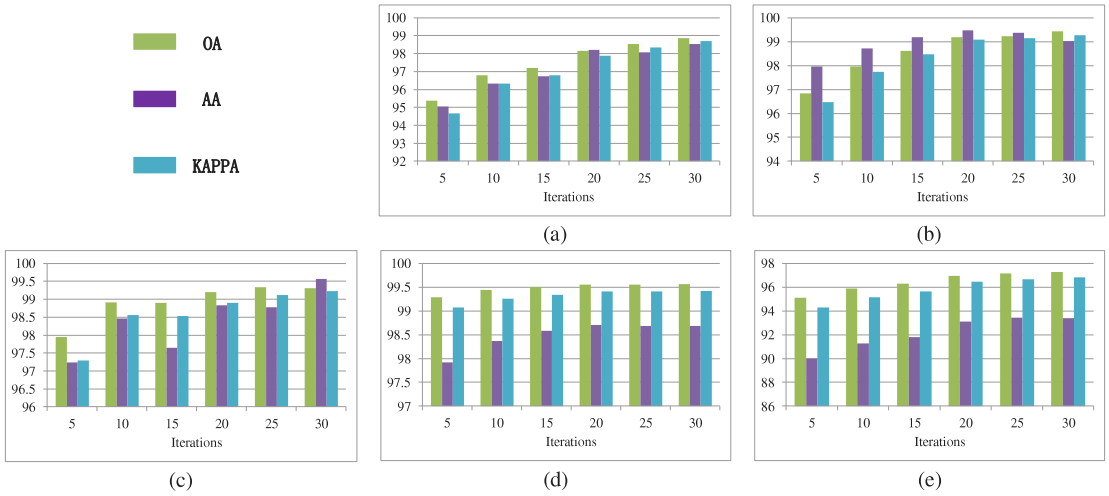


Fig. 10. Experimental results of KSS strategy in different data sets (a) Experimental results on IN (b) Experimental results on SV; (c) Experimental results on UP; (d) Experimental results on LK; (e) Experimental results on HC.

Table 6
Comparison of different modules of CEGAT. (– means the module is not used in those experiments, √ means the module is used in those experiments).

The number of the experiment		1	2	3	4	5
Modules	KSS	–	–	√	√	√
	EGAT	–	√	–	√	√
	SSCA	√	√	√	–	√
Data set	IN	95.79(0.0107)	97.09(0.0071)	97.87(0.0038)	82.07(0.0093)	**98.14(0.0035)**
	SV	96.23(0.0083)	98.83(0.0038)	96.99(0.0047)	85.21(0.0067)	**99.18(0.0029)**
	UP	96.77(0.0068)	98.56(0.0058)	97.85(0.0049)	86.96(0.0079)	**99.20(0.0035)**
	LK	99.02(0.0013)	99.32(0.0011)	99.23(0.0010)	91.20(0.0027)	**99.55(0.0075)**
	HC	91.57(0.0061)	96.31(0.0039)	93.90(0.0050)	87.40(0.0085)	**96.96(0.0036)**

1: CEGAT (only with SSCA).
2: CEGAT (with EGAT and SSCA).
3: CEGAT (with KSS and SSCA).
4: CEGAT (with KSS and EGAT).
5: Complete CEGAT.

selected by KSS through training iterations). On SV, 0.48% of the samples are used for training (when using KSS, the training samples consist of 0.1% of the original training samples and 0.38% of the samples selected by KSS through training iterations). On UP, 0.57% of the samples are used for training (when using KSS, the training samples consist of 0.1% of the original training samples and 0.47% of the samples selected by KSS through training iterations). On LK, 0.2% of the samples are used for training (when using KSS, the training samples consist of 0.1% of the original training samples and 0.1% of the samples selected by KSS through training iterations). On HC, 0.18% of the samples are used for training (when using KSS, the training samples consist of 0.1% of the original training samples and 0.08% of the samples selected by KSS through training iterations). The experimental results are shown in Table 6.

First, the effectiveness of the KSS strategy is verified. Compare Experiment 2 with Experiment 5, it can be seen that the classification accuracy of the complete CEGAT network is significantly improved compared with the network without the KSS strategy on the five datasets. Among them, on the LK dataset, the performance improvement of the network classification using the KSS strategy is slightly weaker than that of other datasets. This is because LK dataset itself has more label samples, which provides a strong guarantee for the network to extract sufficient feature information. In addition, LK samples have fewer classes, which reduces mutual interference between categories in the classification process. These factors make the classification performance of CEGAT network on the LK dataset not excessively depend on the KSS strategy. Nevertheless, compared with CEGAT without KSS strategy, the classification accuracy of CEGAT with KSS strategy on the LK is still improved by 1.06%. These results fully prove the effectiveness of the KSS strategy.

Then, the validity of EGAT module is verified. Specifically, the results of Experiment 3: CEGAT (with KSS and SSCA) and Experiment 5: Complete CEGAT were compared. Obviously, on all data sets, the classification accuracy of CEGAT networks with EGAT modules is improved to some extent compared with networks without EGAT modules. This shows that EGAT module can effectively improve the classification performance of the network.

Finally, the effectiveness of SSCA module is discussed. The results of Experiment 4: CEGAT (with KSS and EGAT) and Experiment 5: Complete CEGAT were compared. It can be seen that compared with the CEGAT network without SSCA, the classification performance of the CEGAT network with SSCA on all datasets has been greatly improved. This is because when the CEGAT network does not use SSCA, there is only a single branch of GAT in the network, which weakens the discrimination ability of the network. And on the GAT branch, SLIC is used to construct the graph structure. Although this can reduce the number of network parameters, it will lose part of the feature information to a certain extent. Therefore, when CEGAT network does not have pixel level feature fusion extracted by SSCA, the network will show poor classification performance. This not only proves that the SSCA module can effectively improve the classification performance of the CEGAT, but also just proves that the super pixel feature and pixel feature are complementary.

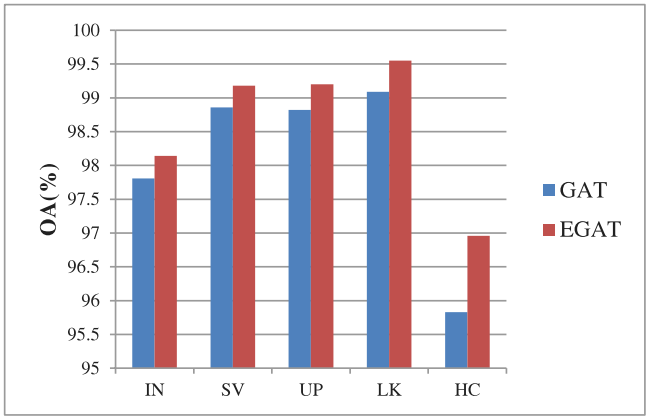


Fig. 11. Comparison between conventional GAT and EGAT.

In addition, the robustness of the proposed module has also been verified in this section. By comparing 1: CEGAT (only with SSCA) with 2: CEGAT (with EGAT and SSCA), and comparing 3: CEGAT (with KSS and SSCA) with 5: Complete CEGAT. From the results, it can be seen that EGAT can not only improve the performance of the network when KSS and SSCA are used, but also stably improve the performance of the network when only SSCA is used. Obviously, this indicates that the EGAT proposed in this article has good robustness. Similarly, by comparing 1: CEGAT (only with SSCA) with 3: CEGAT (with KSS and SSCA), and comparing 2: CEGAT (with EGAT and SSCA) with 5: Complete CEGAT, it can be concluded that IAL also has good robustness.

3.3.2. Comparison experiments

In this part, the effectiveness of EGAT module is further verified. The classification performance of conventional GAT is compared with that of the EGAT proposed in this paper, and the experimental results are shown in Fig. 11. In order to ensure fair comparison of the experiment, other experimental conditions were kept unchanged and only GAT and EGAT were replaced in the proposed network. The experimental results show that the proposed EGAT module has obvious advantages on five datasets. The reason is that EGAT can strengthen the network's attention

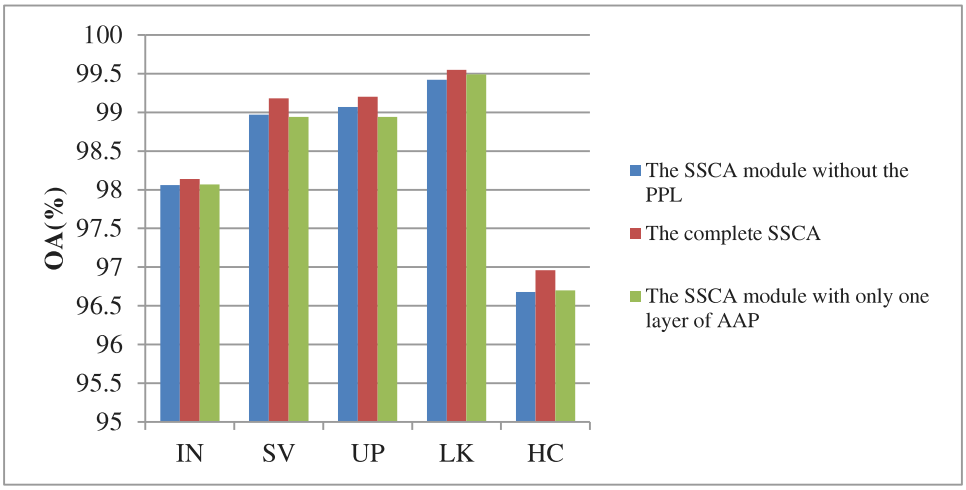


Fig. 12. Comparison between with and without AAP in SSCA.

to features on the basis of conventional GAT, enable the network to extract more discriminative features, and weaken redundant features that interfere with classification.

Furthermore, in order to explore the effect of pyramid pooling layer (PPL) of the proposed SSCA on the classification results, the complete SSCA module (SSCA module with pyramid pooling layer), the SSCA module without the PPL, and the SSCA module with only one layer of AAP are compared in Fig. 12. As can be seen from Fig. 12, on five different data sets, the classification results of the three methods all show that the classification accuracy of the complete SSCA module is higher than that of the other two methods. Comparing the complete SSCA module with the SSCA module without the PPL, it can be seen that introducing the PPL in SSCA is more beneficial to the classification results. Comparing the complete SSCA module with the SSCA module with only one layer of AAP, it can be seen that the use of PPL in SSCA is more beneficial to the classification results than one layer of AAP. This is because the PPL reduces the loss of feature information compared to the one layer of AAP.

3.4. Comparison of different methods

In this part, the CEGAT is compared with seven recent HSIC methods, including DBDA (Li et al., 2020), DBMA (Ma et al., 2019), FDSSC (Wang et al., 2018), SSRN (Zhong et al., 2018), CEGCN (Liu et al., 2021), FDGCN (Liu, Dong, Zhang, & Luo, 2022), WFCG (Dong et al., 2022). Among them, DBDA, DBMA, FDSSC and SSRN are four CNN based hyperspectral image classification methods, while the other three methods are HSIC methods based on the combination of GCN and CNN. Tables 7–11 shows the classification results of all methods on different data sets. The evaluation indicators adopted include OA, AA, KAPPA, parameter quantity and network running time. In Tables 7–11, the best classification results are bold. In addition, in order to further analyze the classification performance of the proposed CEGAT network, this paper visualizes the classification results of all methods, as shown in Figs. 13–17.

In general, the CEGAT has achieved the highest OA, AA and KAPPA in five data sets. This proves that the proposed CEGAT is more competitive than other methods in classification performance. In addition, CEGAT also achieved the best results for the classification accuracy of each category of different data sets. This is due to the fact that CEGAT network removes redundant spectral information of HSI at the very beginning, avoiding the interference of irrelevant information on classification. Not only that, CEGAT also uses EGAT module and SSCA to increase feature differentiation and enhance the discrimination of important features. At the same time, the use of the KSS strategy increases the feature information that can be extracted by the network. The CEGAT not only achieves the highest classification accuracy on all data sets, but also has the least number of network parameters, which saves the hardware memory and provides the possibility for its application to relevant practical applications.

Specifically, CEGAT achieves the best classification performance on IN dataset. Among other methods, compared with the method only based on CNN, the HSIC method based on GCN and CNN has certain advantages. In the HSIC method based on the combination of GCN and CNN, CEGCN and WFCG have greater advantages than FDGCN. This phenomenon also occurs in several other data sets. This is because both CEGCN and WFCG use the method of fusion of super pixel level features and pixel level features to classify. The CEGAT proposed in this paper also uses this method. In contrast, CEGAT removes the redundant information of the original data before extracting the super pixel level features and pixel level features, which makes the classification stage smoother. In addition, CEGAT can achieve the best classification performance on all datasets by using fewer label samples. This is because the proposed KSS strategy can select unlabeled samples with more information in the validation set for training. This not only saves the cost of labeling, but also effectively alleviates the problem of small samples and improves the classification accuracy of the network.

It can be seen from Figs. 13–17 that on five data sets, compared with other methods, the classification map obtained by the CEGAT is closer to the ground truth, with clearer boundaries and less noise. This is because the KSS strategy adds unlabeled samples that are difficult to distinguish in the verification set to the training set through training iteration, which increases the amount of feature information of training samples. At the same time, CEGAT fuses the graph features at the super pixel level with the spatial spectral features at the pixel level through two attention modules, EGAT and SSCA. These reasons reduce the interference between different classes in the dataset and make the classification smoother. Therefore, CEGAT can obtain the classification result map closer to the ground truth.

In order to compare the classification performance of different methods more intuitively, a T-distributed stochastic neighbor embedding (T-SNE) (Van der Maaten & Hinton, 2008) method is utilized on IN and UP datasets to visualize the feature maps of the proposed method and the four more competitive methods. The experimental results are shown in Figs. 18–19. In particular, these four methods include two CNN based methods, i.e., DBDA

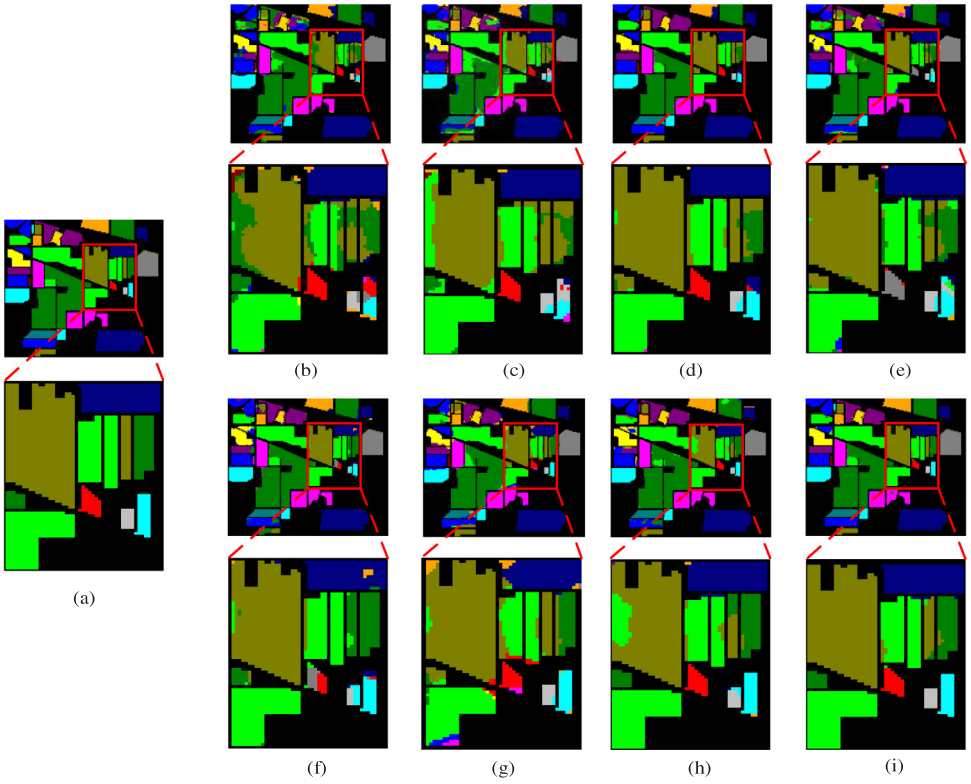


Fig. 13. Classification maps on IN. (a) Ground truth; (b) DBDA; (c) DBMA; (d) FDSSC; (e) SSRN; (f) CEGCN; (g) FDGC; (h) WFCG; (i) CEGAT.

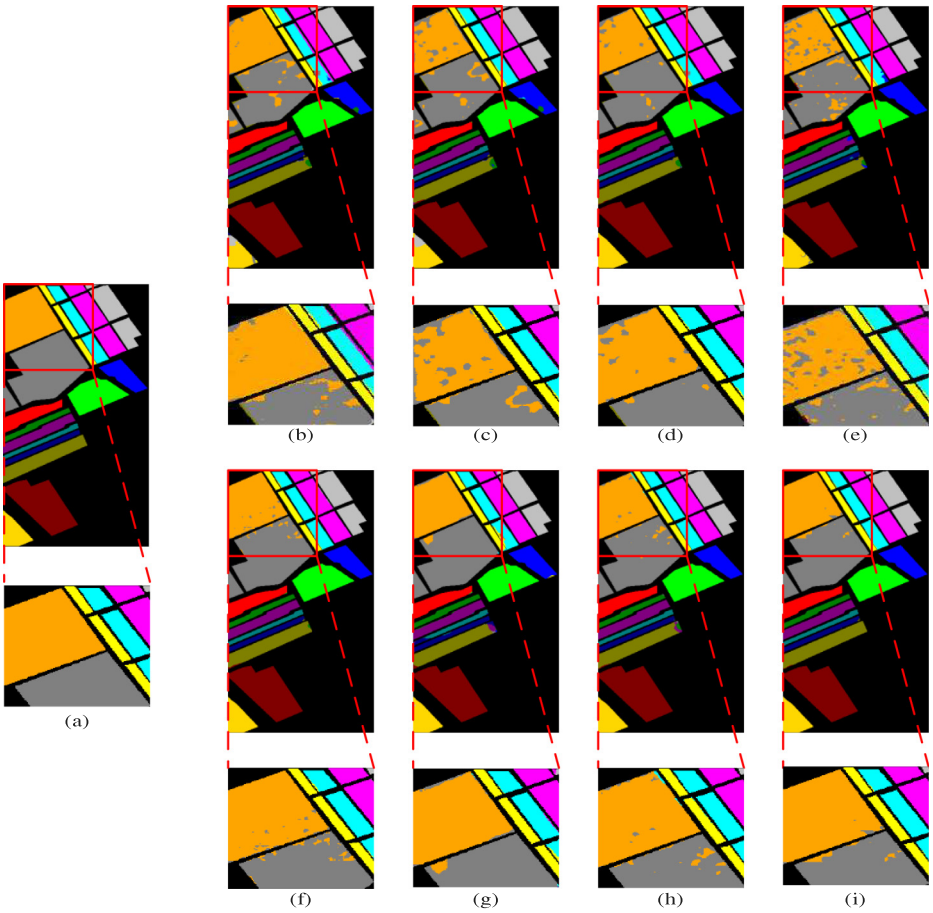


Fig. 14. Classification maps on SV. (a) Ground truth; (b) DBDA; (c) DBMA; (d) FDSSC; (e) SSRN; (f) CEGCN; (g) FDGC; (h) WFCG; (i) CEGAT.

Table 7
Classification results on IN.

Methods	DBDA	DBMA	FDSSC	SSRN	CEGCN	FDGCN	WFCG	CEGAT
Label samples	310(3%)	310(3%)	310(3%)	310(3%)	310(3%)	310(3%)	310(3%)	110(1%)
Unlabel samples	–	–	–	–	–	–	–	200(2%)
1	92.53(0.0659)	82.75(0.1360)	93.01(0.0868)	82.28(0.1509)	48.96(0.2788)	78.10(0.1723)	96.23(0.0186)	**97.92(0.0230)**
2	91.82(0.0657)	90.33(0.0333)	89.41(0.1061)	87.96(0.0301)	93.53(0.0295)	91.14(0.0289)	94.00(0.0240)	**96.56(0.0152)**
3	94.95(0.0327)	92.94(0.0620)	87.54(0.1254)	91.19(0.0264)	90.00(0.0570)	94.86(0.0374)	93.24(0.0314)	**97.06(0.0253)**
4	91.75(0.0413)	87.55(0.0670)	91.79(0.0543)	81.81(0.1014)	77.26(0.1377)	91.18(0.0668)	98.06(0.0177)	**99.64(0.0087)**
5	98.13(0.0146)	93.47(0.0835)	**98.93(0.0162)**	96.11(0.0476)	90.40(0.0600)	94.59(0.0330)	94.21(0.0438)	96.15(0.0394)
6	96.74(0.0235)	96.33(0.0256)	97.76(0.0265)	96.84(0.2894)	**99.50(0.0037)**	96.02(0.0237)	98.94(0.0091)	98.84(0.0076)
7	66.55(0.1726)	53.13(0.1744)	81.77(0.1946)	82.18(0.2235)	50.80(0.3178)	66.61(0.2558)	96.53(0.0676)	**97.46(0.0472)**
8	99.90(0.0006)	99.82(0.0035)	99.95(0.0013)	98.31(0.0145)	99.93(0.0014)	99.61(0.0080)	99.78(0.0040)	**99.97(0.0011)**
9	72.18(0.1696)	60.75(0.2139)	64.07(0.1752)	76.89(0.2197)	26.45(0.1526)	73.18(0.2330)	92.77(0.1630)	**98.38(0.0346)**
10	89.73(0.0512)	86.47(0.1134)	90.69(0.0400)	83.70(0.0605)	93.17(0.0542)	87.62(0.0422)	94.40(0.0394)	**97.45(0.0201)**
11	96.17(0.0196)	92.14(0.0476)	94.39(0.0597)	87.43(0.0425)	97.58(0.0122)	93.27(0.2909)	97.69(0.1440)	**98.39(0.1158)**
12	90.01(0.0832)	82.79(0.1305)	95.69(0.0385)	85.55(0.0524)	94.37(0.0341)	87.43(0.0422)	93.21(0.0664)	**98.85(0.0069)**
13	94.98(0.0595)	94.99(0.0494)	97.74(0.0415)	95.67(0.0784)	**99.79(0.0025)**	93.37(0.0652)	98.00(0.0434)	99.24(0.0184)
14	97.68(0.0114)	96.87(0.0189)	96.38(0.0289)	96.91(0.0128)	98.59(0.0284)	98.14(0.0128)	99.49(0.0035)	**99.82(0.0011)**
15	91.25(0.0517)	84.40(0.0860)	92.97(0.0560)	87.14(0.0555)	86.99(0.0771)	94.02(0.0474)	98.70(0.0132)	**99.86(0.0205)**
16	89.73(0.1087)	91.30(0.1156)	94.75(0.0429)	93.80(0.0396)	90.84(0.0565)	72.51(0.0883)	96.74(0.0459)	**97.13(0.0645)**
OA	94.06(0.0184)	90.68(0.0304)	92.72(0.0369)	89.76(0.0150)	94.47(0.0164)	92.54(0.0134)	96.52(0.0092)	**98.14(0.0035)**
AA	90.68(0.0218)	86.63(0.0163)	91.68(0.0168)	88.99(0.0235)	83.63(0.0487)	88.16(0.0294)	96.41(0.0181)	**98.20(0.0051)**
KAPPA	93.24(0.0209)	89.38(0.0345)	91.71(0.0420)	88.30(0.0173)	93.68(0.0190)	91.48(0.0154)	96.03(0.0105)	**97.88(0.0040)**
Parameter	610.617k	606.101k	1277.490k	364.168k	166.370k	2449.700k	102.422k	**99.213k**
Train time	229.32 s	224.97 s	300.17 s	186.99 s	**11.37 s**	41.63 s	18.76 s	43.79 s
Test time	4.91 s	4.90 s	6.58 s	2.97 s	3.19 s	0.50 s	**0.38 s**	0.43 s

Table 8
Classification results on SV.

Methods	DBDA	DBMA	FDSSC	SSRN	CEGCN	FDGCN	WFCG	CEGAT
Label samples	262 (0.48%)	262 (0.48%)	262 (0.48%)	262 (0.48%)	262 (0.48%)	262 (0.48%)	262 (0.48%)	62 (0.1%)
Unlabel samples	–	–	–	–	–	–	–	200 (0.38%)
1	99.98(0.0001)	99.99(0.0012)	100.0(0.0000)	99.28(0.0174)	99.97(0.0007)	99.90(0.0020)	99.59(0.0085)	**100.0(0.0000)**
2	99.97(0.0051)	99.97(0.0005)	100.0(0.0000)	99.62(0.0105)	100.0(0.0000)	99.60(0.0071)	99.96(0.0010)	**100.0(0.0000)**
3	97.94(0.0222)	97.94(0.0158)	95.46(0.0817)	94.54(0.0491)	99.75(0.0073)	99.20(0.0155)	99.58(0.0067)	**99.96(0.0008)**
4	93.52(0.0529)	91.92(0.0503)	96.56(0.0344)	93.97(0.1112)	99.19(0.0121)	90.68(0.0523)	98.82(0.0108)	**99.63(0.0014)**
5	97.94(0.0307)	98.56(0.0092)	99.39(0.0038)	98.57(0.0144)	98.10(0.0206)	98.18(0.0186)	98.63(0.0135)	**99.46(0.0017)**
6	**99.94(0.0006)**	98.47(0.0164)	99.89(0.0029)	99.85(0.0034)	99.85(0.0026)	99.01(0.0105)	99.87(0.0035)	99.12(0.0096)
7	98.49(0.0245)	98.53(0.0216)	99.92(0.0016)	99.90(0.0024)	99.99(0.0001)	99.20(0.0106)	99.98(0.0004)	**100.0(0.0000)**
8	93.52(0.0335)	93.66(0.0114)	91.85(0.0296)	82.73(0.0627)	97.28(0.1276)	98.41(0.0146)	97.41(0.0214)	**98.72(0.0063)**
9	99.30(0.0092)	99.46(0.0051)	99.67(0.0030)	99.63(0.0024)	100.0(0.0000)	99.67(0.0024)	99.96(0.0006)	**100.0(0.0000)**
10	**98.52(0.0112)**	95.97(0.0308)	97.29(0.0233)	95.69(0.0277)	96.16(0.0169)	97.83(0.0168)	97.76(0.0174)	98.45(0.0060)
11	95.56(0.0200)	95.90(0.0320)	95.49(0.0238)	95.29(0.0219)	99.25(0.0112)	91.55(0.0636)	**99.37(0.0106)**	99.34(0.0039)
12	99.43(0.0142)	98.99(0.0150)	98.69(0.0190)	98.27(0.0134)	100.0(0.0000)	94.58(0.0416)	99.99(0.0001)	**100.0(0.0000)**
13	99.59(0.0054)	99.39(0.0068)	99.85(0.0035)	98.04(0.0184)	**99.87(0.0026)**	90.46(0.0963)	99.26(0.0099)	99.49(0.0024)
14	96.36(0.0175)	95.99(0.0220)	96.73(0.0152)	96.65(0.0317)	98.76(0.0171)	89.22(0.0765)	97.13(0.0251)	**99.83(0.0017)**
15	86.52(0.1578)	92.03(0.0364)	90.25(0.0472)	81.64(0.0792)	**99.13(0.0066)**	90.02(0.0383)	97.61(0.0174)	97.49(0.0123)
16	99.88(0.0023)	99.15(0.0132)	99.66(0.0075)	99.63(0.0082)	99.43(0.0149)	99.26(0.0135)	99.55(0.0131)	**100.0(0.0000)**
OA	94.88(0.0544)	96.43(0.0069)	96.16(0.0099)	92.42(0.0196)	98.89(0.0024)	96.63(0.0082)	98.76(0.0060)	**99.18(0.0029)**
AA	97.28(0.0123)	97.25(0.0046)	97.54(0.0070)	95.83(0.0123)	99.17(0.0026)	96.05(0.0107)	99.03(0.0028)	**99.47(0.0016)**
KAPPA	94.33(0.0595)	96.03(0.0077)	95.72(0.0110)	91.55(0.0219)	98.76(0.0026)	96.26(0.0091)	98.62(0.0067)	**99.08(0.0033)**
Parameter	622.233k	617.717k	1251.490k	370.312k	166.890k	2449.700k	102.942k	**99.213k**
Train time	194.50 s	190.37 s	248.05 s	157.59 s	**23.45 s**	28.35 s	55.52 s	85.49 s
Test time	28.94 s	28.68 s	37.58 s	17.73 s	4.80 s	2.70 s	1.89 s	**1.80 s**

and FDSSC, and two methods based on the combination of GCN and CNN, i.e., CEGCN and WFCG. It can be seen that in Fig. 18, for the feature map of the proposed CEGAT method, different classes can be separated from each other clearly and the same classes can be clustered better. However, there will be many intersections between features of different classes in other methods. And the same conclusion can be drawn in Fig. 19. This fully proves that the CEGAT can provide better classification performance of hyperspectral images. In addition, it can be seen from Figs. 18–19 that the effect of the classification method based on the combination of GCN and CNN is obviously better than that of the classification method based on CNN.

Finally, in order to verify the effectiveness of the fusion of super pixel level features and pixel level features, some experiments are conducted on two more challenging datasets, as shown in Figs. 20–21. From the local feature map of IN, we can see that the local feature map obtained by CEGAT network without EGAT branches is rough and far from the real local feature map. On the contrary, the local feature map obtained by the complete CEGAT network is smoother and closer to the real local feature map. The same conclusion can be drawn from the local feature maps of HC dataset. This is because SSCA is unable to model the correlation of HSI internal structure, and EGAT directly acts on the graph to make up for the deficiency of SSCA. It also proves that the super pixel level features extracted by EGAT and the pixel level features extracted by SSCA are complementary.

4. Conclusions

Due to the excellent feature extraction ability of CNN, it has been widely used in the classification of hyperspectral images. But the regular grid processing method of CNN makes it difficult to establish the internal association of hyperspectral image

Table 9
Classification results on UP.

Methods	DBDA	DBMA	FDSSC	SSRN	CEGCN	FDGCN	WFCG	CEGAT
Label samples	248(0.57%)	248(0.57%)	248(0.57%)	248(0.57%)	248(0.57%)	248(0.57%)	248(0.57%)	48(0.1%)
Unlabel samples	–	–	–	–	–	–	–	200(0.47%)
1	96.63(0.0199)	96.49(0.0093)	98.69(0.0162)	97.76(0.0131)	**99.07(0.0067)**	87.76(0.0574)	98.52(0.0121)	98.84(0.0030)
2	99.37(0.0050)	98.93(0.0049)	99.23(0.0034)	98.34(0.0039)	99.96(0.0003)	99.02(0.0049)	99.95(0.0006)	**100.0(0.0000)**
3	87.39(0.1637)	88.97(0.0811)	88.58(0.0822)	78.55(0.1052)	97.48(0.0323)	94.75(0.0422)	96.88(0.0355)	**98.92(0.0293)**
4	96.44(0.0195)	96.71(0.0177)	99.47(0.0044)	**99.67(0.0043)**	95.07(0.0230)	79.42(0.0632)	95.35(0.0218)	96.44(0.0115)
5	97.78(0.0355)	98.35(0.0189)	99.84(0.0014)	99.96(0.0003)	99.95(0.0009)	95.01(0.0401)	99.97(0.0004)	**100.0(0.000)**
6	96.88(0.0565)	98.67(0.0124)	98.71(0.0106)	96.82(0.0236)	99.97(0.0005)	96.19(0.0236)	99.96(0.0007)	**100.0(0.000)**
7	94.89(0.0409)	97.71(0.0309)	99.07(0.0147)	87.17(0.1647)	98.39(0.0196)	98.36(0.0367)	**99.83(0.0022)**	98.48(0.0200)
8	86.73(0.0683)	88.34(0.0561)	88.43(0.0427)	86.26(0.0650)	97.11(0.0184)	79.80(0.0366)	**98.58(0.0143)**	96.81(0.0236)
9	98.07(0.0147)	95.43(0.0340)	98.82(0.0096)	98.90(0.0105)	96.80(0.0574)	75.18(0.1402)	95.64(0.0618)	**99.57(0.0045)**
OA	96.02(0.0181)	96.71(0.0058)	97.54(0.0058)	95.34(0.177)	98.99(0.0029)	92.45(0.0147)	99.03(0.0033)	**99.20(0.0035)**
AA	94.91(0.0198)	95.51(0.0096)	96.76(0.0101)	93.71(0.0251)	98.20(0.0089)	89.50(0.0167)	98.72(0.0076)	**98.83(0.0035)**
KAPPA	94.74(0.0237)	95.63(0.0078)	96.74(0.0076)	93.82(0.0231)	98.66(0.0039)	89.98(0.0195)	98.72(0.0043)	**98.90(0.0024)**
Parameter	325.202k	320.686k	651.063k	216.537k	152.857k	1987.690k	99.357k	**97.848k**
Train time	134.45 s	129.41 s	161.71 s	113.55 s	36.74 s	**27.68 s**	97.40 s	148.26 s
Test time	13.69 s	13.74 s	17.32 s	10.30 s	6.44 s	**2.09 s**	2.99 s	3.6 s

Table 10
Classification results on LK.

Methods	DBDA	DBMA	FDSSC	SSRN	CEGCN	FDGCN	WFCG	CEGAT
Label samples	405(0.2%)	405(0.2%)	405(0.2%)	405(0.2%)	405(0.2%)	405(0.2%)	405(0.2%)	205(0.1%)
Unlabel samples	–	–	–	–	–	–	–	200(0.1%)
1	99.54(0.0037)	99.37(0.0054)	99.64(0.0029)	99.43(0.0025)	99.77(0.0018)	99.34(0.0052)	99.89(0.0007)	**99.93(0.0003)**
2	96.08(0.0193)	93.24(0.0409)	95.27(0.0328)	93.80(0.0399)	98.30(0.0166)	96.44(0.0433)	98.45(0.0175)	**99.84(0.0035)**
3	94.45(0.0452)	97.69(0.0373)	99.16(0.0158)	97.48(0.0276)	91.01(0.0580)	96.69(0.0250)	94.01(0.0448)	**99.30(0.0123)**
4	97.82(0.0098)	98.11(0.0249)	99.21(0.0003)	98.11(0.0074)	99.50(0.0028)	97.38(0.0156)	99.35(0.0020)	**99.72(0.0011)**
5	91.84(0.0675)	93.94(0.0485)	95.45(0.0032)	90.44(0.1068)	94.41(0.0277)	95.52(0.0349)	93.25(0.0551)	**98.81(0.0102)**
6	99.25(0.0090)	99.60(0.0028)	99.82(0.0351)	99.70(0.0028)	99.02(0.0173)	98.70(0.0105)	98.81(0.0175)	**100.0(0.0000)**
7	99.56(0.0025)	99.70(0.0027)	99.80(0.0017)	99.77(0.0023)	99.92(0.0009)	99.13(0.0051)	**99.96(0.0003)**	99.94(0.0004)
8	87.63(0.0537)	89.35(0.0422)	93.24(0.0350)	92.10(0.0408)	94.06(0.0263)	86.64(0.0362)	**95.50(0.0191)**	94.84(0.0180)
9	91.87(0.1689)	84.99(0.0678)	93.97(0.0234)	91.84(0.991)	87.85(0.0497)	83.93(0.0377)	90.75(0.0414)	**95.97(0.0192)**
OA	**97.48(0.0099)**	97.96(0.0096)	98.91(0.0025)	98.19(0.0086)	98.91(0.0036)	97.54(0.0056)	99.02(0.0023)	**99.55(0.0007)**
AA	94.23(0.0190)	95.12(0.0155)	96.28(0.0099)	95.85(0.0273)	95.99(0.0148)	94.86(0.0083)	96.66(0.0100)	**98.70(0.0025)**
KAPPA	96.69(0.0129)	97.32(0.0128)	98.57(0.0033)	97.62(0.0113)	98.57(0.0048)	96.76(0.0075)	98.71(0.0030)	**99.41(0.0009)**
Parameter	813.050k	808.534k	1647.063k	471.513k	174.567k	1987.609k	111.067k	**97.848k**
Train time	371.70 s	366.94 s	405.81 s	289.62 s	47.09 s	**45.90 s**	115.22 s	151.99 s
Test time	135.48 s	135.14 s	126.06 s	81.80 s	8.68 s	10.25 s	**3.40 s**	4.66 s

Table 11
Classification results on HC.

Methods	DBDA	DBMA	FDSSC	SSRN	CEGCN	FDGCN	WFCG	CEGAT
Label samples	457 (0.18%)	457 (0.18%)	457 (0.18%)	457 (0.18%)	457 (0.18%)	457 (0.18%)	457 (0.18%)	257 (0.1%)
Unlabel samples	–	–	–	–	–	–	–	200 (0.08%)
1	88.35(0.0446)	86.07(0.0841)	91.97(0.0603)	83.61(0.0784)	**99.39(0.0023)**	95.26(0.0156)	98.07(0.0076)	98.65(0.0074)
2	75.10(0.0907)	77.56(0.0765)	79.97(0.0882)	70.48(0.1189)	93.54(0.0225)	92.07(0.0249)	92.27(0.0333)	**95.52(0.0189)**
3	81.19(0.1039)	77.43(0.1930)	81.20(0.1384)	67.94(0.0911)	94.80(0.0391)	92.54(0.0280)	93.90(0.0448)	**95.74(0.0240)**
4	94.39(0.0497)	94.67(0.0501)	94.45(0.0367)	84.18(0.1684)	**97.12(0.0174)**	95.90(0.0521)	94.27(0.0331)	96.54(0.0017)
5	49.41(0.2408)	45.54(0.2070)	38.69(0.0392)	25.08(0.2734)	80.58(0.1590)	71.49(0.1526)	91.66(0.1144)	**92.28(0.0721)**
6	40.78(0.2277)	55.60(0.3003)	55.99(0.2997)	43.84(0.3002)	68.85(0.1019)	76.26(0.0838)	69.95(0.1091)	**94.67(0.0034)**
7	78.15(0.1375)	71.53(0.1655)	67.84(0.1066)	61.96(0.2118)	94.99(0.0164)	85.76(0.0743)	92.18(0.0497)	**95.05(0.0300)**
8	79.76(0.1155)	84.50(0.0911)	88.38(0.0786)	76.14(0.0693)	90.09(0.0326)	87.21(0.0514)	90.91(0.0318)	**95.14(0.0270)**
9	**6**8.92(0.1443)	73.40(0.0802)	76.90(0.0533)	66.12(0.1215)	84.94(0.0613)	82.47(0.0549)	89.37(0.0348)	**93.45(0.0431)**
10	87.10(0.1708)	**99.03(0.0148)**	96.41(0.0789)	86.37(0.1392)	95.07(0.0350)	97.59(0.0277)	95.18(0.0343)	97.76(0.0110)
11	74.81(0.2543)	76.29(0.1430)	83.98(0.088)	73.22(0.1495)	98.90(0.0081)	92.33(0.0240)	98.30(0.0147)	**99.20(0.0059)**
12	56.73(0.3579)	75.77(0.2721)	61.92(0.3261)	45.18(0.3039)	66.98(0.1503)	80.28(0.0816)	87.00(0.1028)	**97.71(0.0211)**
13	60.60(0.1740)	55.73(0.0854)	66.01(0.1396)	54.80(0.1274)	73.33(0.0774)	75.91(0.0880)	72.84(0.0521)	**85.19(0.0704)**
14	72.30(0.0862)	84.43(0.0322)	79.40(0.0540)	72.54(0.0862)	92.23(0.0246)	91.02(0.0298)	90.05(0.0308)	**95.22(0.0128)**
15	**81.06(0.1928)**	75.63(0.1468)	75.40(0.2790)	66.45(0.3745)	30.89(0.1193)	76.23(0.1958)	40.93(0.1562)	60.09(0.1460)
16	96.51(0.0427)	97.61(0.0325)	99.18(0.0095)	95.49(0.0508)	**99.86(0.0010)**	99.21(0.0056)	99.56(0.0028)	99.63(0.0021)
OA	82.35(0.0583)	84.00(0.0623)	87.47(0.0252)	78.48(0.0600)	94.46(0.0053)	92.89(0.0061)	94.28(0.0061)	**96.96(0.0036)**
AA	74.01(0.088)	76.92(0.0773)	77.36(0.0647)	67.08(0.0746)	85.10(0.019)	86.97(0.0154)	87.28(0.0189)	**93.10(0.0101)**
KAPPA	79.23(0.0691)	81.18(0.0739)	85.31(0.0297)	74.67(0.0709)	93.51(0.0062)	91.67(0.0072)	93.31(0.0071)	**96.44(0.0043)**
Parameter	825.513k	820.997k	1671.490k	477.832k	175.990k	2449.700k	112.042k	**99.213k**
Train time	561.33 s	455.03 s	566.82 s	354.99 s	74.28 s	**50.91 s**	201.20 s	262.09 s
Test time	175.74 s	175.06 s	233.66 s	107.87 s	9.88 s	13.53 s	**6.32 s**	10.10 s

coverage objects. In this paper, taking full advantage of CNNs and GCNs, a dual branch CEGAT network is proposed. First, a LD_SICS module is designed to eliminate the spectral redundancy information in the original data, and the feature information is mapped to the low dimensional space. Then, on the GAT branch, the structure of the super pixel encoding graph is utilized. An

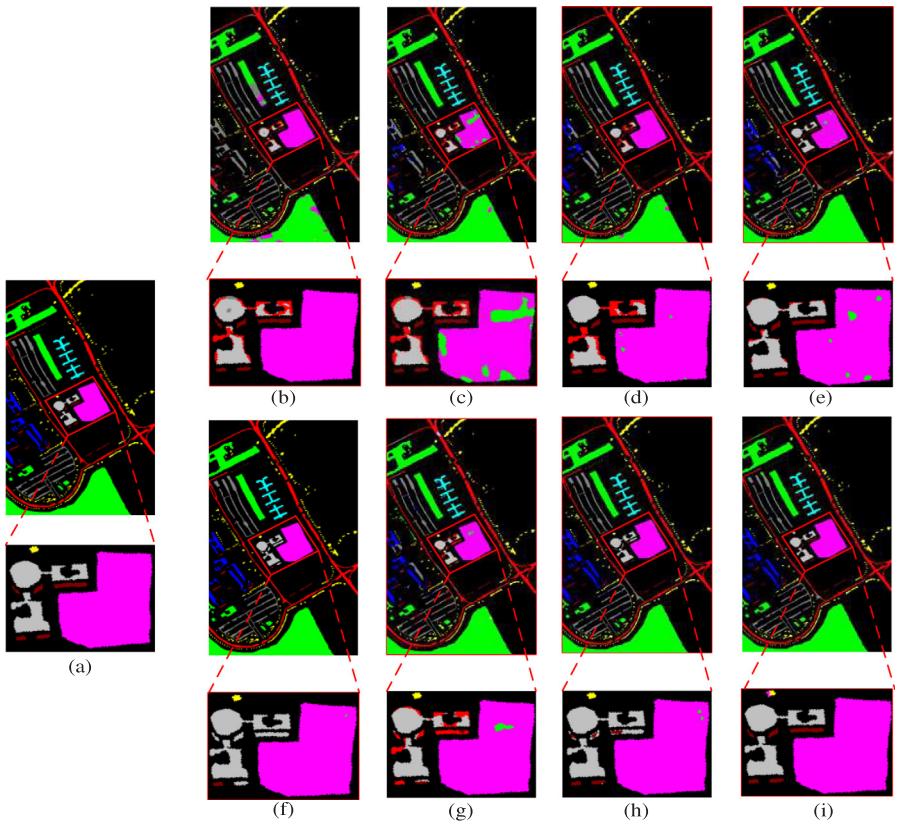


Fig. 15. Classification maps on UP. (a) Ground truth; (b) DBDA; (c) DBMA; (d) FDSSC; (e) SSRN; (f) CEGCN; (g) FDGC; (h) WFCG; (i) CEGAT.

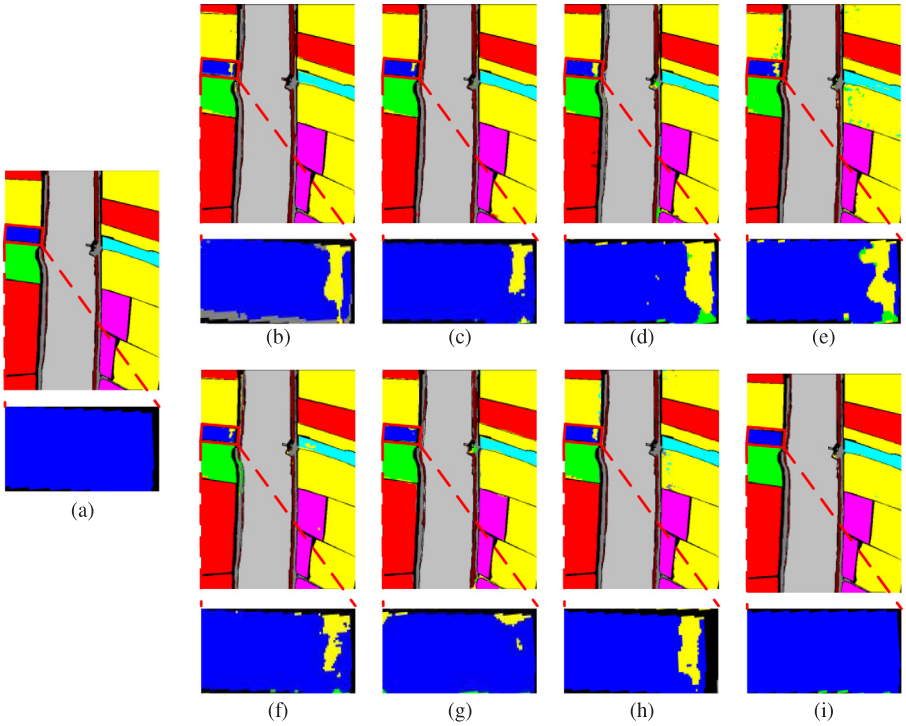


Fig. 16. Classification maps on LK. (a) Ground truth; (b) DBDA; (c) DBMA; (d) FDSSC; (e) SSRN; (f) CEGCN; (g) FDGC; (h) WFCG; (i) CEGAT.

EGAT module is constructed to enhance the network's ability to focus on node features. On the CNN branch, a SSCA module is proposed to extract more discriminative pixel level spatial spectral features and reduce the computational complexity of the network. Then, the features extracted from the two branches are fused for classification. Finally, in order to mitigate the impact of small samples on the network, a KSS strategy is proposed. The KSS strategy increases the amount of feature information of

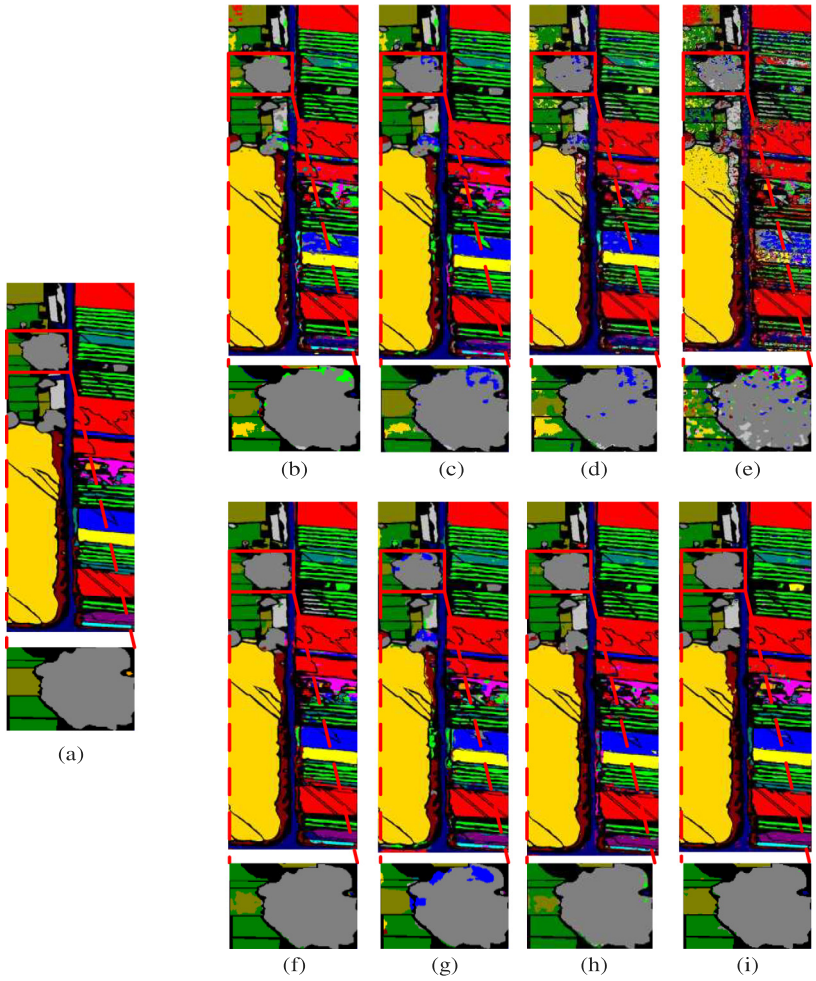


Fig. 17. Classification maps on HC. (a) Ground truth; (b) DBDA; (c) DBMA; (d) FDSSC; (e) SSRN; (f) CEGCN; (g) FDGC; (h) WFCG; (i) CEGAT.

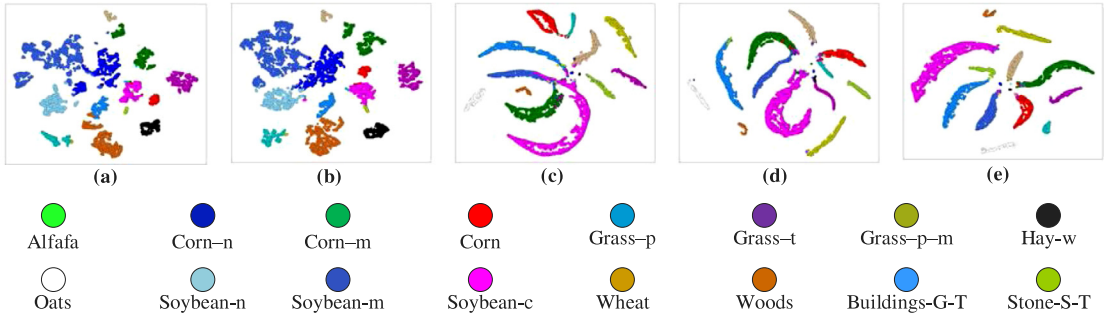


Fig. 18. Feature visualization of different methods on IN. (a)DBDA; (b) FDSSC; (c) CEGCN; (d)WFCG; (e) CEGAT.

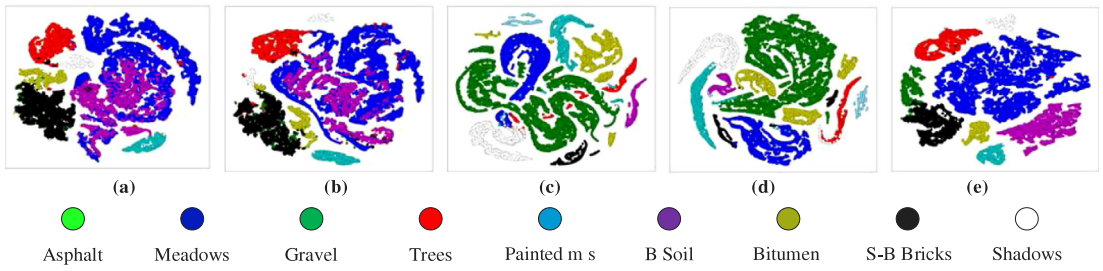


Fig. 19. Feature visualization of different methods on UP. (a) DBDA; (b) FDSSC; (c) CEGCN; (d) WFCG; (e) CEGAT.

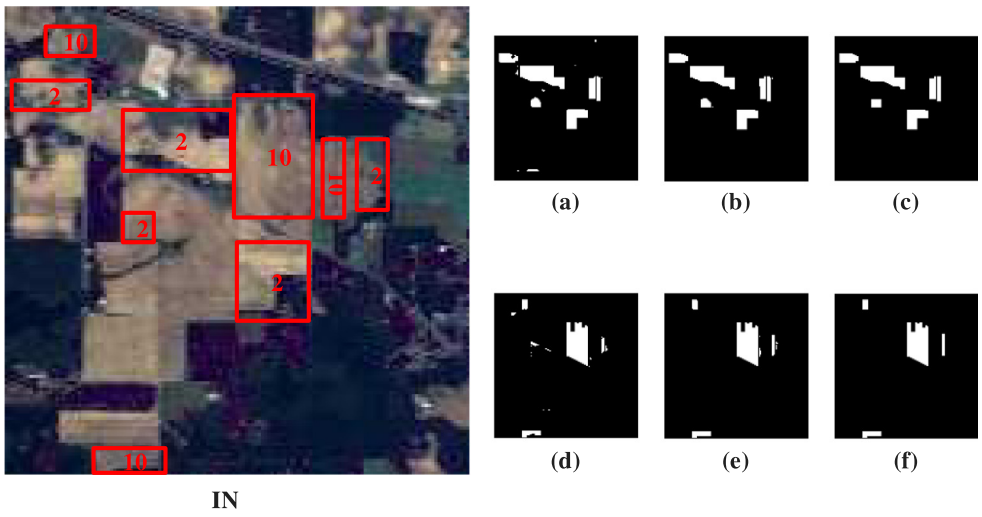


Fig. 20. Visualization of the local feature map of IN. (a) The local feature map of the 2nd class obtained by CEGAT network only removing EGAT branches; (b) The local feature map of the 2nd class obtained by the complete CEGAT network; (c) Local feature map of the real 2nd class; (d) The local feature map of the 10th class obtained by CEGAT network only removing EGAT branches; (e) The local feature map of the 10th class obtained by the complete CEGAT network; (f) Real local feature map of the 10th class.

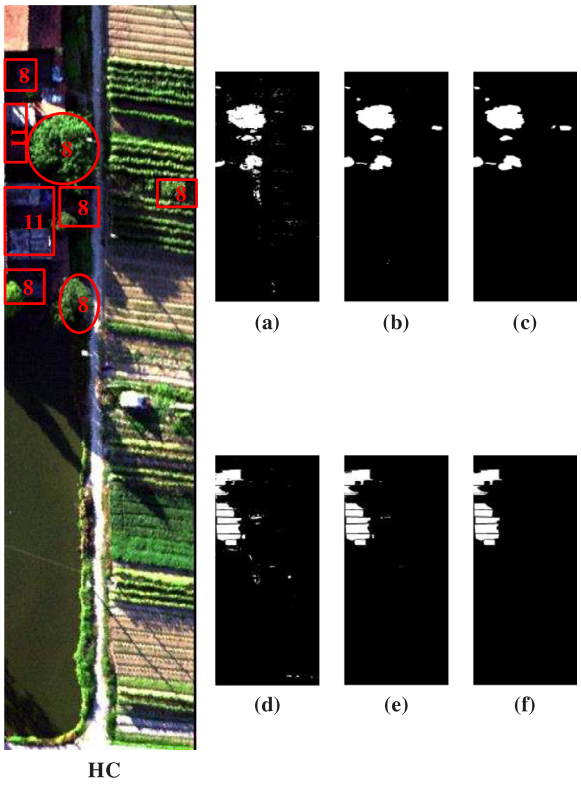


Fig. 21. Visualization of the local feature map of HC. (a) The local feature map of the 8th class obtained by CEGAT network only removing EGAT branches; (b) The local feature map of the 8th class obtained by the complete CEGAT network; (c) Local feature map of the real 8th class; (d) The local feature map of the 11th class obtained by CEGAT network only removing EGAT branches; (e) The local feature map of the 11th class obtained by the complete CEGAT network; (f) Real local feature map of the 11th class.

training samples by selecting unlabeled samples that are difficult to distinguish and adding them to the training set, so as to alleviate the problem of small samples. A large number of experimental results show that the CEGAT can achieve better classification performance than some state-of-the-art methods with few samples. In future work, we will further reduce the computational complexity of the network to realize the lightweight of the network. Moreover, considering that there may be edge feature dependencies between super-pixel blocks with different scales, this may be beneficial for HSIC. Therefore, in future work we will consider introducing multi-scale superpixels as input.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This research was funded in part by the National Natural Science Foundation of China (42271409, 62071084), in part by the Heilongjiang Science Foundation Project of China under Grant LH2021D022, in part by the Leading Talents Project of the State Ethnic Affairs Commission, and in part by the Fundamental Research Funds in Heilongjiang Provincial Universities of China under Grant 145209150.

References

Ahmad, M., et al. (2022). Hyperspectral image classification—Traditional to deep models: A survey for future prospects. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 15*, 968–999.

Chen, Y., Nasrabadi, N., & Tran, T. (2010). Classification for hyperspectral imagery based on sparse representation. In *2010 2nd workshop on hyperspectral image and signal processing: evolution in remote sensing* (pp. 1–4).

Dong, Y., Liu, Q., Du, B., & Zhang, L. (2022). Weighted feature fusion of convolutional neural network and graph attention network for hyperspectral image classification. *IEEE Transactions on Image Processing, 31*, 1559–1572.

Haut, Z., Paoletti, M., Plaza, J., Plaza, A., & Li, J. (2019). Visual attention-driven hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing, 57*(10), 8065–8080.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 37*(9), 1904–1916.

Kuo, B.-C., Ho, H.-H., Li, C.-H., Hung, C.-C., & Taur, J.-S. (2014). A kernel based feature selection method for SVM with RBF kernel for hyperspectral image classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *7*(1), 317–326.

Li, S., Song, W., Fang, L., Chen, Y., Ghamisi, P., & Benediktsson, J. (2019). Deep learning for hyperspectral image classification: An overview. *IEEE Transactions on Geoscience and Remote Sensing*, *57*(9), 6690–6709.

Li, R., Zheng, S., Duan, C., Yang, Y., & Wang, X. (2020). Classification of hyperspectral image based on double-branch dual-attention mechanism network. *Remote Sensing*, *12*, 582.

Liu, Q., Dong, Y., Zhang, Y., & Luo, H. (2022). A fast dynamic graph convolutional network and CNN parallel network for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, *60*, Article 5530215, 1–15.

Liu, Q., Xiao, L., Yang, J., & Wei, Z. (2021). CNN-enhanced graph convolutional network with pixel- and superpixel-level feature fusion for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, *59*(10), 8657–8671.

Ma, W., Yang, Q., Wu, Y., Zhao, W., & Zhang, X. (2019). Double-branch multi-attention mechanism network for hyperspectral image classification. *Remote Sensing*, *11*, 1307.

Mei, S., Chen, X., Zhang, Y., Li, J., & Plaza, A. (2022). Accelerating convolutional neural network-based hyperspectral image classification by step activation quantization. *IEEE Transactions on Geoscience and Remote Sensing*, *60*, Article 5502012, 1–12.

Mei, S., Li, X., Liu, X., Cai, H., & Du, Q. (2022). Hyperspectral image classification using attention-based bidirectional long short-term memory network. *IEEE Transactions on Geoscience and Remote Sensing*, *60*, Article 5509612, 1–12.

Mei, X., et al. (2019). Spectral–spatial attention networks for hyperspectral image classification. *Remote Sensing*, *11*(8), 963.

Mou, L., Lu, X., Li, X., & Zhu, X. (2020). Nonlocal graph convolutional networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, *58*(12), 8246–8257.

Sha, A., Wang, B., Wu, X., & Zhang, L. (2021). Semisupervised classification for hyperspectral images using graph attention networks. *IEEE Transactions on Geoscience and Remote Sensing Letter*, *18*(1), 157–161.

Shi, C., Wu, H., & Wang, L. (2023). A positive feedback spatial-spectral correlation network based on spectral slice for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, *61*, Article 5503417, 1–17.

Sun, H., Zheng, X., Lu, X., & Wu, S. (2020). Spectral–spatial attention network for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, *58*(5), 3232–3245.

Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*(11), 2579–2605.

Wan, Y., Fan, Y., & Jin, M. (2021). Application of hyperspectral remote sensing for supplementary investigation of polymetallic deposits in Huaniushan ore region, northwestern China. *Scientific Reports*, *11*, 440.

Wan, S., Gong, C., Zhong, P., Du, B., Zhang, L., & Yang, J. (2020). Multiscale dynamic graph convolutional network for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, *58*(5), 3162–3177.

Wang, W., Dou, S., Jiang, Z., & Sun, L. (2018). A fast dense spectral–spatial convolution network framework for hyperspectral images classification. *Remote Sensing*, *10*(7), 1068.

Wang, T., Wang, G., Tan, K., & Tan, D. (2020). Spectral pyramid graph attention network for hyperspectral image classification. arXiv:2001.07108.

Wang, L., Yan, J., Mu, L., & Huang, L. (2020). Knowledge discovery from remote sensing images: A review. *WIREs Data Mining Knowledge Discovery*, *10*(5), 1371.

Wang, Z., Yang, T., & Zhang, H. (2020). Land contained sea area ship detection using spaceborne image. *Pattern Recognition Letter*, *130*, 125–131.

Woo, S., Park, J., Lee, J., & Kweon, I. (2018). Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 3–19). Amsterdam, The Netherlands.

Xing, F., Yao, H., Liu, Y., Dai, X., Brown, R., & Bhatnagar, D. (2019). Recent developments and applications of hyperspectral imaging for rapid detection of mycotoxins and mycotoxigenic fungi in food products. *Critical Reviews in Food Science and Nutrition*, *59*(1), 173–180.

Yaling, W., Xiwu, Z., Hui, L., & Yurong, Q. (2021). Overview of the application of convolutional neural network in hyperspectral image classification. *Computer Engineering and Application*, *57*(4), 1–10.

Ying, L., Zhang, H., & Qiang, S. (2017). Spectral-spatial classification of hyperspectral imagery with 3D convolutional neural network. *Remote Sensing*, *9*(1), 67.

Yu, H., Kong, B., Wang, Q., Liu, X., & Liu, X. (2020). 14 - Hyperspectral remote sensing applications in soil: a review. *Earth Observation*, 269–291.

Zhang, X., Chen, S., Zhu, P., Tang, X., Feng, J., & Jiao, L. (2022). Spatial pooling graph convolutional network for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, *60*, Article 5521315, 1–15.

Zhang, L., Song, L., Du, B., & Zhang, Y. (2021). Nonlocal low-rank tensor completion for visual data. *IEEE Transactions on Cybernetics*, *51*(2), 673–685.

Zhang, L., & Zhang, L. (2022). Artificial intelligence for remote sensing data analysis: A review of challenges and opportunities. *IEEE Geoscience and Remote Sensing Magazine*, *10*(2), 270–294.

Zhang, L., et al. (2023). TN-ZSTAD: Transferable network for zero-shot temporal activity detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(3), 3848–3861.

Zhao, Z., Wang, H., & Yu, X. (2021). Spectral–spatial graph attention network for semisupervised hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing Letter*, *19*, 1–5.

Zheng, X., Sun, H., Lu, X., & Xie, W. (2022). Rotation-invariant attention network for hyperspectral image classification. *IEEE Transactions on Image Processing*, *31*, 4251–4265.

Zheng, X., Wang, B., Du, X., & Lu, X. (2022). Mutual attention inception network for remote sensing visual question answering. *IEEE Transactions on Geoscience and Remote Sensing*, *60*, Article 5606514, 1–14.

Zhong, Z., Li, J., Luo, Z., & Chapman, M. (2018). Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework. *IEEE Transactions on Geoscience and Remote Sensing*, *56*(2), 847–858.

Zhong, Z., Li, Y., Ma, L., Li, J., & Zheng, W.-S. (2022). Spectral–spatial transformer network for hyperspectral image classification: A factorized architecture search framework. *IEEE Transactions on Geoscience and Remote Sensing*, *60*, Article 5514715, 1–15.

报告编号：**L24Y2023-0595**

检　索　报　告

一、检索要求

1. 委 托 人：石翠萍 Shi, CP (Shi, Cuiping)
2. 委托单位：齐齐哈尔大学
3. 检索目的：论文被 SCI-E 收录情况

二、检索范围

Science Citation Index Expanded (SCI-EXPANDED)	1990-present	网络版
JCR-(Journal Citation Reports)	2022	网络版
中国科学院文献情报中心期刊分区表(升级版)	2022	网络版

三、检索结果

委托人提供的**1**篇论文被SCI-E收录，论文被收录、所在期刊的JCR影响因子、中科院期刊分区(升级版)情况见附件一。

特此证明！

检索报告人：杨茗惠

东北师范大学科技查新咨询中心

教育部科技查新工作站(L24)

2023 年 **11** 月 **20** 日

附件一：SCI-E收录情况

标题：CEGAT: A CNN and enhanced-GAT based on key sample selection strategy for hyperspectral image classification
作者: **Shi, CP (Shi, Cuiping)**; Wu, HY (Wu, Haiyang); Wang, LG (Wang, Liguo)
来源出版物：NEURAL NETWORKS 卷：168 页：105-122 DOI: 10.1016/j.neunet.2023.08.059 提前访问日期: SEP 2023 出版年: NOV 2023
Web of Science 核心合集中的 "被引频次": 0
被引频次合计: 0
使用次数 (最近 180 天): 6
使用次数 (2013 年至今): 6
引用的参考文献数: 39
摘要: In recent years, the application of convolutional neural networks (CNNs) and graph convolutional networks (GCNs) in hyperspectral image classification (HSIC) has achieved remarkable results. How-ever, the limited label samples are still a major challenge when using CNN and GCN to classify hyperspectral images. In order to alleviate this problem, a double branch fusion network of CNN and enhanced graph attention network (CEGAT) based on key sample selection strategy is proposed. First, a linear discrimination of spectral inter-class slices (LD_SICS) module is designed to eliminate spectral redundancy of HSIs. Then, a spatial spectral correlation attention (SSCA) module is proposed, which can extract and assign attention weight to the spatial and spectral correlation features. On the graph attention (GAT) branch, the HSI is segmented into some super pixels as input to reduce the amount of network parameters. In addition, an enhanced graph attention (EGAT) module is constructed to enhance the relationship between nodes. Finally, a key sample selection (KSS) strategy is proposed to enable the network to achieve better classification performance with few labeled samples. Compared with other state-of-the-art methods, CEGAT has better classification performance under limited label samples.(c) 2023 Elsevier Ltd. All rights reserved.
入藏号: WOS:001084659500001
PubMed ID: 37748391
语言: English
文献类型: Article
作者关键词：Deep learning; Convolution neural network; Graph convolution network; Hyperspectral image classification
KeyWords Plus: CONVOLUTIONAL NETWORKS
地址: [Shi, Cuiping; Wu, Haiyang] Qiqihar Univ, Coll Commun & Elect Engn, Qiqihar 161000, Peoples R China.
[Wang, Liguo] Dalian Nationalities Univ, Coll Informat & Commun Engn, Dalian 116000, Peoples R China.
通讯作者地址: Shi, CP (通讯作者)，Qiqihar Univ, Coll Commun & Elect Engn, Qiqihar 161000, Peoples R China.
电子邮件地址: shicuiping@qqhru.edu.cn; 2021910323@qqhru.edu.cn; wangliguo@hrbeu.edu.cn
Affiliations: Qiqihar University; Dalian Minzu University
作者识别号:
作者 Web of Science ResearcherIDORCID 号
Wu, Haiyang 0000-0003-0157-9926
出版商: PERGAMON-ELSEVIER SCIENCE LTD
出版商地址: THE BOULEVARD, LANGFORD LANE, KIDLINGTON, OXFORD OX5 1GB, ENGLAND
Web of Science Index: Science Citation Index Expanded (SCI-EXPANDED)
Web of Science 类别: Computer Science, Artificial Intelligence; Neurosciences
研究方向: Computer Science; Neurosciences & Neurology
IDS 号: U4QK0
ISSN: 0893-6080
eISSN: 1879-2782
29 字符的来源出版物名称缩写: NEURAL NETWORKS
ISO 来源出版物缩写: Neural Netw.
来源出版物页码计数: 18
基金资助致谢:
基金资助机构 授权号
National Natural Science Foundation of China
42271409

62071084
Heilongjiang Science Foundation Project of China
LH2021D022
Leading Talents Project of the State Ethnic Affairs Commission
Fundamental Research Funds in Heilongjiang Provincial Universities of China
145209150
This research was funded in part by the National Natural Science Foundation of China (42271409, 62071084) , in part by the Heilongjiang Science Foundation Project of China under Grant LH2021D022, in part by the Leading Talents Project of the State Ethnic Affairs Commission, and in part by the Fundamental Research Funds in Heilongjiang Provincial Universities of China under Grant 145209150.

输出日期: 2023-11-20

期刊影响因子 ™ 2022：7.8

中国科学院文献情报中心期刊分区(升级版, 2022)截图如下:

NEURAL NETWORKS

刊名	NEURAL NETWORKS
年份	2022
ISSN	0893-6080
Review	否
Open Access	否
Web of Science	SCIE

	学科	分区	Top期刊
大类	计算机科学	1	是
小类	COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE 计算机：人工智能	2	-
	NEUROSCIENCES 神经科学	2	

The End