Enhanced Spectral–Spatial Residual Attention Network for Hyperspectral Image Classification

Yanting Zhan, *Member, IEEE*, Ke Wu, and Yanni Dong, *Senior Member, IEEE*

***Abstract*—Deep learning has achieved good performance in hyperspectral image classification (HSIC). Many methods based on deep learning use deep and complex network structures to extract rich spectral and spatial features of hyperspectral images (HSIs) with high accuracy. During the process, how to accurately extract the features and information from pixel blocks in HSIs is important. All of the spectral features are treated equally in classification, and the input of the network often contains much useless pixel information, leading to a low classification result. To solve this problem, an enhanced spectral-spatial residual attention network (ESSRAN) is proposed for HSIC in this article. In the proposed network, the spectral-spatial attention network (SSAN), residual network (ResNet) and long-short term memory (LSTM) are combined to extract more discriminative spectral and spatial features. More specifically, SSAN is first applied to extract image features by using the spectral attention module to emphasize useful bands and suppress useless bands. The spatial attention module is used to emphasize pixels that have same category with the central pixel. Then, these obtained features are fed into an improved ResNet, which adopts LSTM to learn representative high-level semantic features of the spectral sequences, since the use of ResNet can prevent gradient disappearance and explosion. The proposed ESSRAN model is implemented on three commonly used HSI datasets and compared to some state-of-the-art methods. The results confirm that ESSRAN effectively improves accuracy.**

***Index Terms*—Hyperspectral image classification (HSIC), long-short term memory (LSTM), residual network (ResNet), spectral-spatial attention network (SSAN).**

I. Introduction

Hyperspectral images (HSIs) contain abundant of narrow and contiguous spectral bands ranging from visible to near-infrared and even thermal infrared, holding plentiful physical properties. The 3-D data block of HSIs also contains extensive detailed spatial distribution information. Both spectral signatures and spatial information can be used to accurately characterize and identify the types of objects of interest, resulting in great potential for land cover identification [1], [2], [3], [4]. Hyperspectral image classification (HSIC), aiming to identify the category of each hyperspectral pixel, has been applied to many applications, such as geological exploration [5], [6], urbanization analysis [7], precision agriculture [8], environmental monitoring [9], change detection [10], [11] and target detection [12].

In early studies of HSIC, machine learning-based methods, such as support vector machines (SVMs) [13], random forests [14], decision trees [15], neural networks (NNs) [16], and logistic regression [17] were dominant. However, these methods simply extract shallow features based on the spectral information of the HSI, using one single pixel and all of its bands as input. Thus, these linear and nonlinear classifiers do not adapt well to the high dimensionality of the spectrum, limiting their application [18]. Feature extraction (FE) methods are well adapted to the high-dimensionality of the spectrum by mapping the raw HSIs to a low-dimensional space. Some of the more advanced FE methods are geodesic-based sparse manifold hypergraph [19] and multistructure unified discriminative embedding [20], etc. These methods classify HSIs by converting them into low-dimensional structures and extracting the sparse relationships and discriminative features from different structures. When deep learning was introduced into HSIC, it achieved remarkable performance. Typical deep learning-based classification methods include deep belief networks [21], sparse autoencoders [22], recurrent neural networks (RNNs) [23], convolutional neural networks (CNNs) [24], and so on. Different from traditional machine learning algorithms, these deep learning-based methods can automatically extract high-level semantic information from HSIs with no handcrafted FE. Among them, CNN can simultaneously extract high-level spectral and spatial features by convolution, showing better classification performance. This spectral-spatial classification method has gradually developed to solve the complex spatial distribution problem in HSIs and obtain higher classification accuracy [25]. The 1-D CNN model is designed to use the pixel vector along the radiometric dimension as a training sample to extract deep features, which is conceptually called the spectral-based classification approach. A 2-D CNN, which is called a spatial-based classification approach, learns spatial information by a convolution operation on the spatial dimension. 3-D CNN combines the advantages of 1-D CNN and 2-D CNN and can extract diagnostic spectral and spatial information from 3-D hypercubes with spectral and spatial continuity, which is also called the spectral-spatial classification method [26].

The 3-D CNN takes a cube containing the target pixel and several adjacent pixels as input. There are pixels in this cube labeled differently from the central pixel. Such bands and pixels contained in this hyperspectral cube obviously have bad effects

Manuscript received 19 May 2022; revised 5 July 2022; accepted 2 August 2022. Date of publication 10 August 2022; date of current version 6 September 2022. This work was supported in part by the Natural Science Foundation of China under Grants U21A2013, 62171417, and 62071438 and in part by the Global Change and Air-Sea Interaction II under Grant GASI-01-DLYG-WIND01. *(Corresponding author: Ke Wu.)*

The authors are with the Institute of Geophysics and Geomatics, China University of Geosciences, Wuhan 430074, China (e-mail: z1606229857@163.com; tingke2000@126.com; dongyanni@cug.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2022.3197934

on the CNN classification [27]. Therefore, it is critical to harvest the information bands and pixels that are beneficial to HSIC in the end-to-end classification process. Such information that facilitates classification should be focused on, while bands with redundant information and pixels with different labels from the target pixels should be suppressed. To solve this problem, the spectral-spatial attention (SSA) mechanism is used in HSIC to learn dependent spectral and spatial features. SSA is composed of a spectral attention (SpeA) module and a spatial attention (SpaA) module. It assigns high weights to useful bands and pixels for feature enhancement of the original image [28]. The attention mechanism comes from the study of human vision, where people selectively focus on useful information of interest and ignore other visible information. Thus, this mechanism increases the sensitivity to features that contain the most valuable information. It was first applied to machine translation [29] and later was also widely used in natural language processing [30], image recognition [31], [32] and speech recognition [33]. Mei first introduced the SSA mechanism into hyperspectral classification to capture high spectral correlation between adjacent spectra and learn spatial dependence in the spatial domain [34]. Later, the attention mechanism was improved or combined with other network structures to improve the classification accuracy of HSI [35]. Pan et al. [36] designed a joint network with a spectral attention bidirectional RNN branch and a spatial attention CNN branch to extract spectral and spatial features for HSIC. Zhu et al. [37] embedded a SSA module into a residual block to avoid overfitting and accelerate the training speed. Lu et al. [38] used a multiscale spatial-spectral residual network to stack the extracted deep multiscale features and input them into the 3-D attention module to improve the classification accuracy. Some researchers combined SSA with graph convolutional networks (GCNs) [39] to adaptively extract spatial and spectral features from neighboring nodes through a graph attention mechanism [40], [41]. It is evident that it is highly feasible and advantageous to use SSA to extract spectral and spatial dependent features and then input them into a deep network model for classification.

The depth of the network is critical to the performance of most models. When the number of network layers is increased, the network extracts more complex features, so theoretically better results could be achieved. However, many experiments have shown that as the depth of the network increases, the CNN model exhibits degradation problems, leading to poor results. Thus, He et al. [42] proposed the residual network (ResNet) on the classification task of ImageNet large scale visual recognition challenge (ILSVRC) 2015. The main contribution of ResNet is the discovery of "degradation" and the invention of a "shortcut connection" aimed at the degeneracy phenomenon, which greatly eliminates the problem of training difficulty in deep NNs. Subsequently, ResNet has been added to deep network models by many scholars to classify HSIs in combination with CNNs. Jiang et al. [43] collaborated on the 3-D separable ResNet with cross-sensor transfer learning to reduce training parameters and achieve better classification performance. Meng et al. [44] proposed a multipath ResNet that employed multiple residual functions in the residual blocks to make the network wider. Li et al. [45] proposed a depthwise separable ResNet, which can separate both spectral and spatial information and also greatly reduce the network size. The residual network will continue to be used in HSIC due to its powerful feature transformation capability.

Although SSA and ResNet have powerful FE and generalization capabilities, the potential relationships between adjacent bands are ignored, resulting in important spectral features to be undetected. ResNet inputs the SSA transformed 3-D feature maps as a whole into the model for training, and connects the upper-level nodes with the lower-level nodes with weights; thus, it ignores the relationship between the nodes of the same layer. Long-short term memory (LSTM), a deep learning algorithm mainly used to handle sequence data, can solve this problem. The aim of the LSTM is to give a typically strong relationship between the given sample and the previous one, where activation at each step depend on the previous step in the hidden layer [46]. The simplest way to classify HSIs using LSTM is to use each band of the pixel spectrum as input data at the corresponding time, serialize the spectral vector band by band, and then extract potential information. Zhou et al. [47] input row vectors of image blocks centered on target pixels into the LSTM model for hyperspectral classification. Liu et al. [48] proposed a bidirectional-convolutional LSTM network to automatically learn the spectral and spatial features from HSI. Tang et al. [49] combined the GCNs with bidirectional LSTM to extract both short and long spatial relationships for HSIC. It can be seen that the addition of LSTM to the network model of HSIC is helpful to improve the classification accuracy.

Based on the above analysis, we propose an enhanced spectral-spatial residual attention network (ESSRAN) algorithm for HSIC. Moreover, small training samples are selected to test the network, which fully demonstrates the advantages of the proposed method, and the pixel cluster (PC) approach is used to solve the problem of insufficient number of training samples for some categories. This network combines the advantages of SSA, ResNet and LSTM, improving the capabilities of spectral and spatial feature learning and the accuracy of classification. The main contributions of this article are as follows.

1) For the problem that hyperspectral cubes often contain redundant pixels and bands, the SSA module is applied to extract discriminative and robust spectral and spatial features. In the spectral dimension, it generates a spectral weight vector emphasizing useful bands to improve the performance of classification. In the spatial dimension, it adaptively emphasizes the spatial information of pixels with the same label as the central pixel by generating a spatial weight matrix that represents the significance of neighborhood pixels.
2) To extract potential relationships between adjacent bands, LSTM is added to the ResNet module to obtain the interdependence of long-range nonlinear channels. Specifically, convolution and LSTM operations are used in ResNet to extract the required spectral and spatial information. The feature map after convolution is produced as spectral sequence data, which is then fed into the LSTM to obtain the relationship between the bands.
3) To adapt small samples of HSIC, we used the PC method to expand the training samples. This method regroups the training samples in order to obtain new pixel blocks. These

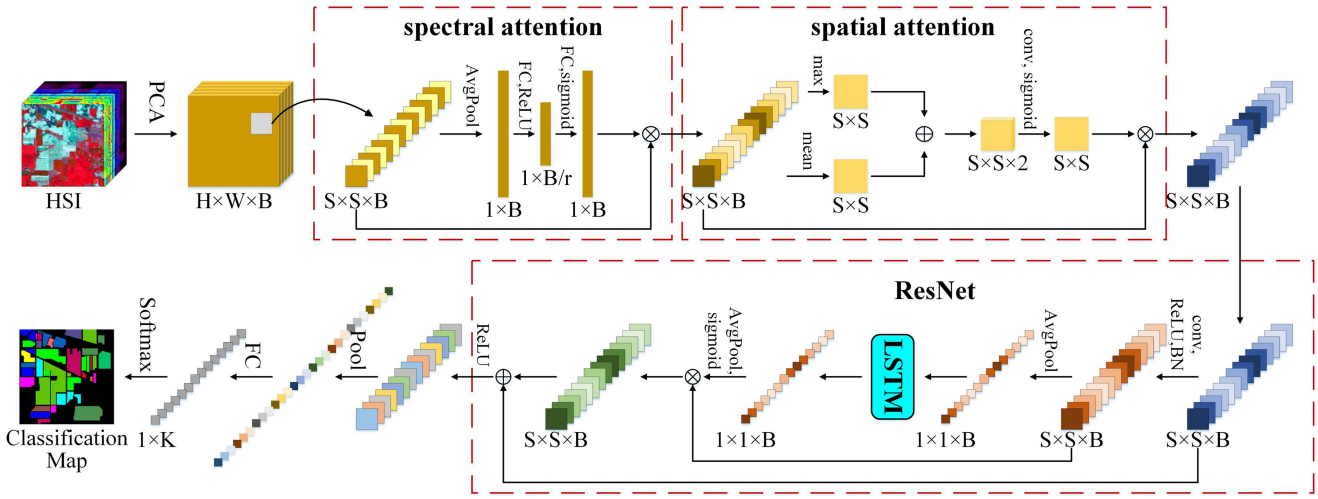

Fig. 1. Framework of the proposed network for HSIC. First, select a 3-D patch cube (i.e., z) from the PCA-converted HSI. S denotes the spatial size of z, and B is the number of spectral bands. Then, SSA extracts spectral and spatial features from z and feeds the features into ResNet with LSTM added for training. Finally, the classification is performed using the softmax layer.

pixel blocks are superimposed on the spectral dimension so that the new data block is of the same size as the original one. This method effectively improves the classification accuracy for classes with small number of samples.

4) We experimentally demonstrate the effectiveness of the proposed deep network modules and illustrate that the proposed ESSRAN outperforms eight compared methods on three HSI datasets.

The rest of this article is organized as follows. Section II introduces the proposed method. Section III evaluates the effectiveness of the proposed method on real hyperspectral datasets, and Section IV draws the conclusion.

II. Proposed Method

In this section, the framework of the proposed method for HSIC is first described in detail. Second, each basic model in the network is introduced in turn, including SSA, ResNet, and LSTM. Finally, a pixel-cluster-based training sample increasing method is presented in detail.

A. Overview of the Proposed Model

Let $X_{\text{hsi}} \in R^{H\times W\times B}$ represent the original HSI data, where H, W, and B represent the height and width of spatial dimensions and the number of spectral bands, respectively. Suppose that the dataset X_{hsi} contains N labeled pixels $X = \{x_1, x_2, \ldots, x_N\} \in R^{1\times 1\times B}$, and their corresponding set of one-hot label vectors $Y = \{y_1, y_2, \ldots, y_N\} \in R^{1\times 1\times K}$, where K is the number of classes. The regions of size $S \times S$ centered at pixel x can be defined as a spectral-spatial vector $Z = \{z_1, z_2, \ldots, z_N\} \in R^{S\times S\times B}$. In this article, each patch cube z_i in Z is used as input to the proposed model to classify its corresponding center pixel x_i in HSI [50].

After the notation of HSI data, all available labeled data are randomly divided into training and test datasets denoted by Z_{train} and Z_{test}, respectively, and corresponding label sets are denoted by Y_{train} and Y_{test}, respectively. Then, Z_{train} is used to optimize the hyperparameters of the proposed model and obtain the best-trained model through cross-validation. Finally, the best-trained model is used to obtain three evaluation metrics of performance by Z_{test} and classify all pixels to form a classification map.

Fig. 1 shows the framework of the proposed ESSRAN network. First, the principal component analysis (PCA) algorithm is used to perform feature transformation on the original HSI, and then a pixel-centric 3-D patch is extracted as the input of the proposed network [51]. Second, SSA is used to extract spectral and spatial features. The spectral attention module assigns a greater weight to the key channels and smaller weight to the less important channels. The spatial attention module similarly uses the weight matrix to enhance the information of pixels with the same label as the central pixel and weaken those different labels. Third, the improved ResNet, which adopts LSTM, that has a strong ability to capture contextual information in the spectral sequence, is used to extract more representative and discriminative semantic features. Finally, a fully connected layer with a softmax function is used for classification.

B. Spectral-Spatial Attention Network

The SSA network (SSAN) extracts deep spectral and spatial features from patch cube z by enhancing useful information and suppressing the effects of interfering information. This is actually an adaptive attention mechanism, which extracts the weight vector w from the patch cube z itself. The weight vector is a significant spectral and spatial feature. Final SSAN output is represented by f_{SSAN}. The detail is formulated as follows:

$$f_{\text{SSAN}}(z) = \sigma\left(w + b_{\text{SSAN}}\right) \otimes z \tag{1}$$

where $\sigma(.)$ represents the activation function, b_{SSAN} represents the bias, and $\otimes$ represents the matrix multiplication. SSAN is composed of two modules: the spectral attention module and the spatial attention module, as represented in Fig. 2. The spectral

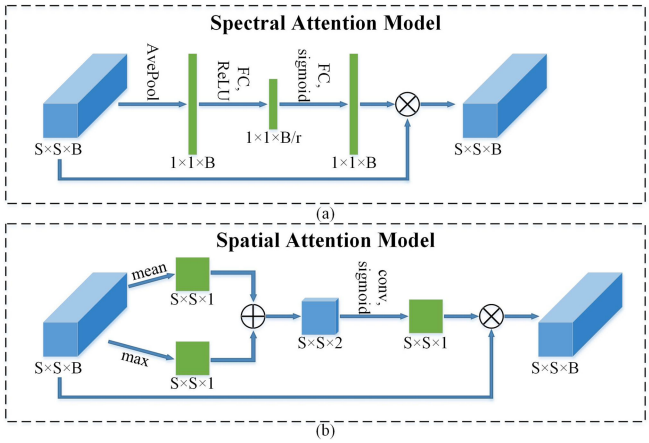

Fig. 2. Two parts of SSAN. (a) Spectral attention model. (b) Spatial attention model.

attention module is utilized to extract spectral features from the patch cube z. The spatial attention module is utilized to capture spatial features from the output of the spectral attention module [52].

1) Spectral Attention Module: The SpeA mechanism emphasizes the spectral band, which helps in the extraction of features and the final classification. The SpeA module is abstracted into three procedures: feature aggregation, feature transformation, and feature enhancement [53].

Feature aggregation calculates the average value of the patch cube z in the spatial dimension as the weight of the corresponding spectral dimension. Specifically, the input feature maps $F^{\text{spec\_in}} \in R^{S\times S\times B}$ are fed into an average pool layer and a new feature map $F^{\text{spec1}} \in R^{1\times 1\times B}$ is obtained

$$F^{\text{spec1}} = \frac{1}{S\times S}\sum_{i=0}^{S-1}\sum_{j=0}^{S-1} F_{i,j}^{\text{spec\_in}}. \tag{2}$$

Feature transformation learns nonlinear channelwise inner relationships by a multilayer perceptron (MLP) module. The MLP module has two linear fully connected layers FC, a ReLU activation function σ_{ReLU}, and a sigmoid activation function σ_{sigmoid}. The bottleneck ratio r of MLP is set to 2 to reduce the computational cost and prevent overfitting. The feature transformation operation function in MLP is expressed as

$$F^{\text{spec2}} = \sigma_{\text{sigmoid}}\left(\text{FC}\left(\sigma_{\text{ReLU}}\left(\text{FC}\left(F^{\text{spec1}}\right)\right)\right)\right) \tag{3}$$

where $F^{\text{spec2}} \in R^{1\times 1\times B}$ represents the output spectral attention map.

Feature enhancement multiplies the converted spectral features F^{spec2} with the original input $F^{\text{spec\_in}}$ to obtain the feature map with enhanced spectral information $F^{\text{spec\_out}} \in R^{S\times S\times B}$

$$F^{\text{spec\_out}} = F^{\text{spec2}} \otimes F^{\text{spec\_in}}. \tag{4}$$

2) Spatial Attention Module: The SpaA mechanism enhances spatial information from the neighborhood pixels with the same class label as the center pixel while it suppresses the information from those with different labels. Similar to SpeA, SpaA also has three procedures [54].

Feature aggregation extracts the average and maximum values of each pixel spectrum from the input feature maps $F^{\text{spa\_in}} \in R^{S\times S\times B}$ and obtains new feature maps $F^{\text{spa1}} \in R^{S\times S\times 1}$ and $F^{\text{spa2}} \in R^{S\times S\times 1}$, respectively,

$$F^{\text{spa1}} = \frac{1}{B}\sum_{b=0}^{B-1} F_b^{\text{spa\_in}} \tag{5}$$

$$F^{\text{spa2}} = \max\left(F^{\text{spa\_in}}\right) \tag{6}$$

where max represents the maximum operation.

Feature transformation connects the above two feature maps horizontally as the input of a new convolutional layer followed by a sigmoid activation function, obtaining the output attention map $F^{\text{spa3}} \in R^{S\times S\times 1}$

$$F^{\text{spa3}} = \sigma_{\text{sigmoid}}\left(\left[F^{\text{spa1}}, F^{\text{spa2}}\right] * k\right) \tag{7}$$

where $*$ is the convolution operation and k is the convolution kernel.

Finally, feature enhancement combines the attention map F^{spa3} and input map $F^{\text{spa\_in}}$ and obtains the output map $F^{\text{spa\_out}} \in R^{S\times S\times B}$

$$F^{\text{spa\_out}} = F^{\text{spa3}} \otimes F^{\text{spa\_in}}. \tag{8}$$

$F^{\text{spa\_out}}$ contains the spatial features of all the positions and highlights the information of important spatial locations.

C. Modified Residual Network

ResNet is proposed to solve the problem that the accuracy of CNN decreases substantially with increasing network depth. CNN is an NN that extracts nonlinear spectral and spatial features through convolution, pooling and activation functions. The convolution layer uses convolutional operations to extract deep features in spectral and spatial dimensions; the pooling layer can reduce the complexity of the network and improve the computational speed, including average pooling and maximum pooling; and the activation function can improve the ability of CNN to deal with nonlinear problems, such as the sigmoid function and the ReLU function [55]. Therefore, it is difficult to achieve a constant transformation between the nonlinear feature map extracted by the deep CNN and the desired label. ResNet connects the original feature map x with the optimized feature map $F(x)$, seeking a balance between linear and nonlinear transformations

$$H(x) = F(x) + x \tag{9}$$

where $H(x)$ is the desired underlying feature map [42].

To propagate information backward and forward in the network, deep ResNet is formed by stacking multiple BasicBlocks together. One BasicBlock contains two convolution layers, two batch normal layers and two activation functions, while the modified ResNet only retains half of these operations. The feature map after convolution and batch normalization operations is input to the LSTM module to obtain the contextual relationships between adjacent spectra

$$x_1 = \text{batch\_norm}(\text{conv}(x)) \tag{10}$$

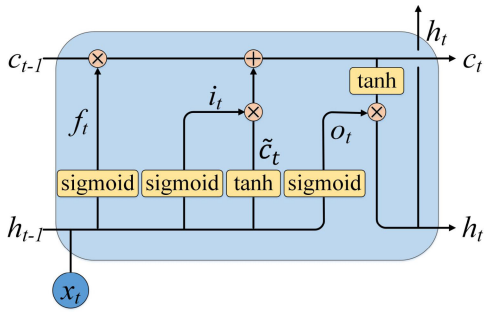

Fig. 3. Structure of LSTM.

$$F(x) = x_1 * \sigma_{\text{sigmoid}}(\text{LSTM}(x_1)) \tag{11}$$

x_1 refers to the feature map extracted by the convolution operation, batch_norm is batch normalization, and conv is the convolution. The channel relationship characteristic obtained by LSTM is multiplied by x_1 to obtain the final feature map. Considering the entire network proposed in this article, spectral and spatial features are extracted using only one residual operation, reducing the redundancy of the original structure. This not only effectively utilizes the advantages of ResNet, but also reduces the computation time and increases the discriminative power of the model.

D. Long-Short Term Memory

LSTM overcomes the problem of gradient explosion or vanishing of RNNs when dealing with long sequence data. LSTM has a chain-like structure, including a forget gate, input gate and output gate. The forget gate decides whether to consider the previous cell state; the input gate decides what new information is stored in the cell state; and the output gate regulates the amount of data passed to the next layer. The cell state carries information from the first timestep to the last timestep, i.e., the footprint of all inputs. Gates have one sigmoid activation function, where 0 indicates forget. The structure of LSTM is shown in Fig. 3. It can be observed that the LSTM cell constantly updates the hidden value and cell value with the help of the three control gates, which are used to discard, retain, or amplify signals to achieve information control and transformation. The calculation process in one LSTM cell at time t is

$$\begin{cases} f_t = \sigma_{\text{sigmoid}}\left(w_{xf}x_t + w_{hf}h_{t-1} + b_f\right) \\ i_t = \sigma_{\text{sigmoid}}\left(w_{xi}x_t + w_{hi}h_{t-1} + b_i\right) \\ \tilde{c}_t = \sigma_{\tan h}\left(w_{xc}x_t + w_{hc}h_{t-1} + b_c\right) \\ c_t = i_t \odot \tilde{c}_t + f_t \odot c_{t-1} \\ o_t = \sigma_{\text{sigmoid}}\left(w_{xo}x_t + w_{ho}h_{t-1} + b_o\right) \\ h_t = o_t \odot \sigma_{\tan h}(c_t) \end{cases} \tag{12}$$

where f_t, i_t, and o_t represent the forget gate, input gate, and output gate respectively. $\tilde{c}_t$ represents cell value. x_t, h_t, and c_t represent the input, hidden, and cell states at time step t, respectively. b_f, b_i, b_c, and b_o are bias terms. The weight matrix subscripts have conventional meanings. For instance, w_{xo} is the input–output gate matrix and w_{hi} is the hidden-input gate matrix. $\sigma(.)$ is the activation function, and $\odot$ is a dot product operator, meaning pixelwise multiplication [56], [57]. In HSIC, the spectral vector is serialized band-by-band, and each spectral band is used as input data for the LSTM model at the corresponding time, extracting relationship information between the bands.

E. Theory of Pixel Cluster

In HSIC, problems such as high imbalance between the number of samples of categories and few known labels for some of the features are very common [58]. The small number of training samples of HSI limits the learning ability of deep learning-based models, which makes it difficult to extract the typical features and affects the classification accuracy. Therefore, the PC algorithm is proposed to solve this problem. Pixel clustering is a process of increasing the number of samples using the principle of permutation. This method selects multiple pixel blocks to be combined after disrupting the training samples, and then forms a new data block. A superposition operation is performed on the selected multiple data blocks in the spectral dimension so that the new data block is of the same size as the original block.

Suppose there is a class that has n training samples. One PC is composed of p pixels, which are randomly selected from the training samples. The number of training samples after data augmentation is

$$n' = \frac{n!}{p!\,(n-p)!}. \tag{13}$$

It is obvious that n' is larger than n when $p \neq 1$, solving the shortage of the training set. In addition, the deep learning model can learn more diverse spatial information from the expanded training samples [59]. For categories with a large number of samples, sample expansion using the PC principle would lead to data redundancy and reduced accuracy. Therefore, categories with sample sizes below-average are selected for the pixel clustering operation to improve the classification performance of the network. The effects of using PCs will be explained in detail in the experimental section.

III. Experimental Results

In this section, we first introduce three experimental datasets and three factors that obviously influence the performance of the proposed model. After that, the results are compared with some state-of-the-art deep learning methods, fully proving the advantages of the proposed algorithm. Finally, the effects of SSA, LSTM, and PCs on the model are discussed separately.

A. Datasets

Three common HSI datasets, i.e., Indian Pines (IP), Pavia University (PU), and Salinas (SA), are considered in our experiments, as given in Table I. The numbers and names of each category, the number of training samples, and the total number of category samples for each of the three datasets are given in Table II. The false color image, ground truth map, and color code are depicted in Fig. 4.

1) *Indian Pines*: This dataset was captured by an airborne visible infrared imaging spectrometer (AVIRIS) sensor

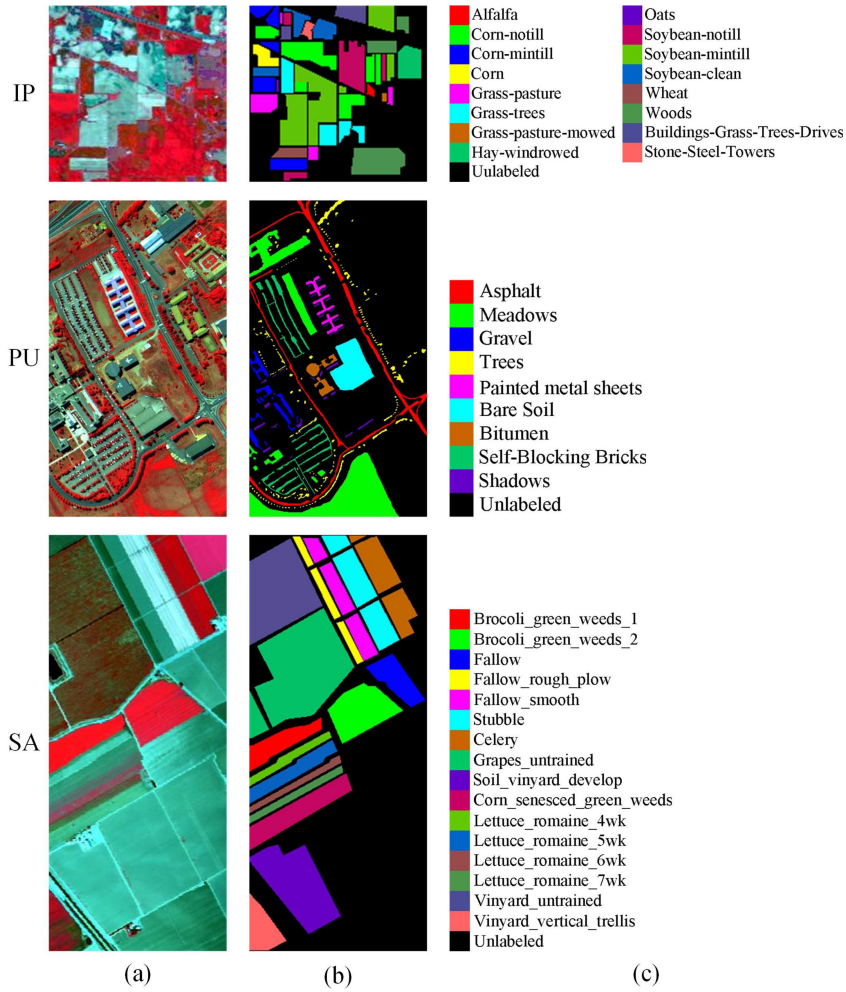

Fig. 4. Graphical illustration of IP, PU and SA. (a) False-color map. (b) Ground-truth map. (c) Color code.

TABLE I
LIST OF THREE DATASETS

Data	Sensor	Time	Spatial	Spectral
IP	AVIRIS	1992.6	145×145	200
PU	ROSIS	2002.7	610×340	103
SA	AVIRIS	-	512×217	204
Data	Wavelength/μm		resolution/m	class
IP	0.40~2.45		20	16
PU	0.43~0.86		1.3	9
SA	0.36~2.50		3.7	16

in Northwestern Indiana on June 1992. It contains 145 × 145 pixels with 20 m spatial resolution, and there are 224 spectral bands in the wavelength range of 0.4–2.45 μm. After removing the bands affected by atmospheric absorption, 200 bands are used for classification. The ground truth contains 16 vegetation classes with 10 249 labeled pixels.

2) *Pavia University*: This dataset was acquired by a reflective optics system imaging spectrometer (ROSIS) sensor over PU, Northern Italy, in July 2002. There are 103 spectral bands in the spectral range from 0.43 to 0.86 μm obtained by removing several noise-corrupted bands. It contains 610 × 340 pixels with a 1.3 m spatial resolution. This dataset contains nine distinguishable urban classes.

3) *Salinas*: The SA dataset was acquired by an AVIRIS sensor over the SA Valley, California, USA. It contains 512 × 217 pixels with a 3.7 m spatial resolution and 224 bands in the spectral range of 0.36–2.5 μm. Similar to the IP scene, 20 water-absorbing bands were discarded, and 204 bands were retained. In addition, it contains 16 ground-truth classes.

B. Experimental Settings

We evaluated the performance of the proposed network model on a server with an NVIDIA GeForce RTX 3090 GPU with 24 GB RAM. The code implementation of all methods is based on Python 3.6 with the library of PyTorch 1.7. Several evaluation indicators, including class-specific accuracy, overall accuracy (OA), average accuracy (AA), and kappa coefficient (kappa), are used to evaluate the proposed method exactly. Approximately

TABLE II
NUMBER OF TRAINING AND TOTAL SAMPLES OF THE THREE DATASETS

Indian Pines				Pavia University				Salinas			
No.	**Categories**	**Train**	**Total**	**No.**	**Categories**	**Train**	**Total**	**No.**	**Categories**	**Train**	**Total**
C1	Alfalfa	5	46	C1	Asphalt	66	6631	C1	Brocoli_green_weeds_1	20	2009
C2	Corn-notill	71	1428	C2	Meadows	186	18649	C2	Brocoli_green_weeds_2	37	3726
C3	Corn-mintill	41	830	C3	Gravel	20	2099	C3	Fallow	19	1976
C4	Corn	11	237	C4	Trees	30	3064	C4	Fallow_rough_plow	13	1394
C5	Grass-pasture	24	483	C5	Painted metal sheets	13	1345	C5	Fallow_smooth	26	2678
C6	Grass-trees	36	730	C6	Bare Soil	50	5029	C6	Stubble	39	3959
C7	Grass-pasture-mowed	5	28	C7	Bitumen	13	1330	C7	Celery	35	3579
C8	Hay-windrowed	23	478	C8	Self-Blocking Bricks	36	3682	C8	Grapes_untrained	112	11271
C9	Oats	5	20	C9	Shadows	9	947	C9	Soil_vinyard_develop	62	6203
C10	Soybean-notill	48	972					C10	Corn_senesced_green_weeds	32	3278
C11	Soybean-mintill	122	2455					C11	Lettuce_romaine_4wk	10	1068
C12	Soybean-clean	29	593					C12	Lettuce_romaine_5wk	19	1927
C13	Wheat	10	205					C13	Lettuce_romaine_6wk	9	916
C14	Woods	63	1265					C14	Lettuce_romaine_7wk	10	1070
C15	Buildings-Grass-Trees-Drives	19	386					C15	Vinyard_untrained	72	7268
C16	Stone-Steel-Towers	5	93					C16	Vinyard_vertical_trellis	18	1807
Total		**517**	**10249**	**Total**		**423**	**42776**	**Total**		**533**	**54129**

TABLE III
ORIGINAL AND INCREASED NUMBER OF TRAINING SAMPLES

class		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	total
IP	trainA	5	71	41	11	24	36	5	23	5	48	122	29	10	63	19	5	517
	trainB	10	71	41	55	276	36	10	253	10	48	122	406	45	63	171	10	1627
PU	trainA	66	186	20	30	13	50	13	36	9								423
	trainB	66	186	190	435	78	50	78	630	36								1749
SA	trainA	20	37	19	13	26	39	35	112	62	32	10	19	9	10	72	18	533
	trainB	190	37	171	78	325	39	35	112	62	496	45	171	36	45	72	153	2067

5% of the samples are randomly selected as the training set for IP, and 1% of the samples are randomly selected as the training set for PU and SA. For categories with fewer samples, at least five samples are randomly selected for the training set, while other samples are used as the test set. Each experiment is optimized for 100 epochs for the training samples. Each experiment is repeated five times to eliminate bias from randomly selected training samples and the AA and the standard deviation of each evaluation criterion are reported. In addition, the batch size is set to 32 [60].

To solve the problem having inadequate number of labeled hyperspectral datasets, experiments are performed using the PC method to add new samples. The number of added training samples are given in Table III. In the table, train A denotes the original training sample, and train B denotes the extended training sample. The number of samples is added only for categories with training samples smaller than the mean; otherwise, the original training samples are used for training. As seen from the table, the total number of training samples increased 3 to 4 times when compared to the originals.

We compared the proposed ESSRAN model with eight representative state-of-the-art HSIC methods: SVM, LSTM [23], 3-D CNN [61], HybridSN [62], DHCNet [63], GCN [64], RSSAN [37], and A2S2K-ResNet [1]. The above methods are described in detail as follows.

1) *Support Vector Machine:* A classical machine learning algorithm using kernel functions. The implementation is based on libsvm.
2) *Long-Short Term Memory:* A method for extracting spectral features by converting spectral values into sequence data.
3) *3-D Convolutional Neural Network*: A method for directly extracting spectral and spatial information using 3-D convolutional operations. This method includes a 3-D convolutional layer and a fully connected layer.
4) *HybridSN*: This method is a hybrid spectral CNN that combines 3-D CNN extracting spectral and spatial features with 2-D CNN extracting spatial abstract features.
5) *DHCNet*: This method introduces the deformable convolutional sampling locations based on 2-D CNN, whose size and shape can be adaptively adjusted according to the complex spatial contexts of HSI.
6) *Graph Convolutional Network*: This method classifies HSIs by encoding them into graphs and using superpixels instead of pixels as nodes to simulate various spatial structures of land cover on the graphs.
7) *RSSAN:* This method first uses SSAN to extract spectral and spatial information, and then embeds the attention mechanism into ResNet to accelerate model training and extract features for classification.

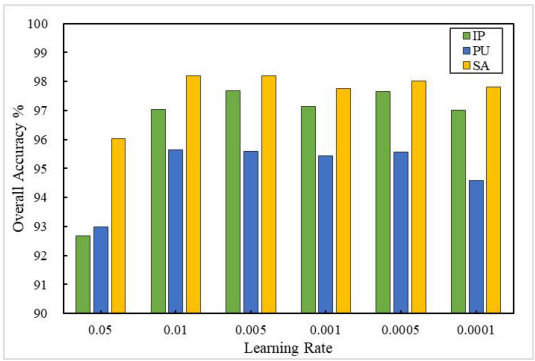

Fig. 5. OA of ESSRAN with different learning rates in the IP, PU, and SA datasets.

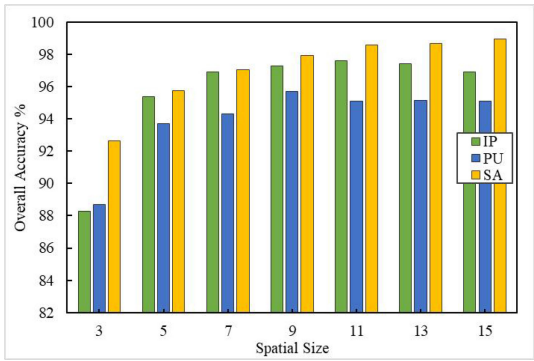

Fig. 6. OA of ESSRAN with different spatial size in the IP, PU, and SA datasets.

8) *A2S2K-ResNet*: This method improves ResNet based on RSSAN, which extracts spectral and spatial features using selective 3-D convolution kernels and improved 3-D residual blocks, and adopts an efficient feature recalibration mechanism to improve classification performance.

C. Parameter Setting

In this part, three pivotal factors that influence the training progress and classification performance of the proposed model are analyzed. These factors are the learning rate, spatial size, and training size, which are called hyperparameters.

1) *Learning Rate*: The learning rate controls the rate of gradient descent and affects the convergence in training progress. A grid search approach is used to find the best learning rate of the proposed model on each dataset. Here, we consider the learning rate sets {0.05, 0.01, 0.005, 0.001, 0.0005, 0.0001}. The results of ESSRAN with different learning rates in the three datasets are shown in Fig. 5. Based on the above results, the highest accuracy is achieved for the IP dataset when the learning rate is 0.005. For the PU and SA datasets, the highest precision learning rate is 0.01.
2) *Spatial Size*: The spatial size determines how much spatial information is used for FE around the target pixel. Thus, a large set of spatial input sizes {3, 5, 7, 9, 11, 13, 15} is used to evaluate the influence on the performance of the ESSRAN. As shown in Fig. 6, the accuracy of the PU dataset reaches its highest value when the space size is 9 $\times$9 and then decreases as the spatial size increases. For the IP dataset, the accuracy increases smoothly until the spatial size is approximately 11 $\times$ 11. For the SA dataset, the larger the spatial size is, the higher the accuracy. It follows that a data cube with a small spatial size cannot be extracted with sufficient spatial information, while a large spatial size affects the classification accuracy due to the presence of other categories at the edges. Consequently, we choose a spatial size of 9 $\times$9 for the later classification experiments.
3) *Training Size*: The number of training samples plays a decisive role in supervised HSIC. Therefore, we analyzed the effect of different training sample sizes on the OA. 1%, 3%, 5%, 10%, 15%, and 20% of labeled pixels are selected as the training set to train the ESSRAN. As shown in Fig. 7, the OA increases as the training size increases for all three HSI datasets and all algorithms. Compared with the other eight methods, the proposed method performs the best on most of the training sizes. It is more obvious on the IP dataset that the accuracy obtained by ESSARN is significantly higher than other methods when the training samples are small.

D. Classification Results

The experimental results of the IP dataset are shown in Fig. 8. To clearly show the difference, we place a local enlarged patch in the corner of each result map, and the same for the PU and SA datasets. The proposed ESSRAN method obtains the best classification results visually, with nearly no misclassification. Both A2S2K-ResNet and GCN show impressive results, but GCN shows some consecutive misclassifications at the edges. Among the remaining methods, the CNN-based 3-D CNN, HybridSN, DHCNet, and RSSAN give better classification results than SVM and LSTM. Table IV gives the average OAs, AAs, and kappas (and their standard deviations based on five runs) of the IP dataset. It can be clearly seen that the ESSRAN has the highest OA, AA, and kappa among the nine methods. The average OA of the ESSRAN is 97.69%, AA is 97.19%, and kappa is 97.37%. Three metrics of ESSRAN also have the smallest standard deviation among all methods. The standard deviation of OA is only 0.13%, indicating that the method has the highest stability. In addition, the ESSRAN method achieves the highest classification accuracy in 11 of the 16 classes due to the extraction of more discriminative spatial and spectral features. The class-specific samples in the IP dataset are highly imbalanced. Four feature types (C1, C7, C9, and C16 respectively) have less than 100 labeled samples and only 5 training samples, in which case ESSRAN achieves the highest classification accuracy. In particular, the OA of the "oats" (C9) is 89.13%, which is 7.46% higher than the highest accuracy of the other methods. This shows that the proposed algorithm has high recognition accuracy for types with few known samples.

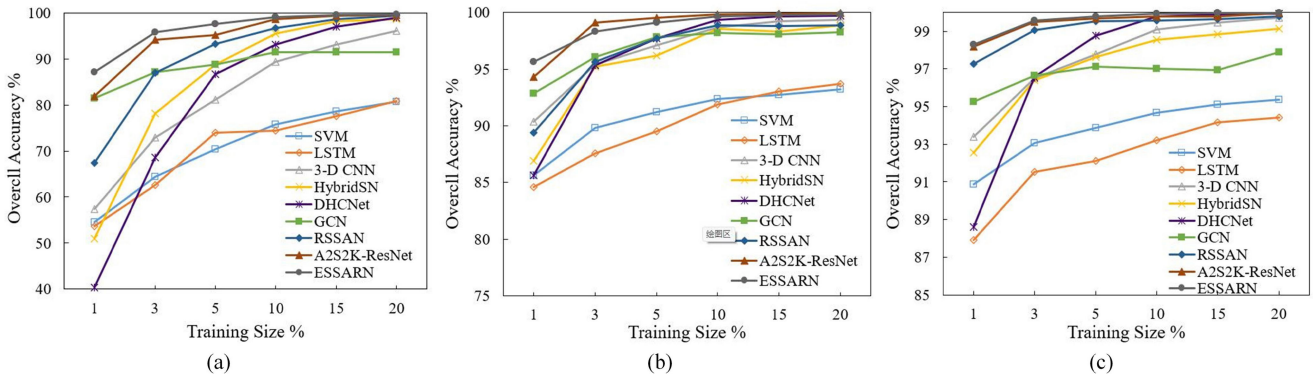

Fig. 7. OA achieved by different methods with varying training samples sizes. (a) Indian Pine. (b) Pavia University. (c) Salinas.

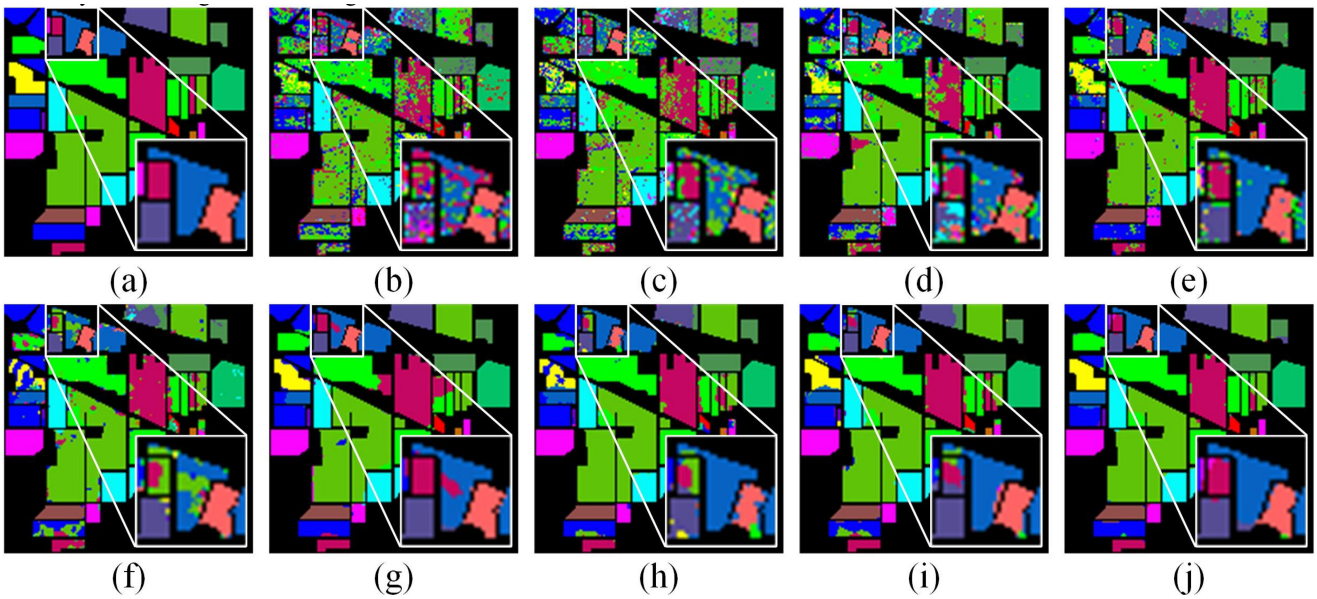

Fig. 8. Classification maps for IP dataset. (a) Ground truth. (b) Support vector machine. (c) Long-short term memory. (d) 3-D convolutional neural network. (e) HybridSN. (f) DHCNet. (g) Graph convolutional network. (h) RSSAN. (i) A2S2K-ResNet. (j) Enhanced spectral-Spatial residual attention network.

The PC algorithm increases the number of samples and contributes significantly to the improvement of classification accuracy. The SVM and LSTM provide worse results than the other methods, due to using only spectral information and missing the spatial relationship.

The experimental results for the PU dataset are shown in Fig. 9. Compared to the ground truth, it can be seen that the proposed algorithm handles the details better and classifies accurately. The other algorithms have poor results on this kind of data with scattered feature types, in particular “asphalt” (C1) and “self-blocking slices” (C8), which are often misclassified, as shown enlarged in the figure. Table V gives the obtained classification results for the PU dataset. From this table, we can see that ESSRAN obtains the highest classification accuracy with 95.87% for OA, 95.37% for AA, and 94.51% for kappa. The standard deviations of category accuracy show that categories with high accuracy generally have low standard deviations. The standard deviations of OA, AA, and kappa of the proposed method are the smallest among all methods, which are less than 0.7, while those of the other methods are larger than 1. This indicates that the proposed method has high stability and can accurately identify the target feature types. A total of 2/3 of the categories have accuracies higher than 97%, and 3 categories obtain the highest category accuracy. The category with the most significant accuracy improvement is ‘’bitumen” (C7), which improved by 7.95% over RSSAN. The high classification accuracy obtained with only 1% of the training samples shows that the proposed ESSRAN has a strong learning capability when the number of samples is small.

Fig. 10 shows the classification results of SA dataset. It can be seen that the misclassified feature types are mainly “vinyard_untrained” (C15) and “grapes_untrained” (C8). The classification maps of SVM and LSTM have obvious dot noise for the worst results, and the results of 3-D CNN and two improved CNN-based methods, HybridSN and DHCNet, also have many mismarks. Due to the use of a unique graph structure, GCN achieves visually smooth results, but in reality there are many misclassifications, such as “brocoli_green_weeds_2” (C2) being misclassified as “brocoli_green_weeds_1” (C1). The classification results of RSSAN and A2S2K-ResNet, which use an SSA mechanism, are better than the previous methods. In particular A2S2K-ResNet, which

TABLE IV
CLASSIFICATION RESULTS ON THE IP DATASET (%)

Class	SVM	LSTM	3D CNN	HybridSN	DHCNet	GCN	RSSAN	A2S2K-ResNet	ESSRAN
1	64.36±23.62	48.02±13.06	69.50±11.44	81.92±5.54	68.18±25.69	98.37±3.65	95.07±6.26	98.17±2.86	**100.00±0.00**
2	57.50±1.89	74.03±5.80	76.10±4.40	84.12±1.08	86.52±4.25	84.77±10.47	96.55±1.18	97.33±0.86	**98.28±0.85**
3	57.84±4.79	62.22±6.68	79.16±10.31	82.34±5.05	80.06±6.12	87.54±8.32	89.47±6.40	95.40±0.74	**96.87±1.90**
4	42.01±6.76	56.65±6.07	64.36±11.32	83.09±8.48	76.98±16.97	93.79±10.67	86.00±9.28	**97.67±3.42**	97.39±3.00
5	85.85±6.16	85.43±5.64	95.91±2.53	96.13±2.40	94.51±3.94	91.95±5.46	**97.87±1.77**	97.14±3.09	97.39±4.77
6	87.12±2.01	88.36±2.88	94.62±5.47	94.75±4.53	95.10±2.75	97.55±3.60	96.81±3.48	97.90±0.93	**99.34±0.80**
7	60.71±15.94	57.93±16.34	55.11±19.75	91.87±9.75	72.50±16.24	70.81±38.48	87.98±12.22	96.68±3.95	**97.98±3.00**
8	95.80±1.67	91.46±2.28	88.73±9.24	96.48±2.14	94.93±4.45	**99.96±0.09**	95.88±2.51	99.30±0.90	**99.96±0.09**
9	33.34±10.63	37.79±8.17	61.34±22.97	74.82±11.51	60.95±35.16	62.08±7.76	81.67±21.86	72.22±7.42	**89.13±12.11**
10	58.57±3.33	60.18±5.59	80.86±6.17	87.51±1.93	83.93±3.87	81.9±10.51	92.45±5.70	91.06±3.27	**95.60±1.73**
11	68.74±1.91	71.71±4.53	77.89±2.84	87.24±4.93	89.14±3.28	95.35±1.90	91.26±3.00	94.40±1.92	**98.18±0.89**
12	60.72±5.94	60.62±7.53	86.29±5.84	89.49±3.27	82.28±17.46	**95.80±3.94**	90.96±4.13	93.25±2.51	95.13±1.62
13	86.81±4.92	86.23±6.56	95.31±6.43	98.72±0.98	78.93±24.86	98.81±2.67	97.74±2.72	97.15±1.97	**99.07±2.09**
14	91.29±0.87	90.88±1.37	91.13±4.68	94.37±3.92	92.74±2.61	98.74±0.53	94.27±1.08	94.46±1.81	**98.75±0.29**
15	54.07±11.47	73.31±6.13	71.61±17.59	88.75±8.05	84.81±12.60	91.34±9.08	95.81±2.28	**96.49±3.17**	95.42±2.44
16	96.48±2.18	87.66±6.15	63.66±14.21	85.98±6.08	74.56±15.54	89.16±6.71	**98.46±3.08**	87.50±6.13	96.51±5.01
OA	70.37±0.86	73.99±1.06	81.16±0.48	88.75±2.07	86.65±2.47	91.27±2.16	93.24±0.71	95.12±0.53	**97.69±0.13**
AA	68.83±1.66	70.78±1.62	78.22±1.53	88.60±2.41	82.26±4.90	89.87±3.87	93.02±2.82	94.13±1.03	**97.19±0.54**
kappa	66.05±0.97	70.33±1.26	78.41±0.58	87.13±2.38	84.77±2.79	90.06±2.45	92.27±0.77	94.43±0.60	**97.37±0.14**

Boldface indicates the highest accuracy of all methods.

TABLE V
CLASSIFICATION RESULTS ON THE PU DATASET (%)

Class	SVM	LSTM	3D CNN	HybridSN	DHCNet	GCN	RSSAN	A2S2K-ResNet	ESSRAN
1	83.05±2.92	86.62±1.93	90.59±2.85	79.21±5.80	79.92±4.94	84.28±5.47	78.84±2.97	91.88±3.98	**92.92±1.35**
2	91.08±1.42	90.09±1.53	94.05±1.66	90.50±2.18	90.34±2.82	98.14±0.42	94.94±1.27	96.90±1.05	**98.22±0.92**
3	65.28±0.84	60.67±10.71	75.55±6.59	71.46±5.70	72.44±5.79	**96.54±7.59**	86.62±19.21	88.57±6.26	88.87±3.81
4	92.81±2.70	89.12±3.97	95.17±3.43	98.60±1.12	96.44±4.65	94.63±1.57	**99.6±0.25**	98.01±2.18	98.42±1.43
5	**100.00±0.00**	96.52±3.81	91.63±6.84	98.07±1.63	93.04±7.18	95.83±3.75	99.47±0.53	94.05±12.80	99.87±0.14
6	81.91±3.71	76.25±4.75	93.45±3.37	86.92±6.25	88.09±4.51	**99.15±0.55**	96.25±0.75	96.78±0.74	98.94±0.51
7	59.79±6.98	71.82±5.01	84.38±7.25	87.39±4.74	71.88±16.03	79.04±26.92	90.67±7.21	88.18±12.63	**98.62±1.11**
8	72.29±1.29	75.50±7.59	79.17±2.51	80.38±8.72	69.88±4.65	79.25±5.98	69.1±7.03	**84.90±4.11**	84.69±2.62
9	**99.83±0.16**	98.06±2.84	79.85±16.10	97.01±2.14	94.40±3.02	93.74±11.57	99.31±0.61	98.69±1.23	97.76±2.18
OA	85.60±0.37	84.60±0.41	90.35±0.79	86.92±2.02	85.64±1.65	92.88±1.51	89.4±2.93	94.29±1.65	**95.87±0.34**
AA	82.89±0.89	82.74±0.86	87.09±2.31	87.73±1.92	84.05±1.44	91.18±3.12	90.53±2.44	93.11±3.45	**95.37±0.63**
kappa	80.75±0.58	79.47±0.46	87.10±1.07	82.24±2.83	80.59±2.36	90.48±2.03	85.75±4.35	92.39±2.20	**94.51±0.46**

Boldface indicates the highest accuracy of all methods.

uses an adaptive adjustment of the kernel size, achieves high accuracy. Compared with other methods, the ESSRAN generates the most accurate and smooth classification maps, especially at the boundary of two different classes. The highest classification rates obtained with ESSRAN are 98.34% for OA, 98.84% for AA, and 98.15% for kappa (see Table VI). Meanwhile, ESSRAN has a low standard deviation of accuracy, and the average standard deviation of the three metrics is 0.18%. Among the other methods, the maximum standard deviation is 2.42% for OA, 1.42% for AA, and 2.73% for kappa. The proposed method achieves a high level of category accuracy, with an accuracy of over 99% for 11 categories. The proposed method gains higher classification accuracy in 8 of the 16 classes, with the accuracy of three categories reaching 100%. The class with the highest accuracy improvement is "lettuce_romaine_7wk" (C14), with an accuracy of 99.34%, which is 2.99% higher than other methods.

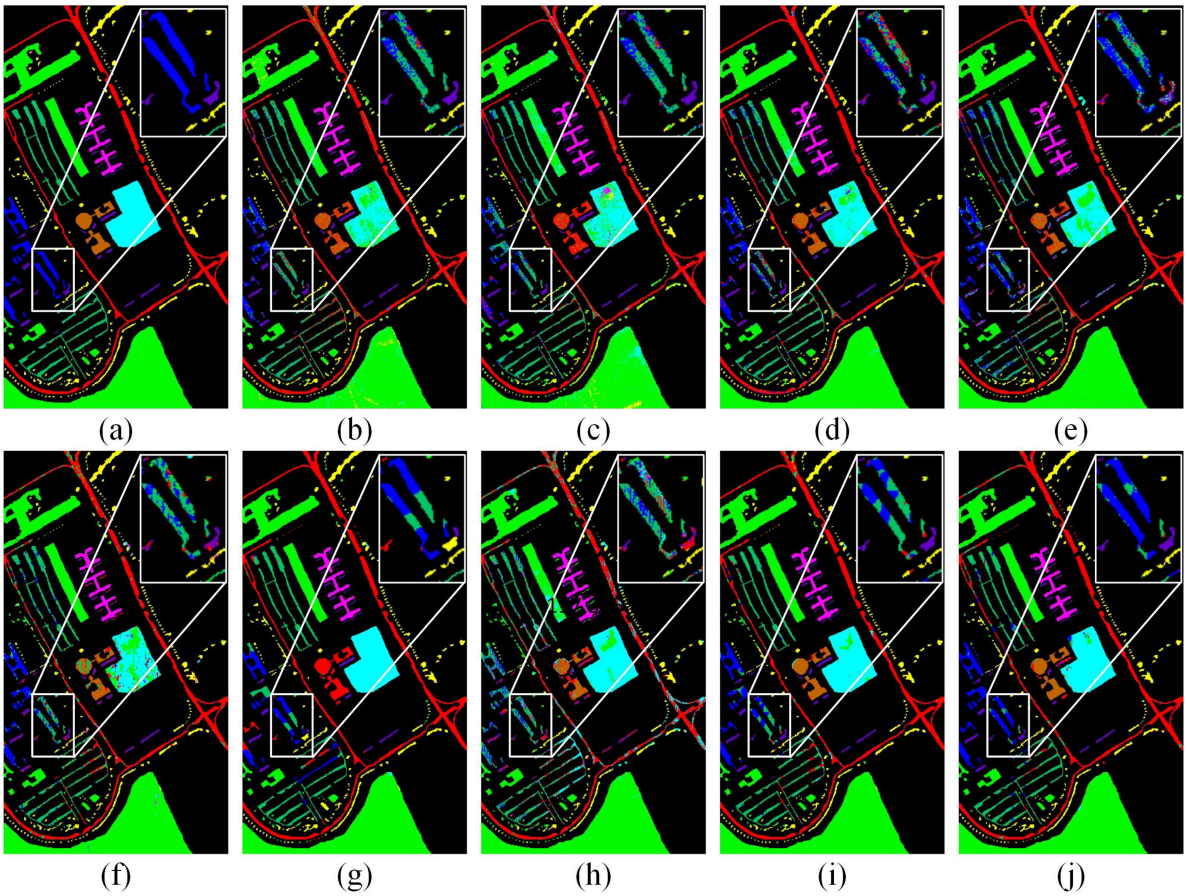

Fig. 9. Classification maps for PU dataset. (a) Ground truth. (b) Support vector machine. (c) Long-short term memory. (d) 3-D convolutional neural network. (e) HybridSN. (f) DHCNet. (g) Graph convolutional network. (h) RSSAN. (i) A2S2K-ResNet. (j) Enhanced spectral-spatial residual attention network.

E. Ablation Study

To further validate the effectiveness of different modules used in the proposed framework, we perform ablation experiments while keeping the other experimental settings unchanged. There are four modules of the proposed framework as follows.

1) The SSA and LSTM and PC are removed from the proposed framework, and FE and classification are performed using ResNet.
2) The spectral attention, spatial attention, and spectral-spatial attention are added to the model of ResNet exclusively and each network is denoted as SpeRAN, SpaRAN, and SSRAN, respectively.
3) The LSTM module is removed from the proposed framework (the resulting model is denoted as PC-SSRAN).
4) The PC module is removed from the proposed framework (the resulting model is denoted as LSTM-SSRAN).

Table VII gives the OA results of the ablation study. With the addition of SpeA and SpaA, the accuracy is improved compared to ResNet, and it is clear that SpeA has a greater effect on improving accuracy. After adding SSA, the accuracy of datasets IP, PU, and SA increased by 3.4%, 4.23%, and 2.16% compared to ResNet, due to the SSA module extracting diagnostic spectral and spatial information, which eliminates the effects of uncorrelated pixels and bands. Moreover, we compare the ESSRAN with SSARN, PC-SSARN (without LSTM) and LSTM-SSRAN (without PC). The results show that the inclusion of both PC and LSTM is important for OA enhancement, and the accuracy of the three datasets IP, PU, and SA is 0.75%, 0.49%, and 0.34% higher than that of SSARN, respectively. It can be concluded that the sample increase and the extraction of the relationship between adjacent bands are of great significance for the improvement of classification accuracy. For the IP and PU datasets, the addition of the spatial attention mechanism does not bring accuracy improvement to SSRAN, but for the SA dataset, the spatial attention mechanism is indispensable. This is related to the complexity and characteristics of the dataset itself.

F. Computational Cost

Table VIII gives the complexity of different methods in terms of training time, the number of trainable weight parameters updated during backpropagation, and computational cost. The results show that the proposed method takes more time to train than other methods due to the use of LSTM-based cell structure. However, the training time of the proposed method is less than the sum of the training time of LSTM and RSSAN, which indicates that the proposed method does not increase the time cost. Since the proposed algorithm does not have deep network layers, the number of parameters used for training is small (9.41×10^4)

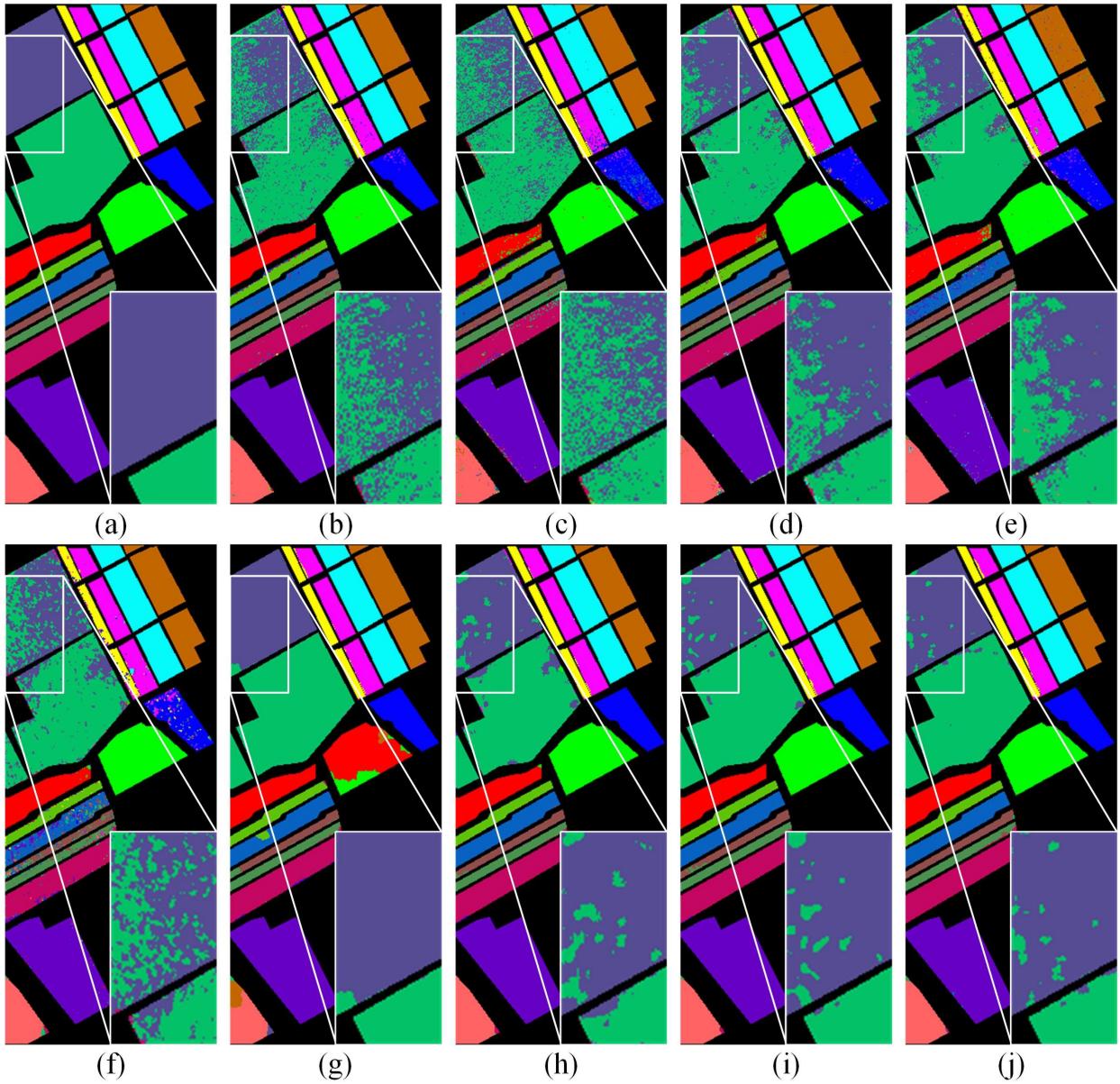

Fig. 10. Classification maps for SA dataset. (a) Ground truth. (b) Support vector machine. (c) Long-short term memory. (d) 3-D convolutional neural network. (e) HybridSN. (f) DHCNet. (g) Graph convolutional network. (h) RSSAN. (i) A2S2K-ResNet. (j) Enhanced spectral-spatial residual attention network.

and is only larger than that of GCN. The computational cost is calculated by floating point operations ($10^6 \times$FLOPs). The results show that the ESSRAN has much smaller FLOPs than the A2S2K-ResNet, which is 399.57×10^6 FLOPs.

G. Discussion

First, three hyperparameters (including learning rate, spatial size, and training size) that affect the experimental performance are tested in cross-validation experiments. The learning rate of the network is closely related to convergence and is set to 0.005 for the IP dataset and 0.001 for the PU and SA datasets. As the spatial size increases, the experimental accuracy increases first and then stabilizes. The best accuracy is achieved when the spatial size is 9×9, considering the input size of the proposed framework. Furthermore, the performance of all methods improves as the number of training samples increases. ESSRAN achieves excellent performance for all training sizes, and OA outperforms all comparison methods, demonstrating the absolute advantage of the ESSRAN method.

Second, the proposed ESSRAN model is compared with state-of-the-art deep learning-based methods by analyzing OA, AA, kappa, and category accuracy on three datasets. ESSRAN shows the best classification results on the IP, PU and SA datasets with 97.69%, 95.87% and 98.34% for OA, 97.19%, 95.37% and 98.84% for AA and 97.37%, 94.51% and 98.15% for kappa, respectively. Moreover, the proposed algorithm shows the most significant improvement in accuracy on the IP dataset. Compared with the A2S2K-ResNet algorithm, the OA of ESSRAN improves 2.57% on the IP dataset, 1.58% on the PU dataset, but only 0.16% on the SA dataset. The main advantages of the proposed algorithm are: richer extraction of spectral and spatial

TABLE VI
CLASSIFICATION RESULTS ON THE SA DATASET (%)

Class	SVM	LSTM	3D CNN	HybridSN	DHCNet	GCN	RSSAN	A2S2K-ResNet	ESSRAN
1	99.93±0.13	98.88±1.28	97.73±2.54	98.38±0.47	99.21±0.50	87.89±27.09	99.98±0.05	**100.00±0.00**	99.95±0.11
2	98.83±0.85	98.38±0.82	99.00±1.02	98.24±0.70	98.81±0.46	98.45±2.29	99.69±0.45	**100.00±0.00**	**100.00±0.00**
3	91.77±3.03	92.59±3.28	96.68±2.96	97.90±0.89	94.63±0.81	99.02±1.97	99.72±0.33	**99.98±0.03**	99.95±0.07
4	98.06±1.24	96.41±1.53	96.85±1.06	97.21±1.42	93.23±2.15	95.26±3.00	**98.34±1.10**	96.80±0.30	94.07±3.01
5	93.74±2.60	96.29±1.78	94.39±3.79	97.13±1.81	87.22±3.12	96.98±2.93	98.58±0.91	99.47±0.31	**99.56±0.32**
6	99.99±0.02	99.72±0.16	99.00±0.76	98.07±1.32	99.09±0.17	99.56±0.26	99.76±0.41	99.86±0.10	**100.00±0.00**
7	99.95±0.03	98.45±1.12	99.18±0.26	99.43±0.29	98.86±0.68	96.97±4.66	99.63±0.71	**99.98±0.05**	99.65±0.69
8	79.69±1.83	74.72±5.71	87.92±2.01	83.76±4.68	78.50±1.57	94.48±6.44	95.26±2.10	95.58±0.50	**97.27±1.12**
9	98.80±0.16	98.16±0.49	98.32±1.65	97.41±1.91	95.81±1.67	99.55±0.45	99.72±0.31	99.80±0.05	**99.94±0.05**
10	95.31±1.08	93.00±2.61	96.00±3.92	95.91±2.09	93.06±1.94	95.97±5.42	97.68±1.63	98.85±0.16	**98.89±0.87**
11	90.90±4.97	91.31±3.92	93.60±3.24	95.21±2.93	97.25±1.58	92.50±5.81	**100.00±0.00**	99.49±1.10	99.57±0.62
12	95.27±1.76	95.98±0.54	97.72±1.24	98.14±1.47	90.28±3.24	93.94±7.50	**99.96±0.08**	99.94±0.07	99.74±0.26
13	86.66±5.78	93.84±6.56	94.14±5.23	87.28±5.79	76.39±6.27	94.77±7.16	99.24±1.17	**99.93±0.11**	98.93±1.28
14	91.26±3.46	92.38±3.95	89.52±7.24	89.33±6.93	77.11±4.41	94.30±6.19	96.35±4.18	93.54±3.98	**99.34±0.74**
15	77.86±2.78	71.50±11.84	84.47±4.74	85.14±5.26	73.64±2.13	95.10±3.17	90.76±2.20	**96.05±1.20**	94.54±2.40
16	99.72±0.11	97.71±1.08	95.00±5.54	98.12±1.26	98.23±0.78	98.21±4.00	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**
OA	90.86±0.37	87.92±1.83	93.40±0.87	92.54±0.53	88.60±0.59	95.27±2.13	97.28±2.42	98.18±0.12	**98.34±0.19**
AA	93.61±0.82	93.08±0.73	94.97±0.65	94.79±0.54	90.71±0.42	95.81±1.42	98.42±1.11	98.70±0.22	**98.84±0.14**
kappa	89.80±0.41	86.53±1.98	92.65±0.96	91.68±0.59	87.29±0.65	94.73±2.38	96.97±2.73	97.97±0.13	**98.15±0.21**

Boldface indicates the highest accuracy of all methods.

TABLE VII
ACCURACY ANALYSIS TERMS OF OA FOR DIFFERENT MODULES OF THE PROPOSED FRAMEWORK

	ResNet	SpaRAN	SpeRAN	SSRAN	PC-SSRAN	LSTM-SSRAN	ESSRAN
IP	93.53±0.52	94.29±0.72	97.14±0.34	96.93±0.96	97.25±0.58	97.60±0.42	**97.69±0.13**
PU	91.15±1.44	91.19±1.19	95.57±0.34	95.38±0.53	95.21±0.12	95.65±0.58	**95.87±0.34**
SA	95.84±0.57	95.86±0.22	97.84±0.26	98.00±0.18	98.20±0.35	98.29±0.49	**98.34±0.19**

Boldface indicates the highest accuracy of all methods.

TABLE VIII
COMPUTATIONAL COST OF THE THREE DATASETS

		SVM	LSTM	3D CNN	HybridSN	DHCNet	GCN	RSSAN	A2S2K-ResNet	ESSRAN
Training Time (s)	IP	3.38	64.12	6.90	21.58	35.05	1.04	35.11	45.62	69.34
	PU	1.84	32.96	5.79	12.77	29.62	6.43	31.55	36.72	43.09
	SA	3.06	63.11	7.63	22.38	34.94	3.10	37.52	57.13	70.10
Parameters ($\times 10^4$)		-	169.10	109.06	353.71	32.71	2.92	11.06	37.08	9.41
Computation Cost ($10^6\times$FLOPs)		-	11.93	15.11	59.68	17.55	194.53	250.68	5454.31	399.57

information; better processing of edge information; and more accurate recognition of categories with small sample sizes.

Third, ablation experiments are used to demonstrate the role of individual structures in the model. We analyzed the impact of using or not using SSAN, LSTM, and PC in the model on the experimental results. The experimental results prove that all the added structures are beneficial to the accuracy. The combination of ResNet and LSTM allows it to better capture contextual information and retain the spectral and spatial features extracted by SSAN, which is extremely important for improving classification performance. In addition, PC has a good effect on the accuracy improvement of

small sample categories, which is more obvious on the IP dataset.

IV. Conclusion

In this article, we proposed an ESSRAN for HSIC. Initially the network uses a spectral-spatial attention mechanism to extract efficient and discriminative spectral and spatial information. Then, deep features are extracted using ResNet with the addition of LSTM, which obtains information about the relationships between adjacent spectra. The residual structure is able to combine the original features with the transformed features to obtain a stronger feature representation, which can further improve classification performance. Adequate experiments on three widely used HSI datasets demonstrate that the proposed ESSRAN model outperforms the state-of-the-art methods and achieves the highest classification accuracy. This network obtains extremely high classification accuracy with a simple structure, which fully demonstrates the advantages of the proposed method. In addition, experiments show that the ESSRAN algorithm has excellent classification results for data with uneven data distribution and a small number of samples, solving difficulties in obtaining labeled hyperspectral data. Considering that the proposed algorithm is a supervised learning classification method, in future research we will learn semisupervised and unsupervised approaches as well as more novel network models to make hyperspectral classification more intelligent accurate, and thus widely applied.

References

[1] S. K. Roy, S. Manna, T. Song, and L. Bruzzone, "Attention-based adaptive spectral–spatial kernel ResNet for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 9, pp. 7831–7843, Sep. 2021, doi: 10.1109/TGRS.2020.3043267.

[2] X. Zhang, S. Shang, X. Tang, J. Feng, and L. Jiao, "Spectral partitioning residual network with spatial attention mechanism for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5507714, doi: 10.1109/TGRS.2021.3074196.

[3] B. Zhang et al., "Progress and challenges in intelligent remote sensing satellite systems," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 1814–1822, 2022, doi: 10.1109/JSTARS.2022.3148139.

[4] X. Li, M. Ding, and A. Pižurica, "Spectral feature fusion networks with dual attention for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5508614, doi: 10.1109/TGRS.2021.3084922.

[5] Y. Tan, L. Lu, L. Bruzzone, R. Guan, Z. Chang, and C. Yang, "Hyperspectral band selection for lithologic discrimination and geological mapping," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 471–486, 2020, doi: 10.1109/JSTARS.2020.2964000.

[6] H. Liu, K. Wu, H. Xu, and Y. Xu, "Lithology classification using TASI thermal infrared hyperspectral data with convolutional neural networks," *Remote Sens.*, vol. 13, no. 16, Aug. 2021, Art. no. 3117, doi: 10.3390/rs13163117.

[7] R. Hänsch and O. Hellwich, "Fusion of multispectral LiDAR, hyperspectral, and RGB data for urban land cover classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 2, pp. 366–370, Feb. 2021, doi: 10.1109/LGRS.2020.2972955.

[8] X. Zhang, Y. Sun, K. Shang, L. Zhang, and S. Wang, "Crop classification based on feature band set construction and object-oriented approach using hyperspectral images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 9, pp. 4117–4128, Sep. 2016, doi: 10.1109/JSTARS.2016.2577339.

[9] S. Kendler, I. Ron, S. Cohen, R. Raich, Z. Mano, and B. Fishbain, "Detection and identification of sub-millimeter films of organic compounds on environmental surfaces using short-wave infrared hyperspectral imaging: Algorithm development using a synthetic set of targets," *IEEE Sensors J.*, vol. 19, no. 7, pp. 2657–2664, Apr. 2019, doi: 10.1109/JSEN.2018.2886269.

[10] X. Zheng, X. Chen, X. Lu, and B. Sun, "Unsupervised change detection by cross-resolution difference learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5606616, doi: 10.1109/TGRS.2021.3079907.

[11] H. Li, K. Wu, and Y. Xu, "An integrated change detection method based on spectral unmixing and the CNN for hyperspectral imagery," *Remote Sens.*, vol. 14, no. 11, May 2022, Art. no. 2523, doi: 10.3390/rs14112523.

[12] X. Sun et al., "Ensemble-based information retrieval with mass estimation for hyperspectral target detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5508123, doi: 10.1109/TGRS.2021.3075583.

[13] S. Zhong, C.-I. Chang, and Y. Zhang, "Iterative support vector machine for hyperspectral image classification," in *Proc. IEEE 25th Int. Conf. Image Process.*, 2018, pp. 3309–3312, doi: 10.1109/ICIP.2018.8451145.

[14] V. Jain and A. Phophalia, "Exponential weighted random forest for hyperspectral image classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 3297–3300, doi: 10.1109/IGARSS.2019.8897862.

[15] M. Wang, K. Gao, L. Wang, and X. Miu, "A novel hyperspectral classification method based on C5.0 decision tree of multiple combined classifiers," in *Proc. Fourth Int. Conf. Comput. Inf. Sci.*, 2012, pp. 373–376, doi: 10.1109/ICCIS.2012.33.

[16] F. Ratle, G. Camps-Valls, and J. Weston, "Semisupervised neural networks for efficient hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 5, pp. 2271–2282, May 2010, doi: 10.1109/TGRS.2009.2037898.

[17] M. Khodadadzadeh, P. Ghamisi, C. Contreras, and R. Gloaguen, "Subspace multinomial logistic regression ensemble for classification of hyperspectral images," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2018, pp. 5740–5743, doi: 10.1109/IGARSS.2018.8519404.

[18] Q. Liu, L. Xiao, J. Yang, and Z. Wei, "Multilevel superpixel structured graph U-Nets for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5516115, doi: 10.1109/TGRS.2021.3112586.

[19] Y. Duan, H. Huang, and T. Wang, "Semisupervised feature extraction of hyperspectral image using nonlinear geodesic sparse hypergraphs," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5515115, doi: 10.1109/TGRS.2021.3110855.

[20] F. Luo, Z. Zou, J. Liu, and Z. Lin, "Dimensionality reduction and classification of hyperspectral image via multistructure unified discriminative embedding," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5517916, doi: 10.1109/TGRS.2021.3128764.

[21] C. Chen, Y. Ma, and G. Ren, "Hyperspectral classification using deep belief networks based on conjugate gradient update and pixel-centric spectral block features," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 4060–4069, 2020, doi: 10.1109/JSTARS.2020.3008825.

[22] L. Liu, Y. Wang, J. Peng, L. Zhang, B. Zhang, and Y. Cao, "Latent relationship guided stacked sparse autoencoder for hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3711–3725, May 2020, doi: 10.1109/TGRS.2019.2961564.

[23] L. Mou, P. Ghamisi, and X. X. Zhu, "Deep recurrent neural networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3639–3655, Jul. 2017, doi: 10.1109/TGRS.2016.2636241.

[24] K. Makantasis, K. Karantzalos, A. Doulamis, and N. Doulamis, "Deep supervised learning for hyperspectral data classification through convolutional neural networks," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2015, pp. 4959–4962, doi: 10.1109/IGARSS.2015.7326945.

[25] L. Wang, Y. Lin, J. Liu, Z. Li, and C. Wu, "Siamese spectral attention with channel consistency for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 10226–10241, 2021, doi: 10.1109/JSTARS.2021.3115129.

[26] C. Yu, R. Han, M. Song, C. Liu, and C.-I. Chang, "A simplified 2D-3D CNN architecture for hyperspectral image classification based on Spatial–Spectral fusion," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 2485–2501, 2020, doi: 10.1109/JSTARS.2020.2983224.

[27] X. Zheng, T. Gong, X. Li, and X. Lu, "Generalized scene classification from small-scale datasets with multitask learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5609311, doi: 10.1109/TGRS.2021.3116147.

[28] Z. Dong, Y. Cai, Z. Cai, X. Liu, Z. Yang, and M. Zhuge, "Cooperative spectral–spatial attention dense network for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 5, pp. 866–870, May 2021, doi: 10.1109/LGRS.2020.2989437.

[29] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *Comput. Sci.*, 2014.

[30] Z. Cheng, F. Bai, Y. Xu, G. Zheng, S. Pu, and S. Zhou, "Focusing attention: Towards accurate text recognition in natural images," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5086–5094, doi: 10.1109/ICCV.2017.543.
[31] Y. Li, J. Zeng, S. Shan, and X. Chen, "Occlusion aware facial expression recognition using CNN with attention mechanism," *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2439–2450, May 2019, doi: 10.1109/TIP.2018.2886767.
[32] H. You, S. Tian, L. Yu, and Y. Lv, "Pixel-level remote sensing image recognition based on bidirectional word vectors," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 2, pp. 1281–1293, Feb. 2020, doi: 10.1109/TGRS.2019.2945591.
[33] C. Shan, J. Zhang, Y. Wang, and L. Xie, "Attention-based end-to-end speech recognition on voice search," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2018, pp. 4764–4768, doi: 10.1109/ICASSP.2018.8462492.
[34] X. Mei et al., "Spectral-spatial attention networks for hyperspectral image classification," *Remote Sens.*, vol. 11, no. 8, Apr. 2019, Art. no. 963, doi: 10.3390/rs11080963.
[35] X. Zheng, B. Wang, X. Du, and X. Lu, "Mutual attention inception network for remote sensing visual question answering," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5606514, doi: 10.1109/TGRS.2021.3079918.
[36] E. Pan et al., "Spectral-spatial classification of hyperspectral image based on a joint attention network," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 413–416, doi: 10.1109/IGARSS.2019.8898758.
[37] M. Zhu, L. Jiao, F. Liu, S. Yang, and J. Wang, "Residual spectral–spatial attention network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 449–462, Jan. 2021, doi: 10.1109/TGRS.2020.2994057.
[38] Z. Lu, B. Xu, L. Sun, T. Zhan, and S. Tang, "3-D channel and spatial attention based multiscale Spatial–Spectral residual network for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 4311–4324, 2020, doi: 10.1109/JSTARS.2020.3011992.
[39] D. Hong, L. Gao, J. Yao, B. Zhang, A. Plaza, and J. Chanussot, "Graph convolutional networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 5966–5978, Jul. 2021, doi: 10.1109/TGRS.2020.3015157.
[40] J.-Y. Yang, H.-C. Li, W.-S. Hu, L. Pan, and Q. Du, "Adaptive cross-attention-driven Spatial–Spectral graph convolutional network for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6004705, doi: 10.1109/LGRS.2021.3131615.
[41] Y. Dong, Q. Liu, B. Du, and L. Zhang, "Weighted feature fusion of convolutional neural network and graph attention network for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 31, pp. 1559–1572, 2022, doi: 10.1109/TIP.2022.3144017.
[42] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.
[43] Y. Jiang, Y. Li, and H. Zhang, "Hyperspectral image classification based on 3-D separable ResNet and transfer learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 12, pp. 1949–1953, Dec. 2019, doi: 10.1109/LGRS.2019.2913011.
[44] Z. Meng, L. Li, X. Tang, Z. Feng, L. Jiao, and M. Liang, "Multipath residual network for spectral-spatial hyperspectral image classification," *Remote Sens.*, vol. 11, no. 16, Aug. 2019, Art. no. 1896, doi: 10.3390/rs11161896.
[45] K. Li et al., "Depthwise separable ResNet in the MAP framework for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 5500305, doi: 10.1109/LGRS.2020.3033149.
[46] W. Hu, H. Li, L. Pan, W. Li, R. Tao, and Q. Du, "Spatial– spectral feature extraction via deep ConvLSTM neural networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 6, pp. 4237–4250, Jun. 2020, doi: 10.1109/TGRS.2019.2961947.
[47] F. Zhou, R. Hang, Q. Liu, and X. Yuan, "Hyperspectral image classification using spectral–spatial LSTMs," *Neurocomputing*, vol. 328, pp. 39–47, 2019.
[48] Q. Liu, F. Zhou, R. Hang, and X. Yuan, "Bidirectional-convolutional LSTM based spectral–spatial feature learning for hyperspectral image classification," *Remote Sens.*, vol. 9, no. 12, Dec. 2017, Art. no. 1330, doi: 10.3390/rs9121330.
[49] X. Tang et al., "Hyperspectral image classification based on spectral graph and bidirectional LSTM network," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2021, pp. 3661–3664, doi: 10.1109/IGARSS47720.2021.9553035.
[50] D. Hong et al., "More diverse means better: Multimodal deep learning meets remote-sensing imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4340–4354, May 2021, doi: 10.1109/TGRS.2020.3016820.
[51] K. Wu, G. Xu, Y. Zhang, and B. Du, "Hyperspectral image target detection via integrated background suppression with adaptive weight selection," *Neurocomputing*, vol. 315, pp. 59–67, 2018.
[52] H. Sun, X. Zheng, X. Lu, and S. Wu, "Spectral–spatial attention network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3232–3245, May 2020, doi: 10.1109/TGRS.2019.2951160.
[53] L. Mou and X. X. Zhu, "Learning to pay attention on spectral domain: A spectral attention module-based convolutional network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 1, pp. 110–122, Jan. 2020, doi: 10.1109/TGRS.2019.2933609.
[54] Z. Xie, J. Hu, X. Kang, P. Duan, and S. Li, "Multilayer global spectral–spatial attention network for wetland hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5518913, doi: 10.1109/TGRS.2021.3133454.
[55] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Adv. Neural Inf. Process. Syst.*, vol. 25, pp. 1097–1105, 2012.
[56] W.-S. Hu, H.-C. Li, T.-Y. Ma, Q. Du, A. Plaza, and W. J. Emery, "Hyperspectral image classification based on tensor-train convolutional long short-term memory," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2020, pp. 858–861, doi: 10.1109/IGARSS39084.2020.9324095.
[57] Y. Xu, L. Zhang, B. Du, and F. Zhang, "Spectral–spatial unified networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 10, pp. 5893–5909, Oct. 2018, doi: 10.1109/TGRS.2018.2827407.
[58] K. Wu, D. Zhao, Y. Zhong, and Q. Du, "Multi-probe based artificial DNA encoding and matching classifier for hyperspectral remote sensing imagery," *Remote Sens.*, vol. 8, no. 8, Aug. 2016, Art. no. 645, doi: 10.3390/rs8080645
[59] S. Dong, Y. Quan, W. Feng, G. Dauphin, L. Gao, and M. Xing, "A pixel cluster CNN and spectral-spatial fusion algorithm for hyperspectral image classification with small-size training samples," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 4101–4114, 2021, doi: 10.1109/JSTARS.2021.3068864.
[60] K. Wu, X. Feng, H. Xu, and Y. Zhang, "A novel endmember extraction method using sparse component analysis for hyperspectral remote sensing imagery," *IEEE Access*, vol. 6, pp. 75206–75215, 2018, doi: 10.1109/ACCESS.2018.2882187.
[61] 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, doi: 10.1109/TGRS.2016.2584107.
[62] S. K. Roy, G. Krishna, S. R. Dubey, and B. B. Chaudhuri, "HybridSN: Exploring 3-D–2-D CNN feature hierarchy for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 277–281, Feb. 2020, doi: 10.1109/LGRS.2019.2918719.
[63] J. Zhu, L. Fang, and P. Ghamisi, "Deformable convolutional neural networks for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 8, pp. 1254–1258, Aug. 2018, doi: 10.1109/LGRS.2018.2830403.
[64] 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, doi: 10.1109/TGRS.2020.3037361.

Yanting Zhan (Member, IEEE) received the B.S. degree in geoinformation science and technology in 2020 from the China University of Geosciences, Wuhan, China, where she is currently working toward the M.S. degree in resources and environment.

Her current research interests include hyperspectral image classification and deep learning.

Ke Wu received the B.S. and Ph.D. degrees in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2002 and 2008, respectively.

He is currently a Professor with the Institute of Geophysics and Geomatics, China University Geosciences. His current research interests include hyperspectral image processing, artificial neural network, and geological remote sensing.

Yanni Dong (Senior 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, Wuhan, China, in 2017.

She is currently an Associate Professor with the Institute of Geophysics and Geomatics, China University of Geosciences, Wuhan, China. 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 is a Reviewer of more than 20 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, *Journal of Public Relations Research*, and IEEE GEOSCIENCE AND REMOTE SENSING LETTERS. She regularly serves as PC member of *International Journal of Computing and Artificial Intelligence* and *Advancement of Artificial Intelligence*.