Received June 23, 2020, accepted July 5, 2020, date of publication July 8, 2020, date of current version July 22, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3008029

SSDANet: Spectral-Spatial Three-Dimensional Convolutional Neural Network for Hyperspectral Image Classification

XIN ZHANG[1,2], YONGCHENG WANG[1], NING ZHANG[1,2], DONGDONG XU[1], HUIYUAN LUO[1,2], BO CHEN[1], AND GUANGLI BEN[1,2]

[1]Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China
[2]College of Materials Science and Opto-Electronic Technology, University of Chinese Academy of Sciences, Beijing 100049, China

Corresponding author: Yongcheng Wang (wangyc@ciomp.ac.cn)

ABSTRACT Recently, the classification of hyperspectral images has made great process. Especially, the classification methods based on three-dimensional convolutional neural network have remarkable performance due to the uniqueness of hyperspectral images. However, the hyperspectral classification still faces great challenges due to a series of problems such as the insufficient extraction of spectral-spatial features, the lack of labeled samples, the large amount of noise, the tendency of overfitting and so on. Therefore, SSDANet is proposed to solve the above problems and promote the further development of hyperspectral classification technology based on deep learning. SSDANet is a spectral-spatial three-dimensional convolutional neural network with a deep and wide structure that can significantly improve classification performance. In SSDANet, the spectral-spatial dense connectivity is put forward to protect the integrity of information. It is made up of the spectral branch and the spatial branch, which can learn and reuse the spectral-spatial features. Besides, the spectral-spatial attention mechanism is proposed to adapt the special structure of hyperspectral images. It can excite important spectral-spatial information and suppress unimportant spectral-spatial information. In addition, a series of optimization methods including data augmentation, batch normalization, dropout, exponential decay learning rate, and L2 regularization are adopted to alleviate the problem of overfitting and improve the classification results. To verify the performance of SSDANet, experiments were implemented on two widely used datasets—Pavia University and Indian Pines. Under the condition of limited labeled samples, the classification evaluation indexes of OA, AA, and Kappa on the two datasets all exceeded 99%, reaching state-of-the-art performance.

INDEX TERMS Artificial intelligence, hyperspectral imaging, image processing, pattern recognition, remote sensing.

I. INTRODUCTION

Images obtained by the hyperspectral remote sensing sensor or imaging spectrometer are called hyperspectral images (HSIs), which contain hundreds of spectral channels from visible bands to infrared bands [1]. Compared with the traditional RGB images, HSIs have richer and more detailed spectral information, which is helpful for classification and recognition tasks [2]. Hyperspectral classification aims to classify each pixel in the image into a specific category [3], which has been widely used in civil and military applications, such as food analysis [4], mineral resource exploitation [5], agriculture development [6], anomaly detection [7], etc. However, it still faces many problems. The main challenges are listed as follows. (1) In the process of supervised learning, the imbalance between high-dimensional data and limited training samples can easily lead to the phenomenon that classification results decline with the increase of dimensions, which is called the curse of dimensionality [8]. (2) The high cost of manual labeling of HSIs leads to the shortage of label samples [9]. (3) The spatial layout of HSIs is complicated. What is worse, different materials have the same spectral characteristics, which further increases the difficulty of classification [10].

With the continuous development of machine learning technology, advanced methods emerge one after

The associate editor coordinating the review of this manuscript and approving it for publication was Qichun Zhang.

another [11], [12] and are widely applied in various fields, such as medical fitting prediction [13], battery capacity and aging prediction [14]–[16], natural language processing [17], image processing [18] and so on. And the classification methods of HSIs are also gradually developed, which can be roughly divided into two categories according to whether the use of high-level features—the traditional classification methods and the classification methods based on deep learning.

At the early stage, the traditional classification methods are based on spectral information, which generally includes two main elements: feature engineering and classifiers [19]. The function of feature engineering is to reduce the dimension of HSIs and obtain discriminative features or bands. Feature extraction and feature selection are two common methods in feature engineering [20]. And the purpose of feature extraction is to transform the hyperspectral data of high dimension space into low dimension space so that different categories can be easily distinguished [21]. Typical methods of feature extraction include independent component analysis (ICA) [22], linear discriminant analysis (LDA) [23], principal component analysis (PCA) [24], minimum noise fraction (MNF) [25] and so on. Whereas the function of feature selection is to retain the spectral information of the most representative bands from the raw HSIs and discard the bands that contribute less to the classification. Common methods of feature selection include Jeffries-Matusita distance [26], spectral angle mapper (SAM) [27], Bhattacharyya distance [28], etc. Features generated by feature engineering are used as the input of the classifier. Representative classifiers include k-nearest neighbor (KNN) [29], random forest (RF) [30], support vector machine (SVM) [31], etc. However, the traditional classification methods based on spectral information do not make full use of the spatial information of HSIs. Therefore, the traditional classification methods based on spectral-spatial information are proposed. Generally, these methods extract the spatial features by morphological profiles [32], super-pixel [33], multi-kernel learning [34], sparse representation [35] and so on, and then the spatial features are integrated with spectral features. Nevertheless, the traditional classification methods of HSIs, whether based on spectral features or spectral-spatial features, all rely on hand-crafted features with limited representation ability, which cannot fit the classification task well. What is worse, the traditional classification methods rely on the prior information of experienced experts, which leads to poor generalization ability of these methods for other scenarios.

Recently, the research of the hyperspectral classification methods based on deep learning has become a hotspot, because it can solve the problems existing in the traditional methods [36]. The deep learning model has a hierarchical structure, which can learn high-level semantic information from the data automatically. It can transform images into more recognizable features, thus making the classification task of HSIs effective and robust. Typical deep learning methods include deep belief network (DBN) [37], stacked auto-encoder (SAE) [38], convolutional neural network (CNN) [39] and so on, which have been widely used for classification of HSIs. The classification methods based on deep learning can also be divided into the classification methods based on spectral information and the classification methods based on spectral-spatial information. And the classification methods based on deep learning using spectral information only extract spectral features of HSIs, which generally include the methods based on DBN [40], the methods based on SAE [41], [42], and the methods based on 1-D CNN [43], [44]. These methods perform better than traditional classification methods, but the input samples need to be flattened into a one-dimension vector, resulting in the spatial information of HSIs can not be fully extracted. Fortunately, the classification methods based on deep learning using spectral-spatial features can yield better results than the methods using spectral features alone. And they contain two types of implementation: (1) spectral features and spatial features are extracted respectively, after that the features are fused to carry out classification [45], [46]; (2) the spectral-spatial features are extracted by 3-D CNN directly [2]. Since HSIs are the form of 3-D cube, 3-D CNN can make full use of the structural characteristics of HSIs by performing 3-D convolution operation on the data, so as to achieve satisfactory classification results [47].

Nowadays, different 2-D CNN models have been proposed, such as LeNet [48], AlexNet [49], GoogleNet [50], VGGNet [51], ResNet [52], DenseNet [53], SENet [54], CliqueNet [55] and so on. On the basis of these models, various classification methods based on 3-D CNN of HSIs have also been proposed. Zhong *et al.* proposed the methods based on 3-D residual connections for the classification of HSIs [56]. Zhang *et al*. introduced the 3-D densely connected convolutional network to extract spectral-spatial feature of HSIs [9]. Wang *et al.* proposed a deep and fast 3-D CNN framework based on dense connectivity, and obtained satisfactory results [57]. Zhang *et al.* put forward a multi-scale network that used the 3-D dense connection structure to aggregate features at different levels, so as to improve classification performance [36]. Paoletti *et al.* proposed a deep and dense 3-D CNN to make full use of the HSIs' information [58]. Fang *et al.* introduced a network with 3-D dense connectivity and spectral-wise attention mechanism that yielded competitive performance [59].

However, the existing methods based on 3-D dense connectivity for the classification of HSIs are all aimed at using the dense connectivity to make the model deeper, but ignore the width of the network, which will lead to the gradual loss of detailed features as the model deepens. In addition, the classification methods based on attention mechanism so far only use spectral features [59] or simply combine spatial features with spectral features [60], ignoring the special structure of HSIs. Besides, 3-D CNN is prone to overfitting due to the numerous parameters, thus reducing the classification performance of HSIs. In view of these problems, a deep and wide 3-D CNN model—SSDANet is proposed. In SSDANet,

spectral-spatial dense connectivity is put forward to make full use of the spectral-spatial features and strengthen the transmission of information flow, so as to reduce the loss of information. Moreover, the spectral-spatial attention mechanism which can adapt to the 3-D structure of HSIs is proposed. It can recalibrate the spectral and spatial features of HSIs, so that important information can be selected and less important features can be suppressed. In order to alleviate the problem of overfitting and improve the classification results, a series of algorithms are used for optimization. To demonstrate the performance of the proposed method, the SSDANet was trained on two benchmark datasets of HSIs and satisfactory results were obtained. The SSDANet has five advantages over other approaches:

1) It is an end-to-end 3-D CNN model that can learn spatial and spectral information at the same time without any pre-processing or post-processing operations.
2) It increases the width of the model by cascading the spectral branch with the spatial branch. Compared with other 3-D CNN methods based on dense connectivity, it not only deeper, but also wider, so that more discriminative features can be obtained and the integrity of information can be protected.
3) It uses the spectral-spatial attention mechanism, which can redistribute the weights of the spectral-spatial feature maps, so as to capture important information.
4) It uses many algorithms for optimization, including data augmentation, dropout, batch normalization, exponential decay learning rate, and L2 regularization, so as to make the network more robust and generalized.
5) It has achieved satisfactory classification results on two widely used datasets, reaching state-of-the-art level.

The remainder of this article is arranged as follows. The basics of convolutional neural networks for hyperspectral classification is introduced in Section II. And the details of the proposed method are described in Section III. In Section IV, the datasets, experiment setup, and experiment results are described. Furthermore, the analysis and discussion are presented in Section V. In Section VI, conclusions are presented.

II. BASICS OF CONVOLUTIONAL NEURAL NETWORKS FOR HYPERSPECTRAL CLASSIFICATION

There are three types of convolutional neural networks for the classification of HSI—1-D CNN, 2-D CNN, and 3-D CNN. The traditional CNN is 2-D CNN that processes RGB images with two-dimensional structure. Generally, a 2-D CNN mainly includes the convolutional layer, the pooling layer, and the fully connected layer. Since other articles have introduced the specific structure of 2-D CNN [61], it will not be covered here. Nevertheless, as the most important difference among 1-D CNN, 2-D CNN, and 3-D CNN, the convolutional layer is described in detail.

The convolutional layer of 1-D CNN uses one-dimensional convolutional kernels to operate on the one-dimensional input. The calculation equation of $v_{l,j}^{x}$ which represents the neuron at position x on the jth feature map in the lth layer is as follows:

$$v_{l,j}^{x} = f\left(\sum_{m}\sum_{h=0}^{H_l-1} k_{l,j,m}^{h} v_{(l-1),m}^{(x+h)} + b_{l,j}\right), \tag{1}$$

where, m refers to the index of the feature map in the $(l-1)$th layer. And H_l represents the length of the one-dimensional convolutional kernel. In addition, the weight of the jth convolutional kernel at position h on the mth feature map in the lth layer is represented by $k_{l,j,m}^{h}$. And the value of neuron at the position $(x+h)$ on the mth feature map in the $(l-1)$th layer is represented by $v_{(l-1),m}^{(x+h)}$. Moreover, $b_{l,j}$ refers to the bias on the jth convolutional kernel in the lth layer. And $f(\cdot)$ represents the activation function, which is universal in 1-D CNN, 2-D CNN, and 3-D CNN.

The convolutional layer of 2-D CNN uses the two-dimensional convolutional kernels to operate on the two-dimensional input. The value of the neuron $v_{l,j}^{x,y}$ at position (x, y) on the jth feature map in the lth layer can be calculated by:

$$v_{l,j}^{x,y} = f\left(\sum_{m}\sum_{h=0}^{H_l-1}\sum_{w=0}^{W_l-1} k_{l,j,m}^{h,w} v_{(l-1),m}^{(x+h),(y+w)} + b_{l,j}\right), \tag{2}$$

where, m refers to the index of the feature map in the $(l-1)$th layer. Furthermore, the height and the width of the convolutional kernel are represented by H_l and W_l respectively. Besides, the weight of the jth convolutional kernel at position (h, w) on the mth feature map in the lth layer is represented by $k_{l,j,m}^{h,w}$. Additionally, the value of neuron at the position $(x+h, y+w)$ on the mth feature map in the $(l-1)$th layer is represented by $v_{(l-1),m}^{(x+h),(y+w)}$. And $b_{l,j}$ is the bias.

Similarly, the convolutional layer of 3-D CNN uses the three-dimensional convolutional kernels to operate on the three-dimensional input. The calculation equation of $v_{l,j}^{x,y,z}$ which represents the neuron at position (x, y, z) of the jth feature map in the lth layer can be expressed by:

$$v_{l,j}^{x,y,z} = f\left(\sum_{m}\sum_{h=0}^{H_l-1}\sum_{w=0}^{W_l-1}\sum_{r=0}^{R_l-1} k_{l,j,m}^{h,w,r} v_{(l-1),m}^{(x+h),(y+w),(z+r)} + b_{l,j}\right), \tag{3}$$

where, the index of the feature map in the $(l-1)$th layer is represented by m. Besides, the height, the width, and spectral dimension of the convolutional kernel are represented by H_l, W_l, and R_l respectively. Additionally, $k_{l,j,m}^{h,w,r}$ represents the weight of the jth convolutional kernel at position (h, w, r) on the mth feature map in the lth layer. Furthermore, the value of neuron at the position $(x+h, y+w, z+r)$ on the mth feature map in the $(l-1)$th layer is represented by $v_{(l-1),m}^{(x+h),(y+w),(z+r)}$. And $b_{l,j}$ is the bias.

It can be seen from the different expressions of the convolution layer in 1-D CNN, 2-D CNN, and 3-D CNN that the dimensional type of CNN is closely related to the form of

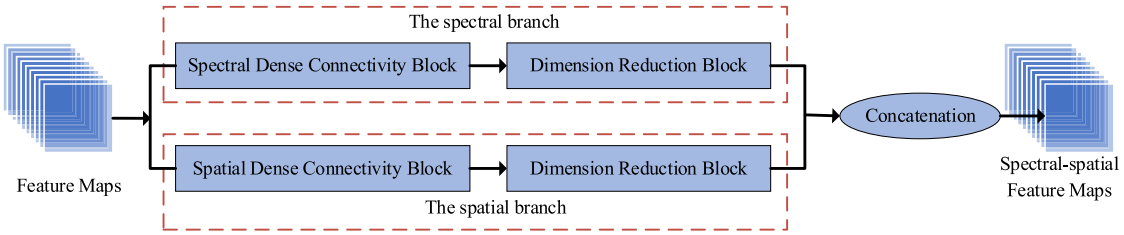

FIGURE 1. The framework of the spectral-spatial dense connectivity.

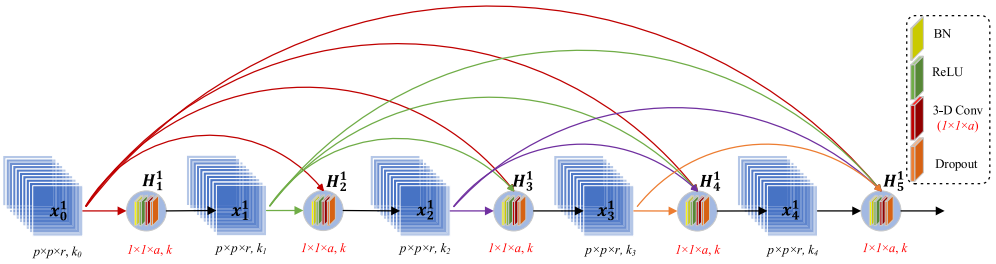

FIGURE 2. The schematic diagram of the spectral dense connectivity block.

input data. Therefore, 3-D CNN is more consistent with the 3-D structure of HSIs than 1-D CNN and 2-D CNN.

III. PROPOSED METHOD

SSDANet for hyperspectral classification is introduced in this section. SSDANet consists of three important components: spectral-spatial dense connectivity that can learn spectral and spatial features simultaneously, spectral-spatial attention mechanism that can squeeze and excite the spectral-spatial features, and optimization methods that can improve the classification performance. These three parts are elaborated in detail. What is more, the whole framework of SSDANet is summarized in a graphical flowchart for further understanding.

A. SPECTRAL-SPATIAL DENSE CONNECTIVITY

The spectral-spatial dense connectivity is as shown in Fig. 1. It is made up of two branches—the spectral branch and the spatial branch. In the spectral branch, the spectral features with redundant information are obtained by using the spectral dense connectivity block, and then the dimension reduction block is used to reduce the computation. Similar to the spectral branch, the spatial branch contains the spatial dense connectivity block and the dimension reduction block to acquire spatial information. In the end, the spectral-spatial information is obtained by combining the features extracted from the two branches. The spectral-spatial dense connectivity can not only make the network wider than other methods, but also can extract discriminant spectral-spatial features. Details of the spectral dense connectivity block, spatial dense connectivity block, and dimension reduction block are elaborated below.

1) SPECTRAL DENSE CONNECTIVITY BLOCK

The dense connectivity [53] used for the classification of traditional RGB images can enable the reuse of features, strengthen the transmission of information flow, and alleviate the problem of gradient disappearance. Based on the dense connectivity, the spectral dense connectivity block is proposed. Similar to the dense connectivity, the input of the current layer in the spectral dense connectivity block is the concatenation of all the previous layers' outputs; the main difference is the traditional dense connectivity adopts 2-D CNN to extract features, whereas the spectral dense connectivity block is more suitable for the structural characteristics of HSIs by using 3-D CNN to extract spectral features. As shown in Fig. 2, the input of the spectral dense connectivity block x_0^1 is k_0 feature maps with the size of $p \times p \times r$. Where, the superscript *1* of x_0^1 means that the feature maps belong to the spectral dense connectivity block, and the subscript *0* represents the position of the feature maps. Assuming that the spectral dense connectivity block contains $l(l \in N^*)$ layers, and each layer implements the nonlinear transformation $H_l^1(\cdot)$. Where, the superscript *1* of $H_l^1(\cdot)$ means that the nonlinear transformation belongs to the spectral dense connectivity block, and the subscript l refers to the index of the layer. More specifically, $H_l^1(\cdot)$ is the composite function of batch normalization (BN) [62], ReLU, 3-D convolution, and dropout [49]. In the operation of 3-D convolution, k convolutional kernels with the size of $1 \times 1 \times a$ are used. Where, $\{a \mid a > 1, a \in N^*\}$. Moreover, the convolutional operation adopts the "SAME" mode, so the size of the feature maps remains unchanged during the forward propagation. The output of the spectral dense connectivity block x_l^1 can be calculated by:

$$x_l^1 = H_l^1([x_0^1, x_1^1, \cdots, x_{l-1}^1]), \tag{4}$$

where, $[x_0^1, x_1^1, \cdots, x_{l-1}^1]$ represents the concatenation of the feature maps. And the output number of feature maps k_l can be expressed as:

$$k_l = k \times (l-1) + k_0, \tag{5}$$

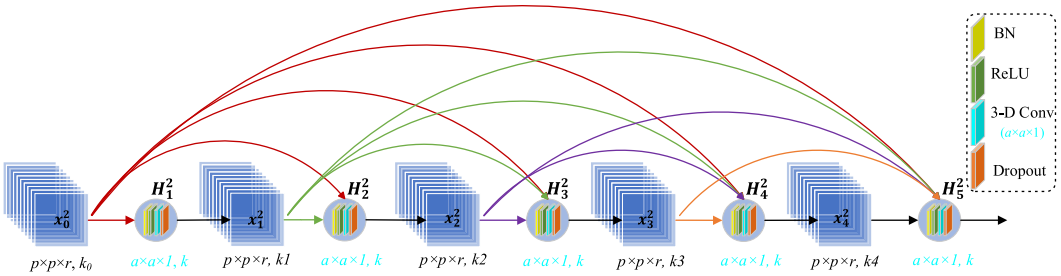

FIGURE 3. The schematic diagram of the spatial dense connectivity block.

2) SPATIAL DENSE CONNECTIVITY BLOCK

The basic principle of the spatial dense connectivity block is similar to that of the spectral dense connectivity block, except that the kernel size used for 3-D convolution is different. As shown in Fig. 3, the input of the spatial dense connectivity block is also k_0 feature maps with the size of $p \times p \times r$, just like the spectral dense connectivity block. If the number of nonlinear transformation layers in the spatial dense block is l, and the nonlinear transformation is represented by $H_l^2(\cdot)$. Where, the superscript *2* of $H_l^2(\cdot)$ means that it belongs to the spatial dense connectivity block, and the subscript l is the index of the layer. And the nonlinear transformation in the spatial dense connectivity block is also composed of BN, ReLU, 3-D convolution, and dropout. The 3-D convolution of the spatial dense connectivity block uses k convolutional kernels with the size of $a \times a \times 1$, which is the main difference from the spectral dense connectivity block. The output equation of the spatial dense connectivity block x_l^2 can be expressed by:

$$x_l^2 = H_l^2([x_0^2, x_1^2, \cdots, x_{l-1}^2]), \tag{6}$$

And the output number of feature maps in the spatial dense connectivity block is the same as that of spectral dense connectivity block.

3) DIMENSION REDUCTION BLOCK

Since the features extracted by the spectral dense connectivity block and the spatial dense connectivity block contain redundant information, the dimension reduction block is used to reduce the computation, accelerate the training process, and reduce the overfitting problem. Fig. 4 is the schematic diagram of the dimension reduction block. The input of the dimension reduction block is the output of the spectral connectivity block or the spatial connectivity block—k_l feature maps with the size of $p \times p \times r$. The dimension reduction block is composed of BN, ReLU, 3-D convolution, dropout, and 3-D average pooling. Among them, BN, ReLU, and dropout are adopted to enhance the nonlinear discrimination ability, improve the training speed, and avoid overfitting. Particularly, the 3-D convolution and the 3-D average pooling are indispensable parts of dimension reduction block. There are m convolutional kernels with the size of $1 \times 1 \times 1$ in the 3-D convolution. Where, $m < k_l$. Through the 3-D convolution, the number of feature maps reduces from k_l to m. Additionally, the 3-D average pooling using the filter of $b \times b \times b$ with the stride of (c, c, c) is adopted to reduce the size of the feature maps. Where, $\{b | b \leq p \leq r\}$. After operation of average pooling, m feature maps with the size of $\left\lfloor \frac{p-b}{c} + 1 \right\rfloor \times \left\lfloor \frac{p-b}{c} + 1 \right\rfloor \times \left\lfloor \frac{r-b}{c} + 1 \right\rfloor$ are obtained. Where, $\lfloor \cdot \rfloor$ represents the operation of rounding down.

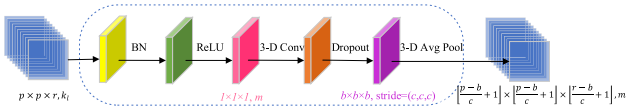

FIGURE 4. The schematic diagram of the dimension reduction block.

B. SPECTRAL-SPATIAL ATTENTION MECHANISM

Different from the attention mechanism [54] used for the processing of traditional RGB images, the spectral-spatial attention mechanism can learn the importance of each channel from HSIs automatically, so as to promote useful spectral-spatial features and suppress useless spectral-spatial information. The spectral-spatial attention mechanism is composed of two important elements: spectral-spatial squeeze and spectral-spatial excitation. To illustrate the details of the proposed method, Fig. 5 shows the schematic diagram of the spectral-spatial attention mechanism. The input of the spectral-spatial attention mechanism X is a set of C feature maps with the size of $P \times Q \times L$. Where, $X = [x_1, x_2, \cdots, x_C]$. First, the input data are squeezed to embed the global information of HSIs. Different from the traditional attention mechanism which uses 2-D average pooling to squeeze each channel, the proposed spectral-spatial attention

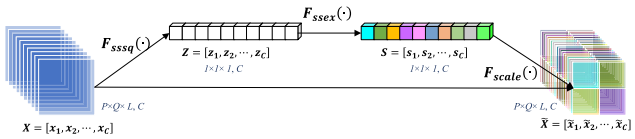

FIGURE 5. The schematic diagram of the spectral-spatial attention mechanism.

mechanism uses 3-D average pooling to squeeze the global spectral-spatial features into channel-wise statistics Z with the size of $1 \times 1 \times 1 \times C$. Where, $Z = [z_1, z_2, \cdots, z_C]$. And z_C squeezed by x_C can be calculated by:

$$z_C = F_{sssq}(x_C) = \frac{1}{P \times Q \times L} \sum_{i=1}^{P} \sum_{j=1}^{Q} \sum_{k=1}^{L} x_C(i, j, k), \quad (7)$$

where, $F_{sssq}(\cdot)$ represents the operation of squeeze. Then, the channel statistics Z is mapped into a set of channel weights S through the operation of spectral-spatial excitation. The main purpose of the spectral-spatial excitation is to recalibrate the information aggregated from the operation of spectral-spatial squeeze adaptively. It adopts a simple self-gating mechanism by using two fully connected layers with different activation functions to calculate the weights of each channel, so as to fully obtain channel-wise dependencies of HSIs.

Where, $S = [s_1, s_2, \cdots, s_C]$, which can be expressed by:

$$S = F_{ssex}(Z, W) = \sigma(W_2 \delta(W_1 Z)), \quad (8)$$

where, $\delta(\cdot)$ represents the activation function—ReLU, $\sigma(\cdot)$ represents the activation function—sigmoid. And $F_{ssex}(\cdot)$ represents the operation of spectral-spatial excitation. $W_1 \in {}^{\frac{C}{r} \times C}$ and $W_2 \in {}^{C \times \frac{C}{r}}$ represent the weight matrices of the two fully connected layer respectively. Where, r is the reduction ratio. Finally, the set of channel weights S is multiplied by the input X to obtain the final output $\tilde{X}$. Where, $\tilde{X} = [\tilde{x}_1, \tilde{x}_2, \cdots, \tilde{x}_C]$, and it can be expressed as follows:

$$\tilde{X} = F_{scale}(S, X) = [s_1 x_1, s_2 x_2, \cdots, s_C x_C], \quad (9)$$

where, $F_{scale}(\cdot)$ represents the channel-wise multiplication.

C. OPTIMIZATION METHODS

In the field of the hyperspectral classification, the large amount of noise in the HSIs, the limited number of labeled samples, the complex structure of the model, and the numerous parameters of 3-D CNN all lead to the phenomenon of overfitting. To prevent overfitting and improve the accuracy, a series of optimization methods including data augmentation, batch normalization, dropout, exponential decay learning rate, and L2 regularization are adopted.

1) DATA AUGMENTATION

In view of the small number of labeled samples in HSIs, the strategy of data augmentation is proposed to improve the robustness and alleviate the overfitting of the constructed model, which is shown in Fig. 6. In this paper, the data cube with the size of $n_H \times n_W \times n_B$ is taken as a sample. Where, n_H, n_W and n_B represent the height, width, and the spectral dimension of the data cube respectively. In the strategy of data augmentation, the original samples are expanded to 5 times by means of flipping along the up-down direction, flipping along the left-right direction, rotating at a random angle, and adding random Gaussian noise in the training process.

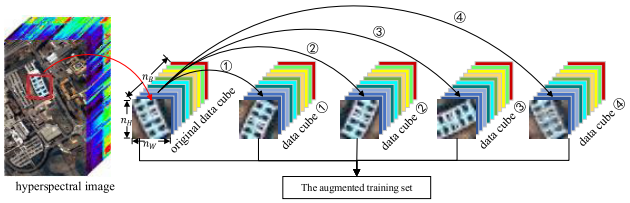

FIGURE 6. The strategy of data augmentation. Where, ① represents the operation of flipping along the up-down direction; ② represents the operation of flipping along the left-right direction; ③ represents the operation of rotating at a random angle; ④ represents the operation of adding random Gaussian noise.

2) BATCH NORMALIZATION

To alleviate the problem of overfitting and accelerate the convergence of the network, the optimization method of BN is used in the paper. Suppose the input of BN is $X = [x_1, x_2, \cdots, x_n]$. Where, x_n represents one of the samples, and n represents the batch size. The mean μ_B and variance σ_B^2 of the input data can be calculated by (10) and (11), respectively:

$$\mu_B = \frac{1}{n} \sum_{i=1}^{n} x_i, \quad (10)$$

$$\sigma_B^2 = \frac{1}{n} \sum_{i=1}^{n} (x_i - \mu_B)^2, \quad (11)$$

Next, each element of the input is normalized, as show in (12). Where, ε represents a constant.

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}}, \quad (12)$$

Finally, the final output y_i is obtained through scaling and shifting, as shown in the following equation:

$$y_i = \gamma \hat{x}_i + \beta, \quad (13)$$

3) DROPOUT

Dropout is adopted to alleviate the problem of overfitting. The basic principle of dropout is that the weights of some neurons in the hidden layer stop updating according to a certain probability in the training process, so as to ease the complex co-adaptation relationship between neurons. And Fig. 7 is used to further illustrate the difference between the networks with and without dropout.

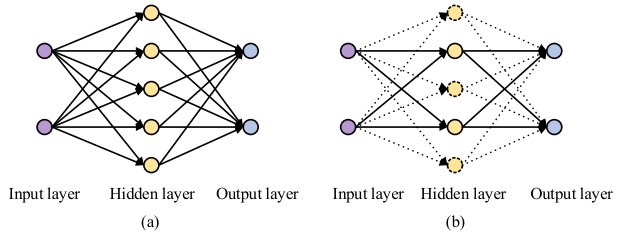

FIGURE 7. The neural networks with and without dropout. (a) The neural network without dropout; (b) The neural network with dropout.

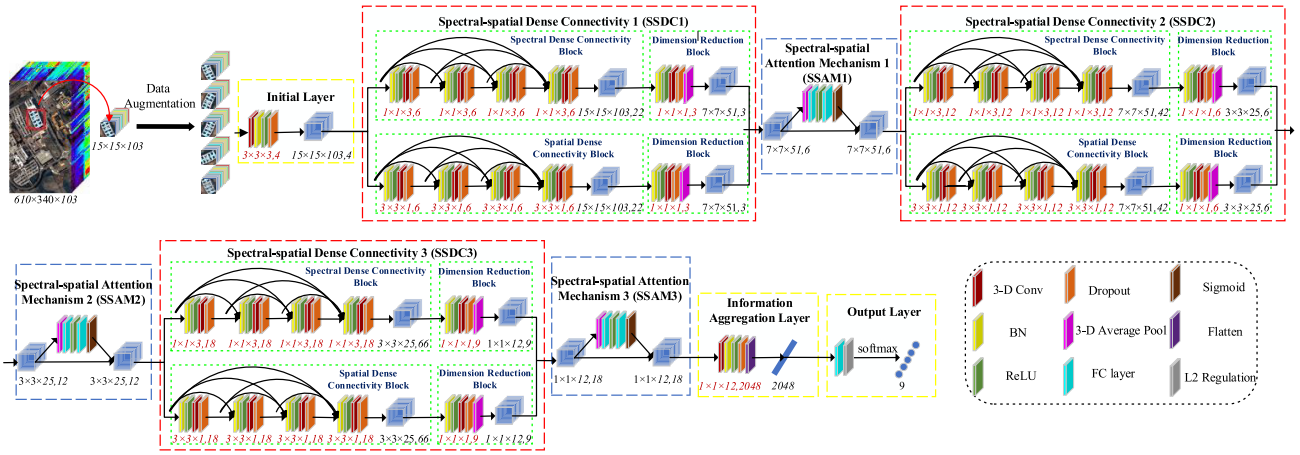

FIGURE 8. The framework of SSDANet for hyperspectral image classification on Pavia University. Where, the size of the 3-D convolutional kernel is represented by red numbers, and the size of the feature maps is represented by black numbers.

4) EXPONENTIAL DECAY LEARNING RATE

The setting of learning rate is very important, which determines whether the model converges to the global optimal value and affects the running speed. If the learning rate is too large, the gradient of the model will oscillate back and forth on both sides of the global optimal solution and cannot converge. And if the learning rate is too small, the convergence speed of the algorithm will be very slow and the training time will increase, resulting in the waste of resources. To solve these problems, the exponential decay learning rate is used. The core idea of exponential decay learning rate is to obtain the sub-optimal solution quickly by using a large learning rate at the beginning, and then gradually reduce the learning rate as the iteration continues, so as to make the gradient converge to the optimal value. The equation of exponential decay learning rate η_d is calculated as follows:

$$\eta_d = \eta \times d_r^{\frac{g_s}{d_s}}, \tag{14}$$

where, η represents the initial learning rate, d_r represents the decay rate, g_s represents the global step and d_s is the decay step.

5) L2 REGULARIZATION

The basic idea of L2 regularization which can alleviate the problem of overfitting is to add an L2 norm penalty to the loss function as a constraint. The loss function J with L2 regularization is calculated as follows:

$$J = J_0 + \frac{\lambda}{2n} \sum_w w^2, \tag{15}$$

where, J_0 represents the original loss function, $\frac{\lambda}{2n} \sum_w w^2$ is the L2 norm penalty, λ is the hyperparameter that controls the ratio of the L2 norm penalty, n is the size of the training samples and w represents the weights of the model.

D. FRAMEWORK OF SSDANET FOR HYPERSPECTRAL CLASSIFICATION

SSDANet has universality. For different datasets, what needs to be changed is the input data, the number of batch size, and the number of output categories. The model based on Pavia University is taken an example to illustrate the design of SSDANet, as shown in Fig. 8. SSDANet is mainly composed of the initial layer, three modules of spectral-spatial dense connectivity which are called SSDC1, SSDC2, and SSDC3 respectively for short, three modules of spectral-spatial attention mechanism which are called SSAM1, SSAM2, and SSAM3 respectively for short, the information aggregation layer, and the output layer. Details of SSDANet are described below.

It is known to us that the size of Pavia University dataset is $610 \times 340 \times 103$. And the input of the model is the data cube with the size of $15 \times 15 \times 103$ selected from the original dataset. First of all, the input data are amplified by the strategy of data augmentation, and the number of training samples obtained is 5 times that of the input samples. After that, the initial layer is used to capture the general features of the training samples. The initial layer is composed of 3-D convolution with the kernel size of $3 \times 3 \times 3 \times 4$, BN, ReLU, and dropout, after which 4 feature maps with the size of $15 \times 15 \times 103$ can be obtained. Then the SSDC1 is adopted to extract the relatively fine features. As illustrated before, the spectral-spatial dense connectivity mainly consists of spectral dense connectivity block, spatial dense connectivity block, and dimension reduction block. In SSDC1, the 3-D convolution with the kernel size of $1 \times 1 \times 3 \times 6$ is adopted in the spectral dense connectivity block to extract spectral features, whereas the 3-D convolution with the size of $3 \times 3 \times 1 \times 6$ is used in the spatial dense connectivity block to extract spatial features. Through the spectral dense connectivity block and spatial dense connectivity block, feature maps with the size of $15 \times 15 \times 103 \times 22$ can be obtained respectively. In the dimension reduction block of the SSDC1,

the 3-D convolution with the kernel size of $1 \times 1 \times 1 \times 3$ and the 3-D average pooling using the filter of $2 \times 2 \times 2$ with the stride of $(2, 2, 2)$ are adopted. After the operation of dimension reduction, the feature maps with the size of $7 \times 7 \times 51 \times 3$ can be obtained. Since spectral features and spatial features need to be concatenated in SSDC1, the feature maps with the size of $7 \times 7 \times 51 \times 6$ can be obtained as input of SSAM1. In the spectral-spatial attention mechanism, the reduction ratio of the fully connected layer is set as 4 in the paper. Next, the feature maps with size of $7 \times 7 \times 51 \times 6$ obtained by the SSAM1 are used as input of SSDC2. Different from the SSDC1, the 3-D convolutional kernel sizes of spectral dense connectivity block, spatial dense connectivity block, and dimension reduction block in SSDC2 are $1 \times 1 \times 3 \times 12$, $3 \times 3 \times 1 \times 12$, and $1 \times 1 \times 1 \times 6$, respectively. And the feature maps with the size of $3 \times 3 \times 25 \times 12$ can be obtained through SSDC2, which are used as the input of SSAM2. The size of the output feature maps of SSAM2 is still $3 \times 3 \times 25 \times 12$. Then, the feature maps with the size of $1 \times 1 \times 12 \times 18$ can be obtained by the SSDC3. The 3-D convolutional kernel sizes of the spectral dense connectivity block, spatial dense connectivity block, and dimension reduction block in the SSDC3 are $1 \times 1 \times 3 \times 18$, $3 \times 3 \times 1 \times 18$, and $1 \times 1 \times 1 \times 9$, respectively. And then, the SSAM3 is used to recalibrate the spectral-spatial features. After that, the information aggregation layer which consists of 3-D convolution, BN, ReLU, Dropout, and the flatten operation is used to aggregate global features into a one-dimensional matrix with the dimension of 2048. Finally, the probabilities of different categories are produced by the output layer. The output layer is mainly composed of the fully connected operation and is optimized by L2 regularization. The final output vector $\hat{y} = [\hat{y}_1, \hat{y}_2, \cdots, \hat{y}_L]$ is obtained after the softmax function. Where, L is the number of categories of the selected dataset.

SSDANet is a deep and wide model with end-to-end structure, which can effectively extract discriminant features from HSIs automatically. In SSDANet, the spectral-spatial dense connectivity is put forward, which can not only extract spectral-spatial features, but also enhance the reuse of features, so as to alleviate the problems of gradient vanishing and protect the integrity of information effectively. Equally important, the spectral-spatial attention mechanism is put forward to learn the importance of each feature channel automatically, so that important spectral-spatial features can be excited and unimportant spectral-spatial features can be suppressed. Additionally, a series of optimization methods including data augmentation, batch normalization, dropout, exponential decay learning rate, and L2 regularization are adopted, which can not only alleviate the overfitting problem, but also can enhance robustness of the proposed model.

IV. EXPERIMENTS AND RESULTS

The datasets adopted, the experimental setup and the experimental results are introduced in this section.

A. DATASETS

The datasets of Pavia University and Indian Pines are adopted for the experiments.

The Pavia University dataset was acquired in 2001. It contains the scenes of Pavia in northern Italy with 9 ground-truth classes. The uncorrected dataset of Pavia University is made up of 610×340 pixels and 115 bands. The geometric resolution is 1.3 m and the wavelength is from 0.43 to 0.86 μm. After removing 12 noise bands, the corrected dataset contains 103 bands.

The Indian Pines dataset was obtained in June 1992. It contains the scenes in Northwestern Indiana with 16 ground-truth classes. The uncorrected Indian Pines is made up of 145×145 pixels and 224 bands. The spatial resolution is 20 m and the wavelength is from 0.4 to 2.5 μm. After removing 24 noise bands, the corrected dataset contains 200 bands.

B. EXPERIMENT SETUP

In this paper, 20% samples of each category were randomly selected as the training set and the remaining 80% as the test set. The batch size of Indian Pines dataset was 32 and that of Pavia University dataset was 10. Besides, the Adam optimizer was adopted to make SSDANet converge quickly. Moreover, the optimization method of exponential decay learning rate was used to improve the performance of SSDANet. Where, the initial learning rate was 0.001, the decay rate was 0.9, and the decay step was 20000. Furthermore, the training iteration was 100000. And the Dropout rate was 0.5.

The experimental hardware platform was a server with Xeon Gold 6139 CPU, Tesla V100 GPU, 16G graphic memory and 64G random access memory. The experimental software platform was based on the Ubuntu18.04 operating system with CUDA10.0.13, Tensorflow1.13.1, Keras2.3.1, and Python3.6.

The classification evaluation indexes adopted in this paper are overall accuracy (OA), average accuracy (AA), and kappa coefficient (K). Where, OA represents the ratio between the number of correctly classified samples and the number of total samples. AA represents the mean value of each category's classification accuracy. And Kappa coefficient measures the consistency between the results and the ground-truth.

C. EXPERIMENT RESULTS

In this section, the proposed method is compared with other methods. The comparison methods include Naive Bayes, Decision Tree, KNN, SVM, 1-D CNN [43], 2-D CNN [63], 3-D CNN [64], HybridSN [65], and the proposed method—SSDANet. Where, Naive Bayes, Decision Tree, KNN, and SVM were implemented by using *scikit learn*.[1] 1-D CNN, 2-D CNN, 3-D CNN and SSDANet were implement via Tensorflow. And HybridSN—a network that combines 2-D CNN and 3-D CNN was realized through Keras. For the sake of fair comparison, 20% samples were used for training in

[1] http://scikit-learn.org

TABLE 1. Results of different classification methods on the Pavia University Dataset.

Class	Naive Bayes	Decision Tree	KNN	SVM	1-D CNN	2-D CNN	3-D CNN	HybridSN	SSDANet
1	66.97	87.46	89.37	94.93	95.93	97.74	98.74	100.00	**100.00**
2	66.71	90.64	97.33	98.43	96.09	97.45	99.75	100.00	**100.00**
3	27.50	60.00	68.21	81.67	82.32	93.93	94.82	99.94	**99.94**
4	92.74	85.93	85.11	96.74	96.49	96.98	98.25	99.31	**99.92**
5	99.63	98.51	99.07	100.00	99.63	100.00	100.00	100.00	**100.00**
6	33.60	69.23	62.90	89.71	92.52	98.83	99.06	100.00	**100.00**
7	87.22	69.08	83.55	88.16	90.04	92.11	99.15	100.00	**100.00**
8	78.07	74.54	85.03	90.02	85.17	95.01	98.37	**99.90**	99.80
9	98.94	99.87	99.87	100.00	99.99	99.87	100.00	100.00	**100.00**
OA (%)	66.60	84.19	88.37	94.60	94.07	97.21	99.04	99.89	**99.97**
AA (%)	72.49	81.70	85.60	93.29	93.13	96.88	98.68	99.92	**99.96**
Kappa (%)	57.42	79.02	84.30	93.30	92.16	96.31	98.73	99.92	**99.97**
Time (s)	30.82	11.88	63.73	37.67	4141.75	3269.94	6975.33	498.89	13318.28

TABLE 2. Results (%) of different classification methods on the Indian Pines Dataset.

Class	Naive Bayes	Decision Tree	KNN	SVM	1-D CNN	2-D CNN	3-D CNN	HybridSN	SSDANet
1	67.57	45.95	32.43	64.86	62.16	75.68	48.65	97.30	**100.00**
2	45.67	59.49	62.20	81.19	83.99	80.58	91.08	96.85	**98.86**
3	9.49	47.89	56.93	71.08	85.84	83.43	89.76	99.70	**100.00**
4	13.68	42.63	32.63	70.00	54.74	66.32	87.89	**100.00**	96.31
5	8.53	85.27	82.43	93.80	90.96	95.87	96.38	99.74	**100.00**
6	76.20	85.96	96.58	97.09	94.69	98.12	99.32	**99.83**	99.66
7	56.52	21.74	69.57	78.26	60.87	47.83	43.48	100.00	**100.00**
8	81.98	91.38	98.17	79.65	97.65	98.43	100.00	100.00	**100.00**
9	56.25	18.75	12.50	50.00	49.99	93.75	100.00	93.75	**100.00**
10	65.42	62.72	73.52	79.18	82.78	82.90	92.54	**98.71**	98.59
11	20.77	66.24	70.67	87.79	76.02	89.71	95.16	99.39	**99.69**
12	29.05	43.16	36.00	76.00	84.63	76.00	90.32	96.63	**97.47**
13	84.76	79.27	92.07	98.78	98.78	98.17	99.39	**99.39**	97.56
14	96.54	83.99	95.16	97.73	95.65	96.15	97.33	100.00	**100.00**
15	26.86	49.84	21.04	60.52	53.40	69.58	77.35	99.68	**99.68**
16	81.33	68.00	85.33	84.00	85.33	89.33	95.67	97.30	**97.33**
OA	45.89	66.61	70.87	85.21	83.57	87.27	93.37	98.83	**99.29**
AA	51.29	59.52	63.58	80.50	78.59	83.87	87.71	98.98	**99.07**
Kappa	39.77	61.97	66.73	83.08	81.34	85.47	92.44	98.64	**99.19**
Time	11.31	2.25	13.29	9.79	1997.20	1806.19	3274.23	214.43	25814.89

all methods. The comparison methods can be classified by two ways. On the one hand, Naive Bayes, Decision Tree, KNN, and SVM belong to traditional classification methods; whereas 1-D CNN, 2-D CNN, 3-D CNN, HybridSN, and SSDANet are all the classification methods based on deep learning. On the other hand, Naive Bayes, Decision Tree, KNN, SVM, and 1-D CNN are all the classification methods based on spectral information; 2-D CNN is the classification method based on spatial information; whereas 3-D CNN, HybridSN, and SSDANet are the classification methods based on spectral-spatial information. Table 1 and Table 2 show the results of different classification methods on the Pavia university dataset and the Indian Pines dataset respectively.

The following conclusions can be got from Tables 1 and Table 2:

1) The classification methods based on deep learning are generally superior to the traditional classification methods. The traditional classification methods extract features manually. Whereas the classification methods based on deep learning can automatically mine features from the data through the hierarchical structure, so they are more powerful than the traditional methods in feature extraction.
2) The classification methods based on spectral-spatial information have better performance than the methods based on spectral information and the methods based on spatial information. This is because the features extracted by the classification methods based on spectral-spatial information include not only spectral information but also spatial information, which can realize the effective use of features.
3) The classification performance of SSDANet can reach the level of state-of-the-art, which is because SSDANet has excellent structure and can extract more discriminant information from HSIs compared with others.
4) In order to fully learn the features from the input data, the running time of classification methods based on

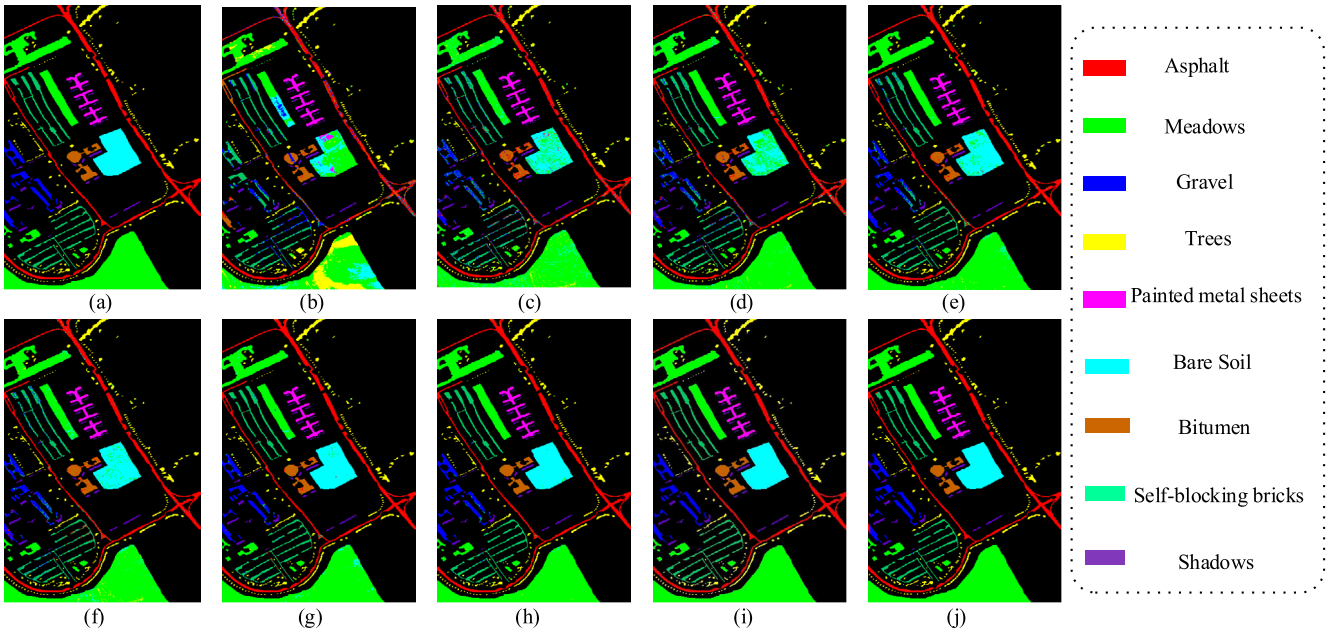

FIGURE 9. Classification maps on the Pavia University dataset. (a) Ground truth; (b) Naive Bayes; (c) Decision Tree; (d) KNN; (e) SVM; (f) 1-D CNN; (g) 2-D CNN; (h) 3-D CNN (i) HybridSN; (j) SSDANet.

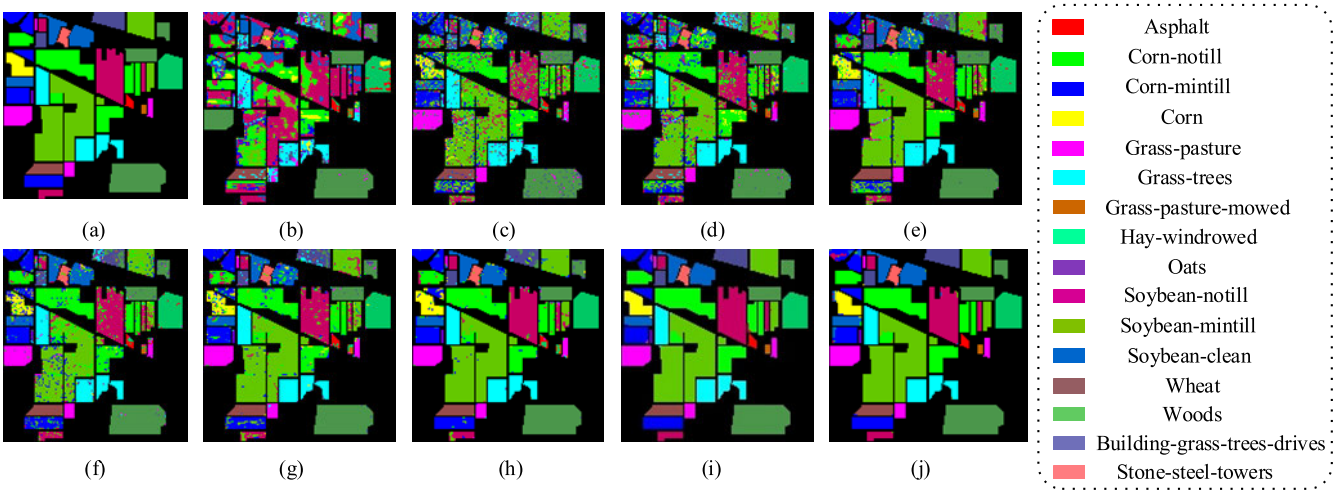

FIGURE 10. Classification maps on the Indian Pines dataset. (a) Ground truth; (b) Naive Bayes; (c) Decision Tree; (d) KNN; (e) SVM; (f) 1-D CNN; (g) 2-D CNN; (h) 3-D CNN (i) HybridSN; (j) SSDANet.

deep learning is generally longer than that of traditional methods. Although the classification methods based on 3-D CNN have higher classification performance compared with other classification methods based on deep learning, there are numerous parameters, so the computational effort is relatively large and the running time is relatively long.

Classification maps of different methods on the Pavia University dataset and the Indian Pines dataset are presented to further validate the performance of SSDANet, as shown in Fig. 9 and Fig. 10.

It can be seen that the classification maps of SSDANet have less noise and the boundaries of objects are clearly defined. Compared with other methods, the classification maps of SSDANet on the two datasets are closest to the ground truth maps. The above analysis can prove the superiority of SSDANet.

V. ANALYSIS AND DISCUSSION

In this part, the analysis and comparison of ablation studies are used to illustrate the importance of each component in SSDANet. In addition, in order to explain the characteristics of SSDANet, the spatial size of input cube and the ratio of training samples are analyzed.

A. ANALISIS AND COMPARISON OF ABLATION STUDIES

Ablation experiments were carried out to prove the effectiveness of SSDANet. The OA, AA, and Kappa of

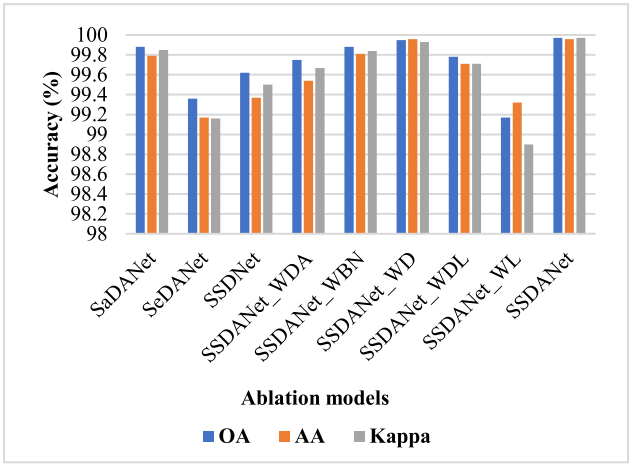

FIGURE 11. Ablation studies on Pavia University.

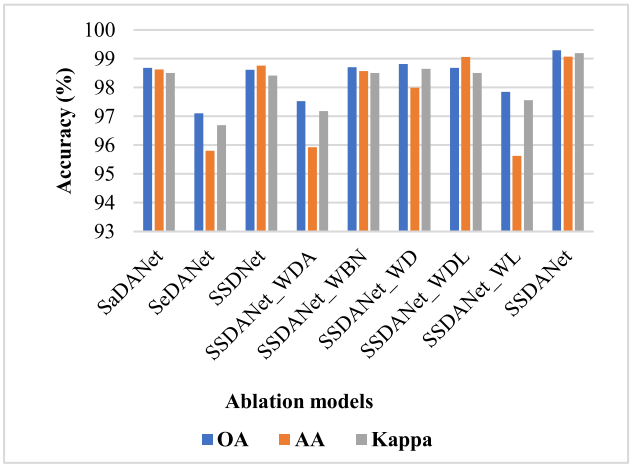

FIGURE 12. Ablation studies on Indian Pines.

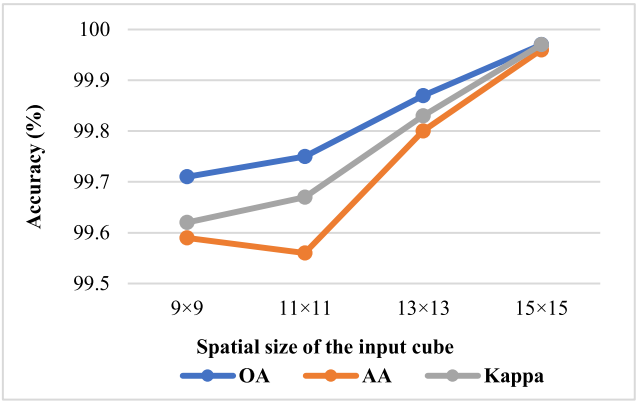

FIGURE 13. The influence of different spatial sizes of the input cube on Pavia University.

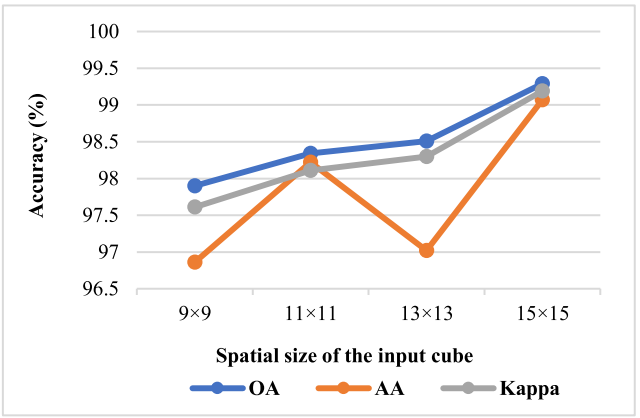

FIGURE 14. The influence of different spatial sizes of the input cube on Indian Pines.

benchmark model—SSDANet were compared with those of eight ablation models—the model without spectral branch (SaDANet), the model without spatial branch (SeDANet), the model without spectral-spatial attention mechanism (SSDNet), the model without data augmentation (SSDANet_WDA), the model without batch normalization (SSDANet_WBN), the model without dropout (SSDANet_WD), the model without exponential decay learning rate (SSDANet_WDL), and the model without L2 regularization (SSDANet_WL) on the Pavia University dataset and the Indian Pines dataset. The other settings of SSDANet and eight ablation models are the same, except for the unused modules. Fig. 11 and Fig. 12 are the comparison results. The performance of SSDANet on the two datasets is higher than that of other ablation models, thus proving the importance of each component in the proposed method.

B. ANALYSIS OF THE SPATIAL SIZE OF THE INPUT CUBE

In the spectral-spatial classification methods of HSIs, it is generally believed that the target pixel and its neighbor pixels belong to the same category. So the input of these methods is usually presented in the form of 3-D cube to reduce the intraclass variance, thus improving the classification performance. And the spatial size of the input cube has great influence on the classification performance. If the spatial size of the input cube is too small, then the receptive field for feature extraction will not be sufficient, resulting in the loss of information and the reduction of the classification ability. And if the spatial size of the input cube is too large, additional noise will be introduced, resulting in the degradation of the model. Therefore, the classification performance is analyzed to find the optimal spatial size of the input cube. Fig. 13 and Fig. 14 show the influence when the spatial size of the input cube is 9×9, 11×11, 13×13, and 15×15 on the Pavia University dataset and Indian Pines dataset. And when the spatial size of the input cube is 15×15, the evaluation indexes reach the optimal values on the two datasets. Therefore, the spatial size of 15×15 is regarded as the most suitable spatial size of the SSDANet's input cube under the condition that the hardware platform allows.

C. ANALYSIS OF TRAINING SAMPLE RATIOS

In the Pavia University dataset and the Indian Pines dataset, 1%, 5%, 10%, 15%, and 20% samples were randomly

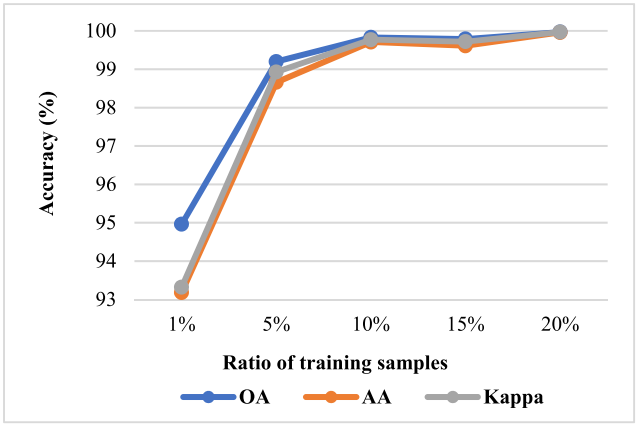

FIGURE 15. The influence of the training sample ratios on Pavia University.

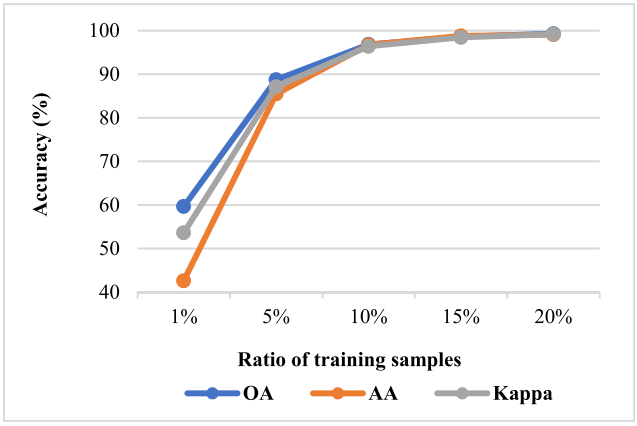

FIGURE 16. The influence of the training sample ratios on Indian Pines.

selected as the training set to explore the impact of different training sample ratios on the performance of SSDANet. The results are shown in Fig. 15 and Fig. 16. When the training sample ratio is small, the problem of overfitting is likely to occur, resulting in poor performance of the model. And as the training sample ratio increases, the learning ability of the model also improves. There are sufficent samples in the Pavia University dataset, so even though the training sample ratio is small, the performance of SSDANet on this dataset is still very high. Whereas the total sample number in the Indian Pines dataset is relatively small, so the training sample ratio has a great influence on the classification results of SSDANet. In other words, the proportion of training samples needs to be relatively large for the model to achieve high performance on the Indian Pines dataset. To balance the performance of the two datasets, 20% training samples are used as the training set.

VI. CONCLUSIONS

A deep and wide network with end-to-end structure for the classification of HSIs—SSDANet has been proposed in the paper. In SSDANet, the spectral-spatial dense connectivity has been put forward, which can learn spectral and spatial features simultaneously and can protect the integrity of information effectively. Equally important, spectral-spatial attention mechanism has been introduced to excite the important spectral-spatial information and suppress the less important spectral-spatial information by the means of squeeze and excitation. In addition, a series of optimization methods have been used to prevent overfitting and improve accuracy of the model. The experiment showed that OA, AA, and Kappa on the datasets of Pavia University and Indian Pines all exceeded 99%, reaching the level of state-of-the-art.

Although the classification methods based on 3-D CNN can adapt to the characteristics of HSIs and improve the classification ability, the computational effort is huge. therefore, in the follow-up study, we will explore how to improve the performance of deep learning-based hyperspectral classification methods with less computational effort.

REFERENCES

[1] D. Landgrebe, "Hyperspectral image data analysis," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 17–28, Aug. 2002.
[2] 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.
[3] L. Zhu, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Generative adversarial networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 9, pp. 5046–5063, Sep. 2018.
[4] N. Caporaso, M. B. Whitworth, S. Grebby, and I. D. Fisk, "Non-destructive analysis of sucrose, caffeine and trigonelline on single green coffee beans by hyperspectral imaging," *Food Res. Int.*, vol. 106, pp. 193–203, Apr. 2018.
[5] N. Yokoya, J. Chan, and K. Segl, "Potential of resolution-enhanced hyperspectral data for mineral mapping using simulated EnMAP and Sentinel-2 images," *Remote Sens.*, vol. 8, no. 3, p. 172, Feb. 2016.
[6] L. Liang, L. Di, L. Zhang, M. Deng, Z. Qin, S. Zhao, and H. Lin, "Estimation of crop LAI using hyperspectral vegetation indices and a hybrid inversion method," *Remote Sens. Environ.*, vol. 165, pp. 123–134, Aug. 2015.
[7] H. F. Tan, T. W. Luo, G. Yang, and Q. Q. Meng, "Research on background depression in hyperspectral image anomaly detection," *J. Optoelectron. Laser*, vol. 27, no. 2, pp. 177–181, 2016.
[8] G. Hughes, "On the mean accuracy of statistical pattern recognizers," *IEEE Trans. Inf. Theory*, vol. 14, no. 1, pp. 55–63, Jan. 1968.
[9] C. Zhang, G. Li, S. Du, W. Tan, and F. Gao, "Three-dimensional densely connected convolutional network for hyperspectral remote sensing image classification," *J. Appl. Remote Sens.*, vol. 13, no. 1, 2019, Art. no. 016519.
[10] J. M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. Nasrabadi, and J. Chanussot, "Hyperspectral remote sensing data analysis and future challenges," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 2, pp. 6–36, Jun. 2013.
[11] X. Tang, K. Liu, X. Wang, F. Gao, J. Macro, and W. D. Widanage, "Model migration neural network for predicting battery aging trajectories," *IEEE Trans. Transport. Electrific.*, vol. 6, no. 2, pp. 363–374, Jun. 2020.
[12] K. Liu, Y. Shang, Q. Ouyang, and W. D. Widanage, "A data-driven approach with uncertainty quantification for predicting future capacities and remaining useful life of lithium-ion battery," *IEEE Trans. Ind. Electron.*, early access, Mar. 18, 2020, doi: 10.1109/TIE.2020.2973876.
[13] B. Heinrichs and S. B. Eickhoff, "Your evidence? Machine learning algorithms for medical diagnosis and prediction," *Human Brain Mapping*, vol. 41, no. 6, pp. 1435–1444, Apr. 2020.
[14] K. Liu, Y. Li, X. Hu, M. Lucu, and W. D. Widanage, "Gaussian process regression with automatic relevance determination kernel for calendar aging prediction of lithium-ion batteries," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 3767–3777, Jun. 2020.
[15] K. Liu, X. Hu, Z. Wei, Y. Li, and Y. Jiang, "Modified Gaussian process regression models for cyclic capacity prediction of lithium-ion batteries," *IEEE Trans. Transport. Electrific.*, vol. 5, no. 4, pp. 1225–1236, Dec. 2019.

[16] X. Tang, K. Liu, X. Wang, B. Liu, F. Gao, and W. D. Widanage, "Real-time aging trajectory prediction using a base model-oriented gradient-correction particle filter for lithium-ion batteries," *J. Power Sources*, vol. 440, Nov. 2019, Art. no. 227118.

[17] Y. Ni, D. Barzman, A. Bachtel, M. Griffey, A. Osborn, and M. Sorter, "Finding warning markers: Leveraging natural language processing and machine learning technologies to detect risk of school violence," *Int. J. Med. Informat.*, vol. 139, Jul. 2020, Art. no. 104137.

[18] Y. Fujishiro, T. Furukawa, and S. Maruo, "Simple autofocusing method by image processing using transmission images for large-scale two-photon lithography," *Opt. Express*, vol. 28, no. 8, p. 12342, 2020.

[19] Z. Zhong, J. Li, Z. Luo, and M. Chapman, "Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 847–858, Feb. 2017.

[20] X. Jia, B.-C. Kuo, and M. M. Crawford, "Feature mining for hyperspectral image classification," *Proc. IEEE*, vol. 101, no. 3, pp. 676–697, Mar. 2013.

[21] Y. Xu, B. Du, and L. Zhang, "Beyond the patchwise classification: Spectral–spatial fully convolutional networks for hyperspectral image classification," *IEEE Trans. Big Data*, early access, Jun. 17, 2020, doi: 10.1109/TBDATA.2019.2923243.

[22] J. Wang and C.-I. Chang, "Independent component analysis-based dimensionality reduction with applications in hyperspectral image analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 6, pp. 1586–1600, Jun. 2006.

[23] T. V. Bandos, L. Bruzzone, and G. Camps-Valls, "Classification of hyperspectral images with regularized linear discriminant analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 3, pp. 862–873, Mar. 2009.

[24] X. Kang, X. Xiang, S. Li, and J. A. Benediktsson, "PCA-based edge-preserving features for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 7140–7151, Dec. 2017.

[25] A. A. Nielsen, "Kernel maximum autocorrelation factor and minimum noise fraction transformations," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 612–624, Mar. 2011.

[26] L. Bruzzone, F. Roli, and S. B. Serpico, "An extension of the Jeffreys-Matusita distance to multiclass cases for feature selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 33, no. 6, pp. 1318–1321, Nov. 1995.

[27] N. Keshava, "Distance metrics and band selection in hyperspectral processing with applications to material identification and spectral libraries," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 7, pp. 1552–1565, Jul. 2004.

[28] T. Kailath, "The divergence and bhattacharyya distance measures in signal selection," *IEEE Trans. Commun.*, vol. COMM-15, no. 1, pp. 52–60, Feb. 1967.

[29] E. Blanzieri and F. Melgani, "Nearest neighbor classification of remote sensing images with the maximal margin principle," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 6, pp. 1804–1811, Jun. 2008.

[30] J. Xia, L. Bombrun, Y. Berthoumieu, C. Germain, and P. Du, "Spectral-spatial rotation forest for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 10, pp. 4605–4613, Jul. 2017.

[31] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, Aug. 2004.

[32] J. A. Benediktsson, J. A. Palmason, and J. R. Sveinsson, "Classification of hyperspectral data from urban areas based on extended morphological profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 480–491, Mar. 2005.

[33] L. Fang, S. Li, W. Duan, J. Ren, and J. A. Benediktsson, "Classification of hyperspectral images by exploiting Spectral–Spatial information of superpixel via multiple kernels," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 12, pp. 6663–6674, Dec. 2015.

[34] G. Camps-Valls and L. Bruzzone, "Kernel-based methods for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 6, pp. 1351–1362, Jun. 2005.

[35] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification using dictionary-based sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 10, pp. 3973–3985, Oct. 2011.

[36] C. Zhang, G. Li, and S. Du, "Multi-scale dense networks for hyperspectral remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 9201–9222, Aug. 2019.

[37] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.

[38] B. Schölkopf, J. Platt, and T. Hofmann, "Greedy layer-wise training of deep networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, vol. 19, 2007, pp. 153–160.

[39] K. Fukushima, S. Miyake, and T. Ito, "Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-13, no. 5, pp. 826–834, May 1970.

[40] P. Zhong, Z. Gong, S. Li, and C.-B. Schonlieb, "Learning to diversify deep belief networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 6, pp. 3516–3530, Jun. 2017.

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

[42] L. Zhang, L. Zhang, and B. Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geosci. Remote Sens. Mag.*, vol. 4, no. 2, pp. 22–40, Jun. 2016.

[43] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *J. Sensors*, vol. 2015, pp. 1–12, 2015.

[44] W. Li, G. Wu, F. Zhang, and Q. Du, "Hyperspectral image classification using deep pixel-pair features," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 844–853, Feb. 2017.

[45] J. Yang, Y.-Q. Zhao, and J. C.-W. Chan, "Learning and transferring deep joint spectral–spatial features for hyperspectral classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 8, pp. 4729–4742, Aug. 2017.

[46] J. Feng, J. Chen, L. Liu, X. Cao, X. Zhang, L. Jiao, and T. Yu, "CNN-based multilayer spatial–spectral feature fusion and sample augmentation with local and nonlocal constraints for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 4, pp. 1299–1313, Apr. 2019.

[47] B. Liu, X. Yu, P. Zhang, X. Tan, R. Wang, and L. Zhi, "Spectral-spatial classification of hyperspectral image using three-dimensional convolution network," *J. Appl. Remote Sens.*, vol. 12, no. 1, 2018, Art. no. 016005.

[48] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[49] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Neural Inf. Process. Syst.*, 2012, vol. 141, no. 5, pp. 1097–1105.

[50] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.

[51] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015.

[52] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[53] G. Huang, Z. Liu, L. V. Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. Comput. Vis. Pattern Recognit.*, 2017, pp. 2261–2269.

[54] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, Aug. 2020.

[55] Y. Yang, Z. Zhong, T. Shen, and Z. Lin, "Convolutional neural networks with alternately updated clique," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2413–2422.

[56] Z. Zhong, J. Li, Z. Luo, and M. Chapman, "Spectral-spatial residual network for hyperspectral image classification: A 3-D deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 847–858, Feb. 2018.

[57] W. Wang, S. Dou, Z. Jiang, and L. Sun, "A fast dense spectral–spatial convolution network framework for hyperspectral images classification," *Remote Sens.*, vol. 10, no. 7, p. 1068, Jul. 2018.

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

[59] B. Fang, Y. Li, H. Zhang, and J. C. Chan, "Hyperspectral images classification based on dense convolutional networks with spectral-wise attention mechanism," *Remote Sensing*, vol. 11, no. 2, p. 159, 2019.

[60] L. Wang, J. Peng, and W. Sun, "Spatial–spectral squeeze-and-excitation residual network for hyperspectral image classification," *Remote Sens.*, vol. 11, no. 7, p. 884, Apr. 2019.

[61] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," *Neural Comput.*, vol. 29, no. 9, pp. 2352–2449, Sep. 2017.

[62] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
[63] H. Zhang, Y. Li, Y. Zhang, and Q. Shen, "Spectral-spatial classification of hyperspectral imagery using a dual-channel convolutional neural network," *Remote Sens. Lett.*, vol. 8, no. 5, pp. 438–447, May 2017.
[64] Y. Li, H. Zhang, and Q. Shen, "Spectral–Spatial classification of hyperspectral imagery with 3D convolutional neural network," *Remote Sens.*, vol. 9, no. 1, p. 67, Jan. 2017.
[65] 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.

XIN ZHANG received the bachelor's degree from Northeastern University, Qinhuangdao, in 2016. She is currently pursuing the Ph.D. degree with the Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences. Her research interests include deep learning and image classification.

YONGCHENG WANG received the bachelor's degree from Jilin University, in 2003, and the Ph.D. degree from Chinese Academy of Sciences, in 2010. He is currently a Researcher in Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences. His research interests include image engineering and space payload embedded systems.

NING ZHANG received the bachelor's degree from Northeastern University, Qinhuangdao, in 2017. She is currently pursuing the Ph.D. degree with the Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences. Her research interests include image super-resolution and deep learning.

DONGDONG XU received the bachelor's degree from Shandong University, in 2013, the master's degree from the Harbin Institute of Technology, in 2015, and the Ph.D. degree from Chinese Academy of Sciences, in 2020. He is currently an Assistant Researcher in Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences. His research interests include deep learning, image fusion, and embedded system software development.

HUIYUAN LUO received the B.S. degree from the Harbin Institute of Technology, Weihai, in 2016. He is currently pursuing the Ph.D. degree with the Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences. His current research interests include saliency detection and deep learning.

BO CHEN received the bachelor's degree from Jilin University, in 1984, and the Ph.D. degree from Chinese Academy of Sciences, in 2003. He is currently a Researcher and a Professor in Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences. His research interest includes technology of space optics.

GUANGLI BEN received the bachelor's and master's degrees from Harbin Engineering University, in 2013 and 2016, respectively. He is currently pursuing the Ph.D. degree in the Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences. He is also a Research Assistant in the Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences. His research interests include digital signal processing and space payload embedded systems.