remote sensing

MDPI

Article

Dual-Branch Attention-Assisted CNN for Hyperspectral Image Classification

Wei Huang [1], Zhuobing Zhao [1], Le Sun [2,3,*] and Ming Ju [1]

[1] College of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China
[2] School of Computer Science, Nanjing University of Information Science and Technology, Nanjing 210044, China
[3] Jiangsu Collaborative Innovation Center of Atmospheric Environment and Equipment Technology (CICAEET), Nanjing University of Information Science and Technology, Nanjing 210044, China
* Correspondence: sunlecncom@nuist.edu.cn; Tel.: +86-153-6610-5906

Abstract: Convolutional neural network (CNN)-based hyperspectral image (HSI) classification models have developed rapidly in recent years due to their superiority. However, recent deep learning methods based on CNN tend to be deep networks with multiple parameters, which inevitably resulted in information redundancy and increased computational cost. We propose a dual-branch attention-assisted CNN (DBAA-CNN) for HSI classification to address these problems. The network consists of spatial-spectral and spectral attention branches. The spatial-spectral branch integrates multi-scale spatial information with cross-channel attention by extracting spatial–spectral information jointly utilizing a 3-D CNN and a pyramid squeeze-and-excitation attention (PSA) module. The spectral branch maps the original features to the spectral interaction space for feature representation and learning by adding an attention module. Finally, the spectral and spatial features are combined and input into the linear layer to generate the sample label. We conducted tests with three common hyperspectral datasets to test the efficacy of the framework. Our method outperformed state-of-the-art HSI classification algorithms based on classification accuracy and processing time.

Keywords: hyperspectral image (HSI) classification; pyramid squeeze-and-excitation attention (PSA); spatial–spectral; cross-channel attention

Citation: Huang, W.; Zhao, Z.; Sun, L.; Ju, M. Dual-Branch Attention-Assisted CNN for Hyperspectral Image Classification. *Remote Sens.* **2022**, *14*, 6158. https://doi.org/10.3390/rs14236158

Academic Editor: Edoardo Pasolli

Received: 8 November 2022
Accepted: 1 December 2022
Published: 5 December 2022

1. Introduction

Hyperspectral imaging technology has improved with the advancement of remote sensing technologies. The image data captured by hyperspectral sensors are more accurate, which has promoted the use of hyperspectral images (HSI) in numerous applications, including target detection [1,2], environmental monitoring [3,4], military reconnaissance [5,6], agricultural assessment [7,8], etc. Compared with ordinary images, HSIs have hundreds of continuous spectral bands with rich spectral information and higher resolution [9], so they can distinguish feature categories precisely. The application of HSI classification [10–12] is one of the main research directions in the field of remote sensing at present and determining methods for classifying each pixel quickly and accurately is the core of this problem [13].

A growing number of scholars have investigated HSI classification [14,15] in recent years. Awad et al. [14] proposed a supervised algorithm for HSI classification using spectral information. Wambugu et al. [16] provide a full discussion of the problem with insufficient training samples and summarize the main current solutions. Fabiyi et al. [17] proposed a folded LDA method for dimensionality reduction of small sample data and reduced memory requirements. Polynomial logistic regression [18–20], the K-nearest neighbor (KNN) algorithm [21], and support vector machines (SVM) are examples of traditional classification methods [22–24], which are mainly divided into two steps: feature extraction and training classifier. However, these methods rely on human judgment and labeling because they depend on manual features, which can be labor-intensive and time-consuming.

It is hard to enhance the accuracy of classification further since the information extracted by traditional methods is often limited and has weak generalization ability.

Deep learning models have been demonstrated to provide enormous advantages in the computer vision field during the last decade [25–28]. By using an end-to-end framework to generate more discriminative features, deep learning-based algorithms (unlike conventional classification methods) may optimize both the feature extraction task and the classification problem. Some deep learning models, such as Stacked Auto Encoder (SAE) [29] Networks, Deep Belief Networks (DBN) [30], and Recurrent Neural Networks (RNN) [31], have also been widely employed in the field of HSI classification. However, all these networks require vector data inputs, so they are more suitable for extracting spectral information, but inevitably cause the loss of spatial information. CNNs, a more popular deep learning model [32], extract features more flexibly through local contacts and significantly lower the number of parameters by sharing weights.

HSI classification models based on CNNs can automatically learn and extract distinguishable features in images without much human intervention. Hu et al. [33] designed a method for extracting spectral features using a 1-D CNN. Although pixels in a region can be classified based on spectral information, the results obtained by using spectral information alone are not accurate enough due to the existence of homospectral and heterospectral phenomena in HSI, and combining spatial information can significantly increase the classification accuracy. To emphasize the importance of using spatial information for feature extraction, Makantasis et al. [34] proposed a 2-D CNN. However, these methods did not take full advantage of the 3-D nature of HSI, so researchers proposed 3-D CNN-based methods [35–37] to directly extract 3-D spatial–spectral information using 3-D convolution kernels. To address the gradient disappearing issue, Zhong et al. constructed an end-to-end spatial–spectral residual network (SSRN) [38], which can use 3-D blocks as the original inputs and add residual connections. For HSI classification, Paoletti et al. presented a deep pyramidal residual network [39]. By including the residual structure, the suggested pyramid structure gradually raises the convolutional layer's feature mapping dimension, which reduces the time complexity while obtaining more feature information. However, the deep 3-D CNN model inevitably causes an increase in time and computational effort. A HybridSN network that merged 2-D and 3-D CNNs to jointly extract spatial–spectral information and generate improved classification results was developed by Roy, S.K. et al. [40] as a solution to this issue. Some research has attempted to create a dual-branch network, where one branch obtains spatial information while the other obtains spectral information and the combined results are input to the classifier, in an effort to further reduce the number of parameters and time spent. For example, 1-D and 2-D CNNs are employed to extract spectral features and spatial information, respectively, in the parallel dual-branch model presented in [41].

It is well known that HSI, with spatial information and rich spectral information along with different spectral bands and spatial locations, contributes differently to classification prediction. Making full use of this information can be very helpful for classification. Researchers have added attention mechanisms to computer vision tasks in an effort to imitate human visual perception [42–44]. For HSI classification, attention mechanisms have recently been used widely [45–47]. For example, Haut et al. [45] suggested a model mixing residual networks and attention mechanisms and Sun et al. [46] developed serial spatial-spectral attention networks (SSAN). A double-branch dual-attention (DBDA) mechanism network that captures spatial–spectral features separately was also proposed by Li et al. [47]. It can be observed that adding attention mechanisms to CNNs can give better classification performance and contribute more to the prediction of spectral bands.

Transformer is a new deep learning model that introduces a self-attentive mechanism and a feed-forward neural network. There has been a great success with the transformer model in natural language processing (NLP) [48,49]. Recently, transformer models, named "vision transformer", have also been used to classify images [50]. Hu et al. [51] proposed an unsupervised framework for HSI classification based on a transformer model and con-

trastive learning. Qing et al. [52] proposed a self-attention-based transformer (SAT) model and Hong et al. [53] presented a novel transformer-based network model (SpectralFormer), which implements grouped spectral embedding. By naturally combining a backbone CNN with a transformer structure, Sun et al. [54] created the spatial–spectral tokenization transformer (SSFTT) method.

Existing methods have demonstrated good results, but the model is too complex, leading to long training and testing periods. The current attention-based classification methods simply mix spatial and spectral features, which leads to the neglect of the special structure of HSI. In addition, the use of deeper 3-D CNNs increases the risk of the overfitting phenomenon, which reduces the classification performance of HSI. We designed a novel dual-branch CNN based on spatial–spectral attention for HSI classification to address these problems. The spatial–spectral branch extracts the spatial–spectral information jointly by combining 3D convolution and pyramid squeeze-and-excitation attention (PSA) modules, and via the use of a designed spectral band attention module, the spectral attention branch effectively extracts the spectral information. Then, the features of the dual-branch are connected and each pixel's label is determined using a softmax-based linear classifier. The contribution of this paper can be summarized as threefold:

1. To fully utilize the spatial-spectral features of HSI, we propose a new dual-branch network for classification. It can extract enough different information, where the spectral attention branch extracts more effective spectral features from HSI and connects with the features extracted by the spatial–spectral branch, to achieve higher classification accuracy.
2. Considering the limited training samples, the spatial–spectral branch is designed to extract shallow spatial–spectral features using 3D convolution, and then to use the PSA module to learn richer multi-scale spatial information, while adaptively assigning attention weights to the spectral channels.
3. We designed the spectral attention branch, which uses 2-D CNN to map the original features into the spectral interaction space to obtain a spectral weight matrix, so as to obtain more discriminative spectral information.

The rest of the article is organized as follows. Section 2 presents materials and methods, including convolution, attention mechanisms and the DBAA-CNN classification method. Section 3 describes the datasets and experimental results. Section 4 offers a comprehensive analysis. Finally, Section 5 concludes the paper.

2. Materials and Methods

2.1. Related Work

2.1.1. Basics of CNNs for HSI Classification

CNNs use shared weights for each input, which greatly reduces the number of parameters. In addition, CNNs use local connectivity to extract contextual feature information. Thus, CNNs tend to have better generalization ability when dealing with image problems. In this paper, three types of CNN are used for feature extraction—1-D, 2-D and 3-D. Usually, 2-D CNNs are used in the image processing field. The convolutional layer is the main difference between the three CNNs, which we describe in detail.

The 1-D convolution uses a 1-D convolution kernel to perform sliding in one dimension to achieve feature extraction in the spectral dimension. The following is the calculation equation for $v_{i,j}^{x}$, which indicates the neuron at position x on the *j-th* feature map and the *i-th* layer.

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

where $f(\cdot)$ is the activation function, m is the feature map's index in the $(i-1)$-*th* th layer, L_i is the length of one-dimensional convolution kernel, l is the index of convolution kernel, $k_{i,j,m}^{l}$ is the value of the convolution kernel, and $b_{i,j}$ is the bias.

2-D CNN is a two-dimensional convolutional kernel that slides along two dimensions on the data. The value of the neuron $v_{i,j}^{x,y}$ at position (x, y) on the $j-th$ feature map in the $i-th$ layer can calculated by:

$$v_{i,j}^{x,y} = f\left(\sum_{m}\sum_{l=0}^{L_i-1}\sum_{w=0}^{W_{i-1}} k_{i,j,m}^{l,w} v_{(i-1),m}^{(x+l),(y+w)} + b_{i,j}\right) \quad (2)$$

where $k_{i,j,m}^{l,w}$ is the value of the convolution kernel at position (l, w) and W_i is the width of the convolution kernel.

The 3-D CNN computes the 3D feature map from the three-dimensional input data with a 3D convolutional kernel, which can realize the sharing of weights at different locations and in pixel and depth space. The equation calculating $v_{i,j}^{x,y,z}$, which represents the neuron at position (x, y, z) of the j th feature map in the i th layer, can be expressed by:

$$v_{i,j}^{x,y,z} = f\left(\sum_{m}\sum_{l=0}^{L_i-1}\sum_{w=0}^{W_{i-1}}\sum_{d=0}^{D_i-1} k_{i,j,m}^{l,w,d} v_{(i-1),m}^{(x+l),(y+w),(z+d)} + b_{i,j}\right) \quad (3)$$

where $k_{i,j,m}^{l,w,d}$ is the weight of the convolutional kernel at position (l, w, d) on the m th feature map, D_i is the spectral dimensions of the convolution kernel.

As shown in Figure 1, we used the cube block x_k in layer k as input, where x_k consisting of n_k features of size $w_k \times w_k \times b_k$ and a 3D convolution layer D_{k+1} in layer $k+1$ consisting of n_{k+1} convolution kernels of size $d_{k+1} \times d_{k+1} \times m_{k+1}$ with step size set to (s_1, s_1, s_2). The convolution operation can generate a 3D feature cube x_{k+1} consisting of n_{k+1} features of size $w_{k+1} \times w_{k+1} \times b_{k+1}$, where the output features have width and height $w_{k+1} = (w_k - d_{k+1} + 1)/s_1$, and spectral dimension $b_{k+1} = (b_k - m_{k+1} + 1)/s_2$.

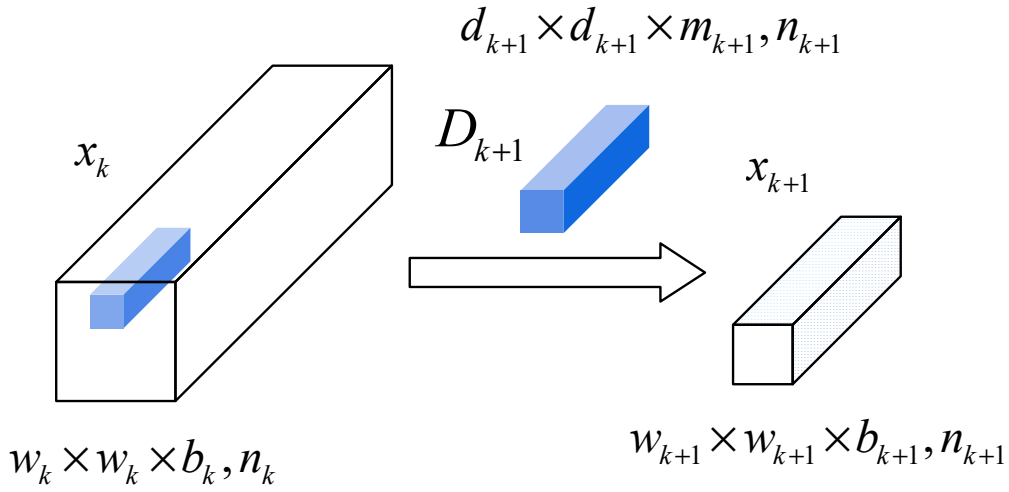


Figure 1. The structure of 3-D convolution.

2.1.2. Squeeze-and-Excitation (SE) Block

Many studies have demonstrated the critical role of visual attention mechanisms in the field of human perception. Inspired by this, many researchers have tried to introduce attentional mechanisms into the field of computer vision [38–40] to improve the efficiency of models, and have had good results.

Recently, Hu et al. [44] presented a light modular SE block that selectively emphasizes the significance of each channel by modeling the interdependencies across channels, increasing speed while reducing the model parameters. The SE block usually has two components: squeeze and excitation. As shown in Figure 2, we let $X \in R^{H \times W \times C}$ represent the input feature map, where W, H, C denotes its width, height and the number of input

channels, respectively. The squeeze operation was performed using a global average pooling operation on X. The features were compressed along the spatial dimension, the spatial dimension of X was compressed from $H \times W$ to 1×1. Each two-dimensional feature map becomes a real number, which was equivalent to the pooling operation with a global perceptual field, and the number of channels C was kept constant, and for each channel, there was a real number corresponding to it. The feature map thus obtained has a global perceptual field. The following equation was used to compute the global average pooling (GAP) operation:

$$S_C = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} X_C(i,j) \tag{4}$$

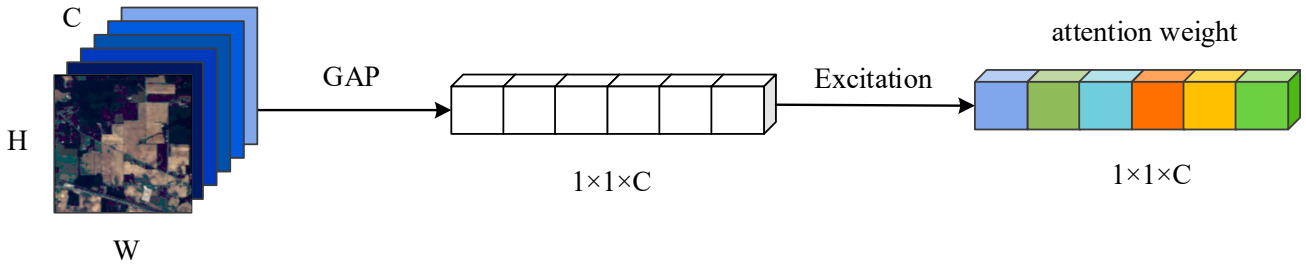


Figure 2. SE block.

The squeeze operation embedded the global information into the feature vector $S_C \in R^{1\times1\times C}$, followed by the excitation operation, wherein two fully connected (FC) layers obtained the feature weights for each channel in the feature map. The weighted features were used as input to the next layer of the network. The attention weight of the *c-th* channel in the SE block can be calculated as follows:

$$E_C = \sigma(W_2\delta(W_1(S_C))) \tag{5}$$

where symbols δ and σ denote the ReLU and sigmoid activation functions, respectively, $W_1 \in \frac{C}{r} \times C$ and $W_2 \in C \times \frac{C}{r}$ denote the weights of the two FC layers, where r represents the reduction ratio. The output feature channels of S_C were matched with the input feature channels of E_C. The SE block allowed the output vector E_C to obtain global information and recalibrated the feature cube X in the channel dimension, enhancing the contributing features and suppressing the useless ones.

2.2. Proposed Method

The proposed dual-branch CNN with spatial–spectral attention is shown in Figure 3. It has two branches: a spatial–spectral branch and another for spectral attention. The first branch includes a spatial–spectral feature extraction module and a PSA module. Shallow spatial–spectral features are directly extracted from the input 3D cut blocks with the spatial–spectral feature extraction module using 3D convolution. The PSA module further extracts spatial–spectral features using multi-scale spatial blocks and cross-channel attention mechanism. The second branch extracts the spectral features by giving the spectral bands weights via spectral attention mechanism. The details of the three modules are specified below.

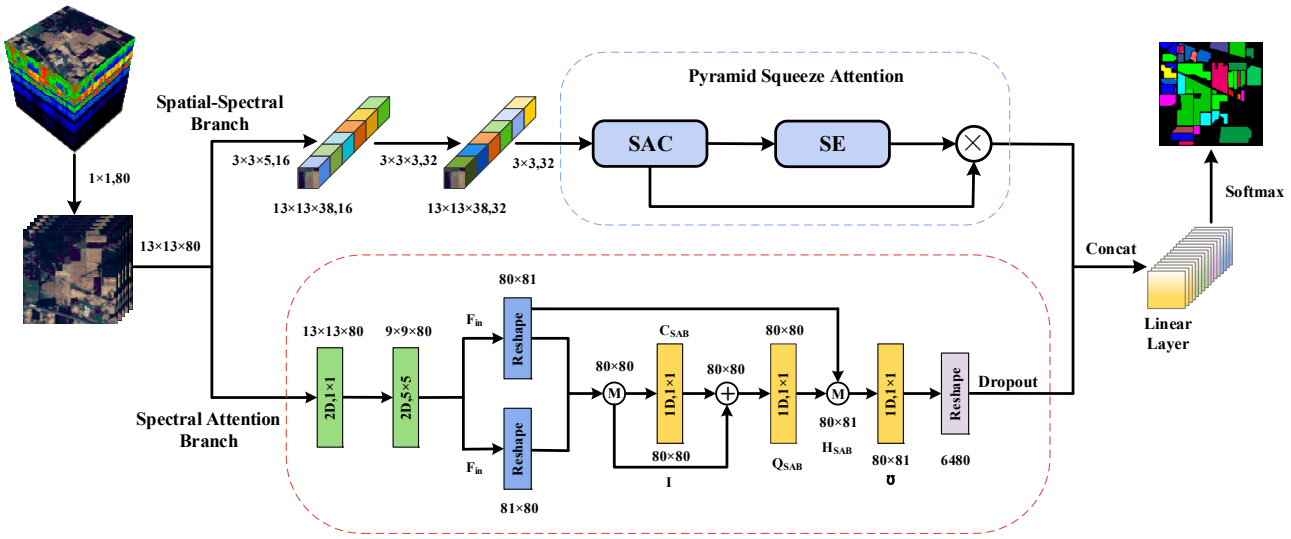


Figure 3. Framework of the DBAA-CNN for HSI classification.

2.2.1. Spatial–Spectral Branch

(1) Spatial–spectral feature extraction module

The original HSI input data is represented as $D \in R^{m \times n \times l}$, where m and n are the width and height of the spatial dimension, respectively, and l denotes the number of spectral bands. HSI contains more bands and each band carries different information for classification. Using all the bands for feature extraction would lead to data redundancy, and the dimensionality reduction method of PCA [55] will drop some bands, which will inevitably cause information loss. Therefore, we performed feature compression in the spectral dimension using 1×1 convolution to remove useless spectral information for the purpose of dimensionality reduction.

We used $I \in R^{m \times n \times b}$ to denote the input after dimensionality reduction, where b represents the number of bands after dimensionality reduction.

Then, the HSI data I is subjected to a blocking operation, and each 3D adjacent region block is represented by $P \in R^{s \times s \times b}$, where $s \times s$ denotes the size of the block. Each block's center pixel location is denoted by (x_i, x_j), where $0 \leq i < m$ and $0 \leq j < n$. The label of the center pixel determines the category label for each block. Since the edge pixels cannot extract the adjacent regions, a fill operation is carried out on them. The size of the fill is set to $(s-1)/2$. Thus, all pixels and bands are covered by the above operation, and the final number of cut blocks obtained is $m \times n$. The unlabeled samples are also removed, and the remaining data are split into two parts: the training sets and test sets.

Compared with 2D convolution to extract features in spatial dimension, 3D convolution can jointly extract spatial–spectral features of HSI, but it also increases the computational effort. Next, the spatial–spectral information of each block is extracted using two 3D convolutional layers. The 3D convolutional layers accept each block of size $s \times s \times b$ as input data. Equation (3) can be used to calculate the value. Assume that the 3D convolution layer contains d_0 convolution kernels of size $d_1 \times d_2 \times d_3$. By convolving this layer, d_0 blocks of 3D cubes of size $(s-d_1+1) \times (s-d_2+1) \times (b-d_3+1)$ are generated. After two layers of 3D convolution operations, we add the rearrangement operation to adjust the feature map and input it to the PSA module.

(2) PSA module.

The shallow spatial–spectral features that can be obtained after two layers of 3D convolution operations are not sufficient to fully describe the feature information. The PSA module [56] can learn richer multi-scale feature representations and adaptively recalibrate the cross-channel attention weights. Due to the lightweight advantage of the module, it

can also improve the model's speed. Figure 4 shows the specific flow of the PSA module. The module consists of four main steps. First, the spatial information at different scales on each channel feature map is obtained through the multi-scale pyramid structure. After that, multi-scale feature maps are input into the SE block to establish the attention mechanism on the multi-scale feature maps channels. Next, the multi-scale attentional channel weights are recalibrated by the softmax algorithm. To obtain the end result, the weights are multiplied by the feature map in the first step to generate a rich multi-scale spatial-spectral representation.

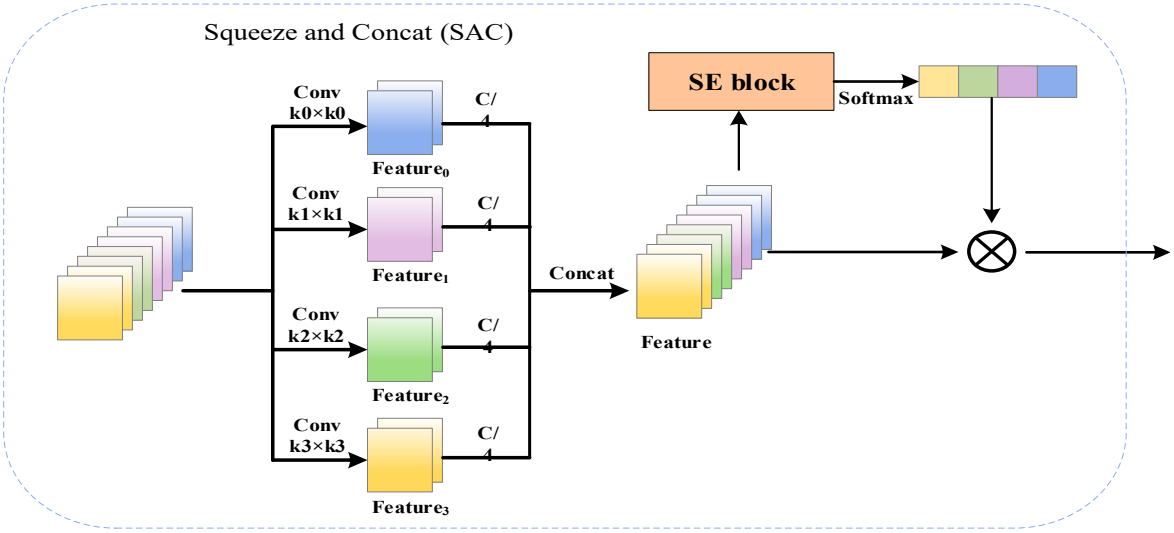


Figure 4. Detailed description of the proposed PSA module when $g = 4$.

In the SAC module, the multiscale pyramid structure implements the extraction of multiscale features. We extracted features in parallel by means of multiple branches, with each branch using a differently-sized convolutional kernel to obtain features with different perceptual fields; the number of input channels for each branch is C, and C/g is the output channel dimension of each branch, where g is the number of groupings. Additionally, padding should be added to ensure that each branch has the same size output feature map. Concatenate the feature maps of multiple branches are to obtain the entire multiscale feature map $F \in R^{H \times W \times C}$, which is obtained from the following equation:

$$F = \text{Concat}([Conv(k_i \times k_i)(X)]), i = 0, 1, 2 \cdots g - 1 \tag{6}$$

where $k \times k$ is the convolution kernel size, and the convolution kernel size is set to $k_i \times k_i = (2i + 3) \times (2i + 3)$ in this paper.

The attention weights may be obtained by the SE block from feature maps of multi-scale. The feature maps' attention weights are recalculated at different scales using the softmax operation, and this step achieves the interaction of local and global information. Afterwards, the feature vectors and attention weights are concatenated to obtain multiscale feature weights. After multiplying the weights with the feature maps of the corresponding scales, the concatenation operation is used to construct the complete feature representation. The following is the specific formula:

$$T_i = F_i \otimes \text{Softmax}(SEWeihgt(F_i)) i = 0, 1, 2, \cdots g - 1 \tag{7}$$

$$Z = Concat\left(\left[T_0, T_1, \cdots, T_{g-1}\right]\right) \tag{8}$$

where F_i represents the feature map at different scales, T_i is the feature map that is given the multi-scale channel attention weight, and $\otimes$ is the multiplication operation on the channel.

2.2.2. Spectral Attention Branch

The HSI has hundreds of narrow spectral bands, unlike RGB images, which only have three channels. We not only used the spectral information of HSI for feature extraction, but the information on the spectral band also has a significant impact on the classification. In order to effectively utilize the spectral information of HSI and reduce the redundancy among the bands, we designed the spectral branching based on the attention mechanism, as shown in Figure 3. Using the slice data $P \in R^{s \times s \times b}$ after dimensionality reduction as the input of the spectral attention branch, we first used two-layer 2D convolution to extract shallow features while adjusting the spatial size, thus reducing the number of parameters. Then, a reshape operation was performed to obtain two feature matrices, and the features were mapped to the spectral interaction space by matrix multiplication to obtain I. Next, each pixel in the region is given an attention weight by a two-layer 1×1 1D convolution, and a weight feature matrix H_{SBA} was utilized to obtain the relationship between the spectral channels. The process can be expressed as follows:

$$H_{SBA} = \varphi(F_{in})\sigma(F_{in})^T(C_{SAB} + I)Q_{SAB} \quad (9)$$

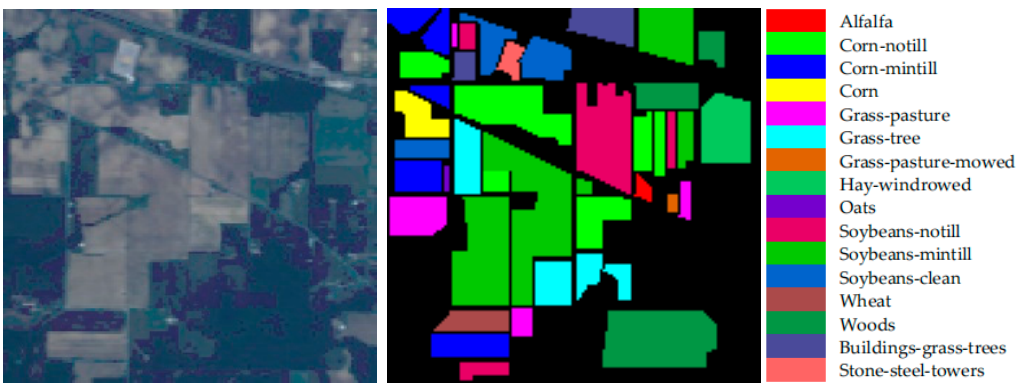
where $\varphi(\cdot)$ and $\sigma(\cdot)$ represent the reshape operation. The obtained spectral feature matrix is reshaped by adding jump connections for back projection to allow the fusion of the next two branches. Specifically, we employed a 1×1 convolution on the generated feature matrix H_{SBA}. Then, the feature matrix was converted into a vector by the reshape operation and the obtained feature vector was connected to the result of another branch. Finally, through the linear layer, the softmax function calculated the likelihood that the input fell into a certain category.

3. Results

3.1. Data Description

A total of three publicly datasets (https://github.com/gokriznastic/HybridSN (accessed on 14 March 2022) were selected for the experiments to validate the classification performance of the proposed model, namely Indian Pines (IP), Pavia University (PU) and Salinas Valley (SV).

The IP data set consists of 145×145 pixels. It includes 220 contiguous spectral bands in the wavelength range of 400–2500 nm. The spatial resolution was 20 m. It was collected by a sensor in northwestern Indiana (AVIRIS). After removing 20 absorption bands, we selected the remaining 200 bands for study. The data were divided into 16 categories containing a variety of crops, such as corn, soybeans, etc. The samples were unevenly distributed, as detailed in Table 1. We randomly selected 10% of each category as the training set. The false-color image and ground-truth map correspond to (a) and (b) in Figure 5, respectively.



(**a**) (**b**) (**c**)

Figure 5. IP dataset. (**a**) False-color image. (**b**) Ground-truth map. (**c**) Color coding for each category.

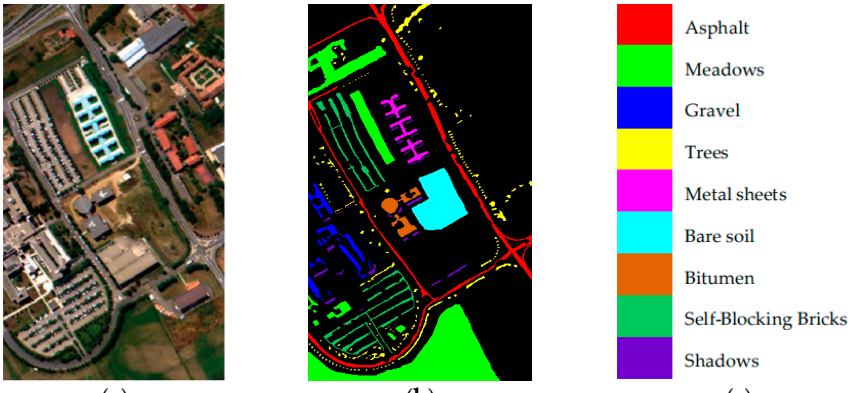
Table 1. Training and test sample division of each class in the IP dataset.

NO.	Class	Train	Test
1	Alfalfa	5	41
2	Corn—notill	143	1285
3	Corn—mintill	83	747
4	Corn	24	213
5	Grass–pasture	48	435
6	Grass-tree	73	657
7	Grass–pasture–mowed	3	25
8	Hay—windrowed	48	430
9	Oats	2	18
10	Soybeans—notill	97	875
11	Soybeans—mintill	245	2210
12	Soybeans—clean	59	534
13	Wheat	20	185
14	Woods	126	1139
15	Buildings–grass–trees	39	347
16	Stone–steel–towers	9	84
	Total	1024	9225

The PU dataset was obtained by spectral imager in the wavelength range of 430–860 nm and contains 115 bands in total. The spatial resolution was 1.3 m. It was collected by the Reflection Optical System Imaging Spectrometer (ROSIS). We removed 12 noisy bands and used the remaining 103 bands for the experiments. The dataset covered 610 × 340 pixels and contained 9 feature classes in total. Table 2 shows the details of the dataset. We used 5% of the data for training. The false-color image and ground-truth map correspond to (a) and (b) in Figure 6, respectively.

Table 2. Training and test sample division of each class in the PU dataset.

NO.	Class	Train	Test
1	Asphalt	332	6299
2	Meadows	932	17,717
3	Gravel	105	1994
4	Trees	153	2911
5	Metal sheets	67	1278
6	Bare soil	251	4778
7	Bitumen	67	1263
8	Self-Blocking bricks	184	3498
9	Shadows	47	900
	Total	2138	40,638



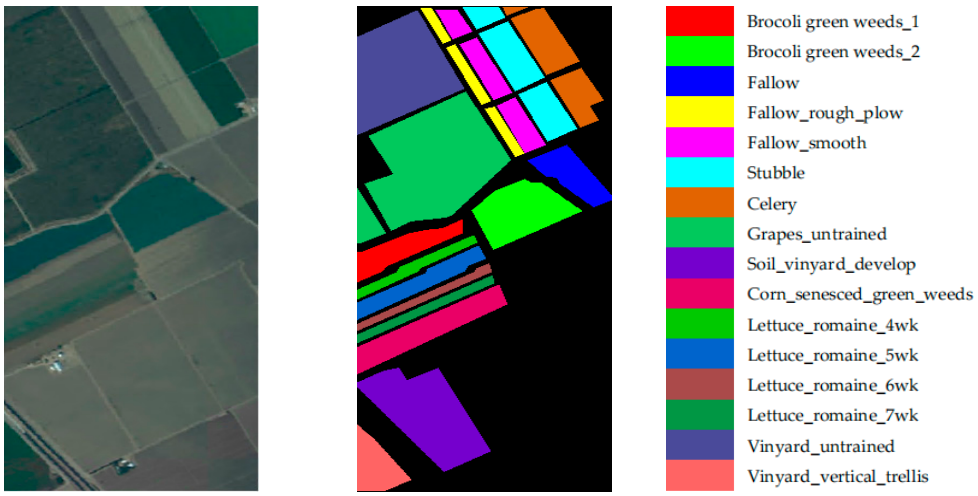
(**a**) (**b**) (**c**)

Figure 6. PU dataset. (**a**) False-color image. (**b**) Ground-truth map. (**c**) Color coding for each category.

The Salinas dataset is an image of the SV taken by the VIRIS imaging spectrometer with a spatial resolution of 3.7 m, and collected by the Airborne Visible Infrared Imaging Spectrometer (AVIRIS) in Salinas Valley, California. The original dataset consists of 224 bands; we remove the water absorbing bands and the bands affected by noise and use the remaining 204 bands for the experiment. The dataset has a size of 512 × 217 and contains 16 categories, mainly various crops, such as broccoli green weeds, celery and Lettuce_romaine, etc. Table 3 shows the details of the dataset. We use 5% of the dataset for training. The false-color image and ground-truth map correspond to (a) and (b) in Figure 7, respectively.

Table 3. Training and test sample division of each class in the SV dataset.

NO.	Class	Train	Test
1	Broccoli green weeds_1	100	1909
2	Broccoli green weeds_2	186	3540
3	Fallow	99	1877
4	Fallow_rough_plow	70	1324
5	Fallow_smooth	134	2544
6	Stubble	198	3761
7	Celery	179	3400
8	Grapes_untrained	564	10,707
9	Soil_vinyard_develop	310	5893
10	Corn_senesced_green_weeds	164	3114
11	Lettuce_romaine_4wk	53	1015
12	Lettuce_romaine_5wk	96	1831
13	Lettuce_romaine_6wk	46	870
14	Lettuce_romaine_7wk	54	1016
15	Vinyard_untrained	364	6904
16	Vinyard_vertical_trellis	90	1717
	Total	2707	51,422



(**a**) (**b**) (**c**)

Figure 7. SV dataset. (**a**) False-color image. (**b**) Ground-truth map. (**c**) Color coding for each category.

3.2. Experimental Setting

A total of six classification evaluation metrics are used in this paper: OA, AA, Kappa, training time, testing time, and accuracy of each class. In addition, we offer visualization of the classification results. To ensure fairness, we conducted ten independent experiments on each dataset, and each experiment randomly selected 10% of the data on the IP dataset, 5% of the data on the PU and SV datasets as the training set, and the rest as the test set.

The proposed method was run in the PyTorch environment. All experiments in this paper were implemented on the same computer running on an NVIDIA GeForce RTX 3060

GPU and an 11th Gen Intel(R) Core(TM) i7-11700 CPU with 16 GB of RAM. The initial learning rate was set to $le-3$, and the initial optimizer chosen was the Adam optimizer. The batch size of all three datasets was 64, and 100 training epochs were set for each dataset.

3.3. Classification Performance

We compared the method presented in this paper with several state-of-the-art HSI classification methods to evaluate its hyperspectral classification performance. These included CNN-based methods, namely 1-D CNN [33], 2-D CNN [57], 3-D CNN [35], 3D-2D hybrid CNN method HybridSN [40], the double-branch dual-attention mechanism method DBDA [47], and the transformer-based method SSFTT [54]. For the sake of fairness, we used uniform settings for all methods and conducted experiments on each of the three datasets. We took the best results from ten experiments for presentation, where the best results in each row are bolded.

On the IP dataset, the 1-D CNN method had the shortest training and testing times. Since only spectral information was used for classification, the accuracy was low. The 2-D CNN method using spatial information for classification further improved the results compared to the 1-D CNN method. In 3-D CNNs, spatial and spectral information were used jointly to further improve classification accuracy, but this took more time. HybridSN combines 3D and 2D to reduce the time cost while improving the classification accuracy. DBDA introduced the attention mechanism for classification. SSFTT combined Transformer with CNN and achieved good classification results. The classification accuracy of our method was approximately 1.2% higher than SSFTT and outperformed other methods in eleven categories, six of which had no incorrect pixels. The results of the comparison experiments on the IP dataset are shown in Table 4. To compare the classification results more visually, Figure 8 shows the ground-truth map and the classification result plots for the seven experiments. It can be observed that the 1-D CNN had a large amount of noise in the visual images and the classification accuracy was low, followed by the 2-D CNN with relatively poor classification results. The 3-D CNN, HybridSN, DBDA and SSFTT had relatively smooth visual images. Compared with other methods, the classification map produced by our method was closest to the ground-truth map, and the edges of the features were clearer.

Table 4. Classification results by different methods for the IP dataset (optimal results are bolded).

Class No.	1-D CNN	2-D CNN	3-D CNN	HybridSN	DBDA	SSFTT	Ours
1	26.83	85.37	48.78	68.29	97.56	97.56	**100**
2	71.75	88.48	92.14	**99.84**	89.96	94.24	96.73
3	53.68	79.92	98.53	95.72	96.79	97.05	**99.06**
4	52.11	76.06	87.79	92.49	99.06	97.65	**100**
5	86.90	90.34	98.39	94.71	**99.54**	99.08	97.93
6	94.06	98.48	96.35	**100**	97.11	98.32	99.85
7	44.00	84.00	**100**	60.00	92.00	88.00	**100**
8	98.84	97.91	**100**	**100**	99.77	**100**	**100**
9	33.33	83.33	72.22	**100**	**100**	72.22	**100**
10	66.40	91.89	96.80	96.00	97.03	96.91	**98.74**
11	81.44	94.21	96.38	98.69	98.64	98.73	**99.64**
12	76.40	85.96	94.19	96.07	96.63	**98.31**	96.25
13	98.38	99.46	98.38	**100**	96.22	97.28	99.46
14	95.52	95.43	99.39	98.95	95.87	**100**	**100**
15	63.98	79.54	91.07	96.54	97.98	97.11	**99.42**
16	79.76	88.10	80.95	44.05	89.29	89.16	**98.80**
OA(%)	78.38	90.10	95.75	97.27	96.49	97.69	**98.89**
AA(%)	70.21	88.65	90.71	90.08	96.47	95.10	**99.12**
Kappa × 100	75.17	89.70	95.18	96.88	95.39	97.36	**98.74**

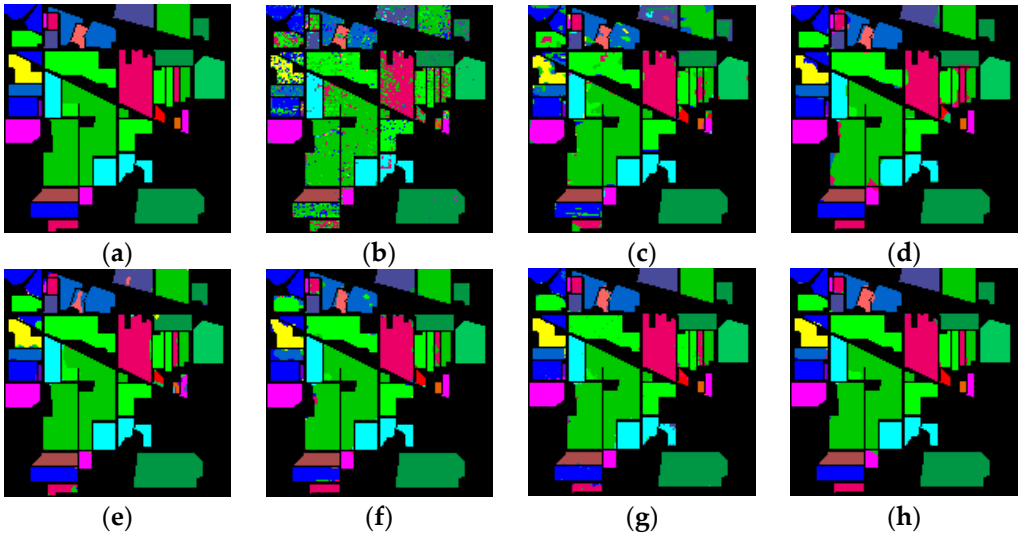


Figure 8. Classification maps of the IP dataset. (**a**) Ground-truth map. (**b**) 1-D CNN. (**c**) 2-D CNN. (**d**) 3-D CNN. (**e**) HybridSN. (**f**) DBDA. (**g**) SSFTT. (**h**) Ours.

Table 5 shows the results of the PU dataset comparison experiments. Our proposed method achieved higher accuracy on OA, AA and Kappa than any other method. We achieved higher accuracy in six categories, three of which reached 100% accuracy. Figure 9a–h show the classification maps of the ground-truth map, 1-D CNN, 2-D CNN, 3-D CNN, HybridSN, DBDA, SSFTT and our method, respectively. The classification maps of our method are closest to the ground-truth map, in which the blue lake region (category 6) is easily mixed with green pixels by the other methods; our method could better identify the category in this region. In addition, it verified the accuracy of our method's classification.

Table 5. Classification results by different methods for the PU dataset (optimal results are bolded).

Class No.	1-D CNN	2-D CNN	3-D CNN	HybridSN	DBDA	SSFTT	Ours
1	94.22	95.22	87.93	98.84	97.98	99.13	**99.17**
2	97.17	96.65	99.77	**99.98**	98.78	99.60	99.92
3	82.55	87.66	96.74	**98.99**	98.45	98.40	96.58
4	93.82	98.97	98.28	98.97	96.74	98.21	**99.66**
5	**100**	99.77	**100**	99.37	**100**	**100**	**100**
6	90.79	92.97	96.40	**100**	98.35	99.98	**100**
7	86.70	88.92	95.09	99.92	97.47	**100**	**100**
8	82.91	79.25	94.37	96.11	96.31	98.48	**98.83**
9	**100**	99.89	99.67	97.11	**100**	96.56	99.78
OA(%)	93.61	94.15	96.68	99.27	98.25	99.27	**99.54**
AA(%)	92.02	93.26	96.47	98.81	98.23	98.93	**99.33**
Kappa × 100	91.53	92.28	95.59	99.03	97.89	99.09	**99.39**

The results of the comparison experiments on the SV dataset are shown in Table 6. Our proposed method achieved higher accuracy on OA, AA and Kappa than any other methods. We achieved the best accuracy in twelve categories. To compare the classification results more intuitively, Figure 10 shows the false-color image, ground-truth map and the classification result images of the seven experiments. We can conclude from comparing the border regions in the image that our method produces the best delineation of the boundaries, thus validating the effectiveness of our method's classification.

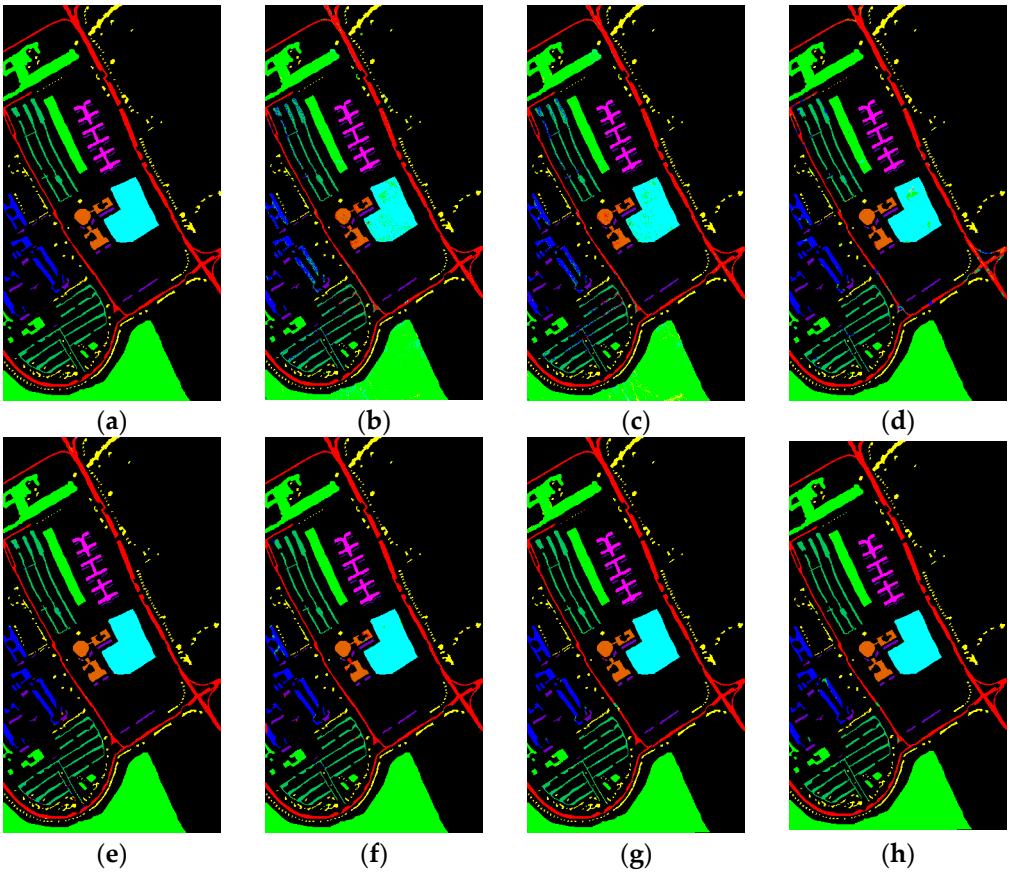


Figure 9. Classification maps of PU dataset. (**a**) Ground-truth map. (**b**) 1-D CNN. (**c**) 2-D CNN. (**d**) 3-D CNN. (**e**) HybridSN. (**f**) DBDA. (**g**) SSFTT. (**h**) Ours.

Table 6. Classification results by different methods for the SV dataset (optimal results are bolded).

Class No.	CNN1D	CNN2D	CNN3D	HybridSN	DBDA	SSFTT	Ours
1	99.79	**100**	**100**	**100**	99.90	99.84	**100**
2	99.94	99.89	**100**	**100**	99.72	98.98	**100**
3	99.73	99.84	**100**	**100**	99.68	98.61	**100**
4	99.70	99.62	98.94	**100**	98.94	98.79	**100**
5	94.58	98.82	99.88	99.49	**99.92**	99.80	99.88
6	99.02	99.92	**100**	**100**	98.78	98.70	99.89
7	99.94	99.59	**99.97**	99.70	98.74	99.91	99.47
8	87.82	94.23	99.99	99.93	**100**	99.93	99.96
9	99.81	**100**	**100**	**100**	99.54	**100**	**100**
10	97.05	98.72	**100**	**100**	99.87	99.58	**100**
11	96.65	99.01	**100**	99.70	99.80	**100**	**100**
12	**100**	**100**	**100**	**100**	**100**	97.76	**100**
13	98.97	**100**	**100**	**100**	99.89	99.43	**100**
14	89.67	99.21	99.90	**100**	99.31	99.61	99.61
15	77.87	93.14	96.48	98.19	98.60	**99.83**	99.46
16	98.95	99.65	99.71	**100**	99.83	99.30	99.59
OA(%)	93.60	97.64	99.48	99.71	99.53	99.55	**99.85**
AA(%)	96.22	98.85	99.68	99.77	99.49	99.38	**99.87**
Kappa × 100	92.87	97.99	99.42	99.67	99.51	99.50	**99.83**

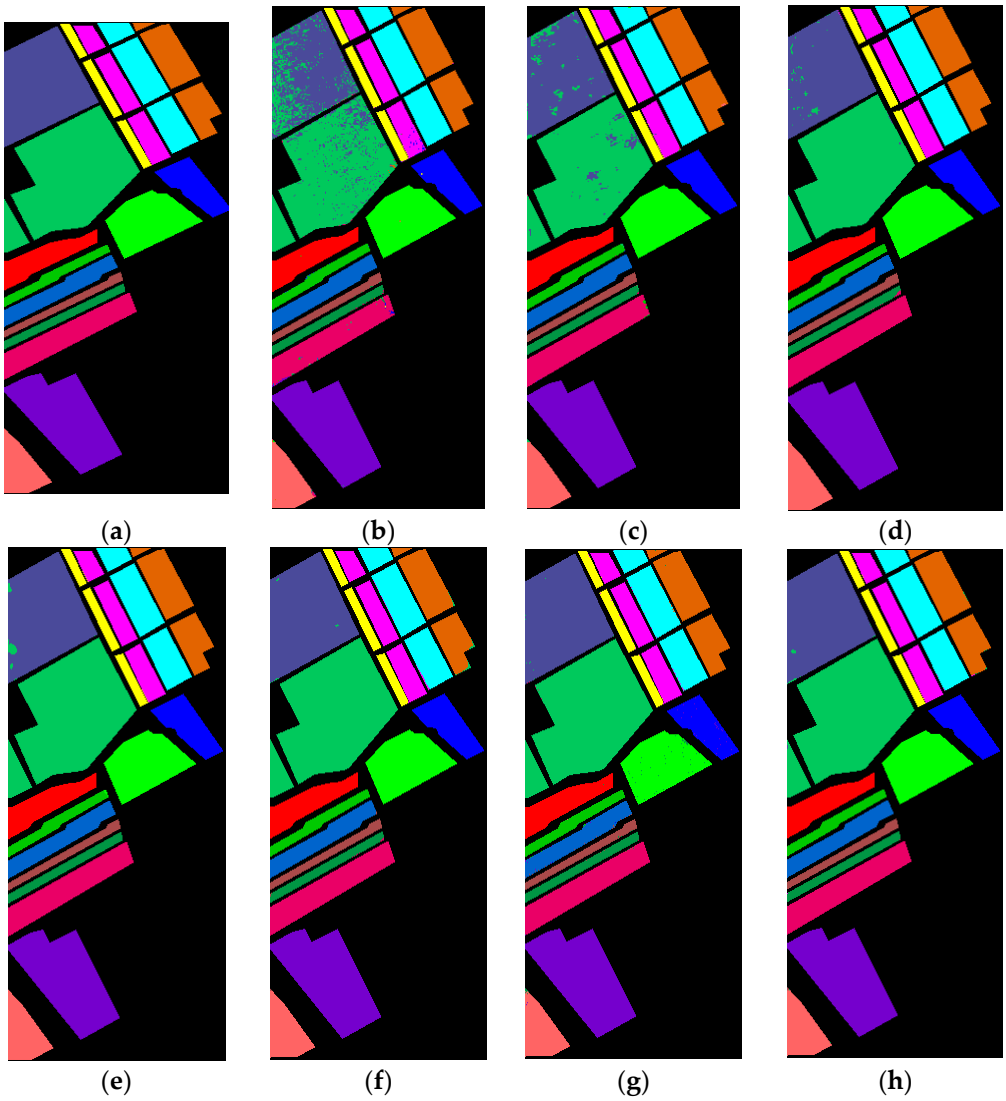


Figure 10. Classification maps of SV. (**a**) Ground-truth map. (**b**) 1-D CNN. (**c**) 2-D CNN. (**d**) 3-D CNN. (**e**) HybridSN. (**f**) DBDA. (**g**) SSFTT. (**h**) Ours.

A comparison of training and testing times of all methods tested for the three datasets is shown in Table 7. 3-D CNN requires longer for both training and testing. Our method uses 3-D CNNs to extract spatial–spectral information, which inevitably increases the time. On the IP dataset, the training time of DBAA-CNN were faster than the other methods besides 1-D CNN. The reduction in time spent by our method is also significant on the PU and SV datasets. In particular, our method's training time on the SV dataset was shorter than that of the other compared methods, which further indicates that our method improves classification accuracy and efficiency while reducing time.

3.4. Parameter Analysis

HSI classification is to determine the class of the central pixel of the cut block. The larger the cut block is, the more neighboring pixels it contains, which may help to classify the central pixel, but also inevitably causes an increase in computational cost. Thus, we analyzed the size of the window on three datasets. Table 8 and Figure 11 show the impact of the window size of the input data on the classification results. We can observe from the table that as the window size increases, the complexity of the computation increases and the training time increases, too. Moreover, the average accuracy rises and then falls

as the window size increases. Considering the two factors of computational volume and classification accuracy, the window size used for three datasets was 13 × 13.

Table 7. Training and testing time of all methods for the three datasets (optimal results are bolded).

Methods	IP		PU		SV	
	Train(s)	Test(s)	Train(s)	Test(s)	Train(s)	Test(s)
1-D CNN	**39.73**	**0.36**	141.32	**1.26**	187.38	**1.64**
2-D CNN	64.77	0.86	220.86	2.23	207.11	1.87
3-D CNN	126.96	1.36	325.96	3.70	296.71	6.52
HybridSN	80.34	1.58	**75.59**	2.50	94.02	3.21
DBDA	62.03	2.54	196.76	14.20	215.26	15.35
SSFTT	57.13	1.90	237.53	8.03	300.38	10.24
Ours	52.19	1.20	105.71	4.64	**88.05**	5.32

Table 8. Performance impact of different window sizes.

Window Sizes	OA			Testing Time(s)		
	IP	PU	SV	IP	PU	SV
9 × 9	98.39	99.19	99.42	0.66	3.01	3.76
11 × 11	98.52	99.44	99.78	0.91	3.79	5.28
13 × 13	98.89	99.54	99.85	1.20	4.64	5.32
15 × 15	98.89	99.45	99.92	1.57	5.64	7.36
17 × 17	98.57	99.47	99.95	1.68	6.36	10.47

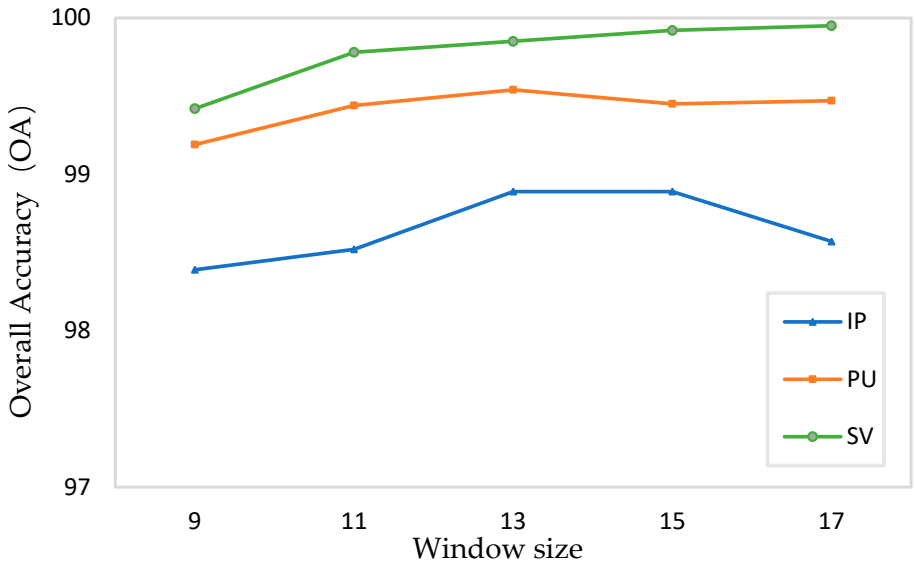


Figure 11. The effect of different window sizes on the OA of DBAA-CNN.

We also chose a different number of bands to test the effects on the classification performance. In HSI classification, the number of bands determines how much spectral information is used by the network. The fewer the bands, the less spectral information is used and the shorter the time spent, and vice versa. The effect of the number of bands on the classification performance is shown in Figure 12. It can be observed that OA increases with the number of bands. In summary, the number of bands for three datasets was chosen to be 80, considering stability and generalization.

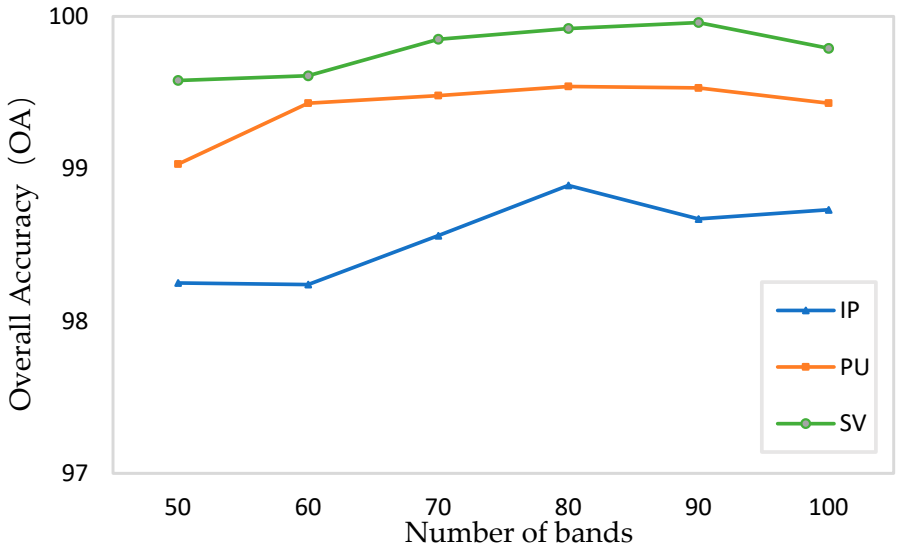


Figure 12. The effect of different numbers of bands on the OA of DBAA-CNN.

3.5. Ablation Experiments

We evaluate the efficacy of each module in a comprehensive manner by conducting ablation experiments on the IP dataset. Table 9 displays the experimental results for different module settings. The study of the ablation experiment data indicates that the three modules cooperate to produce better classification outcomes, further demonstrating the efficacy of our suggested paradigm.

Table 9. Effect of different modules in the DBAA-CNN on the IP data set. (optimal results are bolded).

Cases	3D Conv	PSA Module	Spectral Attention Branch	OA (%)	AA (%)	Kappa (%)
1	×	√	√	96.41	94.66	95.90
2	√	×	√	95.70	93.42	95.38
3	√	√	×	97.96	94.42	95.10
4	√	√	√	**98.89**	**99.12**	**98.74**

The "×" in Table 9 indicates that the module is not included.

4. Discussion

Based on the experimental results, the DBAA-CNN performs significantly better than other classification methods. The 1-D CNN has the worst classification results on the three datasets because the method loses spatial information due to its one-dimensional input data. The 2-D CNN method takes into account the spatial information; thus, the OA was improved compared with the 1-D CNN method. The 3-D CNN method extracts the features directly from the 3D data, which better preserves the original features of the data. The HybridSN and DBDA methods combine 2-D and 3-D CNN to extract spatial and spectral features, integrate them and feed them into the classifier. The SSFTT approach adds the transformer to extract spectral features. The DBDA approach adds the channel and spectral attention blocks to improve the classification effect but it consumed more time on the PU and SV datasets. Our method had the best classification results on the three datasets. The DBAA-CNN combines advantages of the attention mechanism and also reduces time consumption, which reduces the training and testing time while improving the classification results.

Additionally, the performance of classification was evaluated with different window sizes and the amount of bands. The final window size of 13 × 13 was chosen by considering the OA and testing time. In order to retain more spectral information, we chose a different number of bands for the experiments, OA increases with the number of bands, but more

bands inevitably contain redundant information, which causes the OA to rise and then fall, so the number of bands is finally chosen to be 80. Table 8 shows that our method's training and testing times on the three datasets are advantageous, which indicates that we have improved the efficiency of classification.

As shown in Table 8, the training and testing times of our method on three datasets is advantageous, which indicates that we have improved the efficiency of classification.

5. Conclusions

We presented a novel DBAA-CNN classification method for HSI in this paper. A spatial–spectral branch and a spectral attention branch make up the method. The spatial-spectral branch combines 3-D CNN with multiscale squeeze-and-excitation pyramid attention. 3-D convolutional layers are used to obtain shallow spatial–spectral features. The multiscale pyramid module is used to further mine the multiscale information of HSI, and then, integrate the multiscale spatial information with cross-channel attention. The spectral attention branch maps original features to the spectral interaction space for feature representation and learning. In order to generate spatial–spectral features for classification, the features of two branches are finally combined. This enhances the ability of the feature map to extract valid information by utilizing the attention mechanism. Experiments and analysis of the three datasets demonstrate that the method effectively enhances classification performance and reduces time consumption. In future work, we will consider combining graph convolution networks (GCN) to jointly extract spatial and spectral features, thus further enhancing the efficiency and accuracy of classification.

Author Contributions: Conceptualization, M.J.; Methodology, W.H.; Validation, Z.Z.; Supervision, L.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Scientific and technological key project in Henan Province, grant number 212102210102 and 212102210105.

Data Availability Statement: The data presented in this study are available in article.

Acknowledgments: The authors would like to thank the editors and reviewers for their advice.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional Neural Network
HSI	Hyperspectral Image
PSA	Pyramid Squeeze-And-Excitation Attention
KNN	K-Nearest Neighbor
SVM	Support Vector Machines
SAE	Stacked Auto Encoder
DBN	Deep Belief Networks
RNN	Recurrent Neural Networks
NLP	Natural Language Processing
SE	Squeeze-and-Excitation
GAP	Global Average Pooling
FC	Fully Connected
SAC	Squeeze And Concat
IP	Indian Pines
PU	Pavia University
SV	Salinas Valley
OA	Overall Accuracy
AA	Average Accuracy
Kappa	Kappa Coefficient

References

1. Huang, W.; Li, G.; Chen, Q.; Ju, M.; Qu, J. CF2PN: A Cross-Scale Feature Fusion Pyramid Network Based Remote Sensing Target Detection. *Remote Sens.* **2021**, *13*, 847. [CrossRef]
2. Huang, W.; Li, G.; Jin, B.; Chen, Q.; Yin, J.; Huang, L. Scenario Context-Aware-Based Bidirectional Feature Pyramid Network for Remote Sensing Target Detection. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 1–5. [CrossRef]
3. Sabbah, S.; Rusch, P.; Eichmann, J.; Gerhard, J.-H.; Harig, R. Remote Sensing of Gases by Hyperspectral Imaging: Results of Field Measurements. In Proceedings of the Electro-Optical Remote Sensing, Photonic Technologies, and Applications VI, Edinburgh, UK, 24–27 September 2012; Kamerman, G.W., Steinvall, O., Lewis, K.L., Hollins, R.C., Merlet, T.J., Gruneisen, M.T., Dusek, M., Rarity, J.G., Bishop, G.J., Gonglewski, J., Eds.; SPIE: Bellingham, WA, USA, 2012; Volume 8542, p. 854227.
4. Gevaert, C.M.; Suomalainen, J.; Tang, J.; Kooistra, L. Generation of Spectral–Temporal Response Surfaces by Combining Multispectral Satellite and Hyperspectral UAV Imagery for Precision Agriculture Applications. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2015**, *8*, 3140–3146. [CrossRef]
5. Shimoni, M.; Haelterman, R.; Perneel, C. Hypersectral Imaging for Military and Security Applications: Combining Myriad Processing and Sensing Techniques. *IEEE Geosci. Remote Sens. Mag.* **2019**, *7*, 101–117. [CrossRef]
6. Ardouin, J.-P.; Levesque, J.; Rea, T.A. A Demonstration of Hyperspectral Image Exploitation for Military Applications. In Proceedings of the 2007 10th International Conference on Information Fusion, Quebec, QC, Canada, 9–12 July 2007; pp. 1–8.
7. Fan, J.; Zhou, N.; Peng, J.; Gao, L. Hierarchical Learning of Tree Classifiers for Large-Scale Plant Species Identification. *IEEE Trans. Image Process.* **2015**, *24*, 4172–4184. [PubMed]
8. Hsieh, T.-H.; Kiang, J.-F. Comparison of CNN Algorithms on Hyperspectral Image Classification in Agricultural Lands. *Sensors* **2020**, *20*, 1734. [CrossRef] [PubMed]
9. Li, S.; Song, W.; Fang, L.; Chen, Y.; Ghamisi, P.; Benediktsson, J.A. Deep Learning for Hyperspectral Image Classification: An Overview. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 6690–6709. [CrossRef]
10. Dong, Y.; Liang, T.; Zhang, Y.; Du, B. Spectral–Spatial Weighted Kernel Manifold Embedded Distribution Alignment for Remote Sensing Image Classification. *IEEE Trans. Cybern.* **2021**, *51*, 3185–3197. [CrossRef]
11. Yue, J.; Fang, L.; Rahmani, H.; Ghamisi, P. Self-Supervised Learning With Adaptive Distillation for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–13. [CrossRef]
12. Cheng, G.; Li, Z.; Han, J.; Yao, X.; Guo, L. Exploring Hierarchical Convolutional Features for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 6712–6722. [CrossRef]
13. Bioucas-Dias, J.M.; Plaza, A.; Camps-Valls, G.; Scheunders, P.; Nasrabadi, N.; Chanussot, J. Hyperspectral Remote Sensing Data Analysis and Future Challenges. *IEEE Geosci. Remote Sens. Mag.* **2013**, *1*, 6–36. [CrossRef]
14. Awad, M.M. Cooperative Evolutionary Classification Algorithm for Hyperspectral Images. *J. Appl. Remote Sens.* **2020**, *14*, 016509. [CrossRef]
15. Li, J.; Khodadadzadeh, M.; Plaza, A.; Jia, X.; Bioucas-Dias, J.M. A Discontinuity Preserving Relaxation Scheme for Spectral–Spatial Hyperspectral Image Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2016**, *9*, 625–639. [CrossRef]
16. Wambugu, N.; Chen, Y.; Xiao, Z.; Tan, K.; Wei, M.; Liu, X.; Li, J. Hyperspectral Image Classification on Insufficient-Sample and Feature Learning Using Deep Neural Networks: A Review. *Int. J. Appl. Earth Obs. Geoinf.* **2021**, *105*, 102603. [CrossRef]
17. Fabiyi, S.D.; Murray, P.; Zabalza, J.; Ren, J. Folded LDA: Extending the Linear Discriminant Analysis Algorithm for Feature Extraction and Data Reduction in Hyperspectral Remote Sensing. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 12312–12331. [CrossRef]
18. Duan, Y.; Huang, H.; Tang, Y. Local Constraint-Based Sparse Manifold Hypergraph Learning for Dimensionality Reduction of Hyperspectral Image. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 613–628. [CrossRef]
19. Luo, F.; Zhang, L.; Zhou, X.; Guo, T.; Cheng, Y.; Yin, T. Sparse-Adaptive Hypergraph Discriminant Analysis for Hyperspectral Image Classification. *IEEE Geosci. Remote Sens. Lett.* **2020**, *17*, 1082–1086. [CrossRef]
20. Duan, Y.; Huang, H.; Li, Z.; Tang, Y. Local Manifold-Based Sparse Discriminant Learning for Feature Extraction of Hyperspectral Image. *IEEE Trans. Cybern.* **2021**, *51*, 4021–4034. [CrossRef]
21. Ma, L.; Crawford, M.M.; Tian, J. Local Manifold Learning-Based k -Nearest-Neighbor for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 4099–4109. [CrossRef]
22. Fauvel, M.; Benediktsson, J.A.; Chanussot, J.; Sveinsson, J.R. Spectral and Spatial Classification of Hyperspectral Data Using SVMs and Morphological Profiles. *IEEE Trans. Geosci. Remote Sens.* **2008**, *46*, 3804–3814. [CrossRef]
23. Melgani, F.; Bruzzone, L. Classification of Hyperspectral Remote Sensing Images with Support Vector Machines. *IEEE Trans. Geosci. Remote Sens.* **2004**, *42*, 1778–1790. [CrossRef]
24. Huang, W.; Huang, Y.; Wang, H.; Liu, Y.; Shim, H.J. Local Binary Patterns and Superpixel-Based Multiple Kernels for Hyperspectral Image Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 4550–4563. [CrossRef]
25. Paoletti, M.E.; Haut, J.M.; Pereira, N.S.; Plaza, J.; Plaza, A. Ghostnet for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 10378–10393. [CrossRef]
26. LeCun, Y.; Bengio, Y.; Hinton, G. Deep Learning. *Nature* **2015**, *521*, 436–444. [CrossRef]
27. Zhang, K.; Zuo, W.; Zhang, L. FFDNet: Toward a Fast and Flexible Solution for CNN-Based Image Denoising. *IEEE Trans. Image Process.* **2018**, *27*, 4608–4622. [CrossRef]

28. Remote Sensing | Free Full-Text | Combing Triple-Part Features of Convolutional Neural Networks for Scene Classification in Remote Sensing. Available online: https://www.mdpi.com/2072-4292/11/14/1687 (accessed on 12 July 2022).
29. Chen, Y.; Lin, Z.; Zhao, X.; Wang, G.; Gu, Y. Deep Learning-Based Classification of Hyperspectral Data. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 2094–2107. [CrossRef]
30. Chen, Y.; Zhao, X.; Jia, X. Spectral–Spatial Classification of Hyperspectral Data Based on Deep Belief Network. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2015**, *8*, 2381–2392. [CrossRef]
31. Mou, L.; Ghamisi, P.; Zhu, X.X. Deep Recurrent Neural Networks for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3639–3655. [CrossRef]
32. Hang, R.; Li, Z.; Ghamisi, P.; Hong, D.; Xia, G.; Liu, Q. Classification of Hyperspectral and LiDAR Data Using Coupled CNNs. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 4939–4950. [CrossRef]
33. Hu, W.; Huang, Y.; Wei, L.; Zhang, F.; Li, H. Deep Convolutional Neural Networks for Hyperspectral Image Classification. *J. Sens.* **2015**, *2015*, 258619. [CrossRef]
34. Makantasis, K.; Karantzalos, K.; Doulamis, A.; Doulamis, N. Deep Supervised Learning for Hyperspectral Data Classification through Convolutional Neural Networks. In Proceedings of the 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Milan, Italy, 26–31 July 2015; pp. 4959–4962.
35. Chen, Y.; Jiang, H.; Li, C.; Jia, X.; Ghamisi, P. Deep Feature Extraction and Classification of Hyperspectral Images Based on Convolutional Neural Networks. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 6232–6251. [CrossRef]
36. Ben Hamida, A.; Benoit, A.; Lambert, P.; Ben Amar, C. 3-D Deep Learning Approach for Remote Sensing Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 4420–4434. [CrossRef]
37. Li, Y.; Zhang, H.; Shen, Q. Spectral-Spatial Classification of Hyperspectral Imagery with 3D Convolutional Neural Network. *Remote Sens.* **2017**, *9*, 67. [CrossRef]
38. Zhong, Z.; Li, J.; Luo, Z.; Chapman, M. Spectral–Spatial Residual Network for Hyperspectral Image Classification: A 3-D Deep Learning Framework. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 847–858. [CrossRef]
39. Paoletti, M.E.; Haut, J.M.; Fernandez-Beltran, R.; Plaza, J.; Plaza, A.J.; Pla, F. Deep Pyramidal Residual Networks for Spectral–Spatial Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 740–754. [CrossRef]
40. Roy, S.K.; Krishna, G.; Dubey, S.R.; Chaudhuri, B.B. HybridSN: Exploring 3-D–2-D CNN Feature Hierarchy for Hyperspectral Image Classification. *IEEE Geosci. Remote Sens. Lett.* **2020**, *17*, 277–281. [CrossRef]
41. Xu, X.; Li, W.; Ran, Q.; Du, Q.; Gao, L.; Zhang, B. Multisource Remote Sensing Data Classification Based on Convolutional Neural Network. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 937–949. [CrossRef]
42. Wang, F.; Jiang, M.; Qian, C.; Yang, S.; Li, C.; Zhang, H.; Wang, X.; Tang, X. Residual Attention Network for Image Classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 3156–3164.
43. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Springer International Publishing Ag: Cham, Switzerland, 2018; Volume 11211, pp. 3–19.
44. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E. Squeeze-and-Excitation Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 2011–2023. [CrossRef] [PubMed]
45. Haut, J.M.; Paoletti, M.E.; Plaza, J.; Plaza, A.; Li, J. Visual Attention-Driven Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 8065–8080. [CrossRef]
46. Sun, H.; Zheng, X.; Lu, X.; Wu, S. Spectral–Spatial Attention Network for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 3232–3245. [CrossRef]
47. Li, R.; Zheng, S.; Duan, C.; Yang, Y.; Wang, X. Classification of Hyperspectral Image Based on Double-Branch Dual-Attention Mechanism Network. *Remote Sens.* **2020**, *12*, 582. [CrossRef]
48. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30.
49. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *J. Mach. Learn. Res.* **2020**, *21*, 1–67.
50. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image Is Worth 16×16 Words: Transformers for Image Recognition at Scale. *arXiv* **2021**, arXiv:2010.11929.
51. Hu, X.; Li, T.; Zhou, T.; Liu, Y.; Peng, Y. Contrastive Learning Based on Transformer for Hyperspectral Image Classification. *Appl. Sci.* **2021**, *11*, 8670. [CrossRef]
52. Remote Sensing | Free Full-Text | Improved Transformer Net for Hyperspectral Image Classification. Available online: https://www.mdpi.com/2072-4292/13/11/2216 (accessed on 1 September 2022).
53. Hong, D.; Han, Z.; Yao, J.; Gao, L.; Zhang, B.; Plaza, A.; Chanussot, J. SpectralFormer: Rethinking Hyperspectral Image Classification With Transformers. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–15. [CrossRef]
54. Sun, L.; Zhao, G.; Zheng, Y.; Wu, Z. Spectral–Spatial Feature Tokenization Transformer for Hyperspectral Image Classification. IEEE Trans. Geosci. *Remote Sens.* **2022**, *60*, 1–14.
55. Licciardi, G.; Marpu, P.R.; Chanussot, J.; Benediktsson, J.A. Linear Versus Nonlinear PCA for the Classification of Hyperspectral Data Based on the Extended Morphological Profiles. IEEE Geosci. *Remote Sens. Lett.* **2012**, *9*, 447–451. [CrossRef]

56. Zhang, H.; Zu, K.; Lu, J.; Zou, Y.; Meng, D. EPSANet: An Efficient Pyramid Squeeze Attention Block on Convolutional Neural Network. In Proceedings of the Asian Conference on Computer Vision (ACCV), Macau SAR, China, 4–8 December 2022.
57. Zhao, W.; Du, S. Spectral–Spatial Feature Extraction for Hyperspectral Image Classification: A Dimension Reduction and Deep Learning Approach. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 4544–4554. [CrossRef]