remote sensing

MDPI

Article

FusionNet: A Convolution–Transformer Fusion Network for Hyperspectral Image Classification

Liming Yang [1,†], Yihang Yang [1,†], Jinghui Yang [1,*], Ningyuan Zhao [2], Ling Wu [1], Liguo Wang [3] and Tianrui Wang [1]

[1] School of Information Engineering, China University of Geosciences (Beijing), Beijing 100083, China
[2] School of Automation, Nanjing University of Science and Technology (NJUST), Nanjing 210094, China
[3] College of Information and Communication Engineering, Dalian Minzu University, Dalian 116600, China
* Correspondence: yangjh@cugb.edu.cn
† These authors contributed equally to this work.

Abstract: In recent years, deep-learning-based hyperspectral image (HSI) classification networks have become one of the most dominant implementations in HSI classification tasks. Among these networks, convolutional neural networks (CNNs) and attention-based networks have prevailed over other HSI classification networks. While convolutional neural networks with perceptual fields can effectively extract local features in the spatial dimension of HSI, they are poor at capturing the global and sequential features of spectral–spatial information; networks based on attention mechanisms, for example, Transformer, usually have better ability to capture global features, but are relatively weak in discriminating local features. This paper proposes a fusion network of convolution and Transformer for HSI classification, known as FusionNet, in which convolution and Transformer are fused in both serial and parallel mechanisms to achieve the full utilization of HSI features. Experimental results demonstrate that the proposed network has superior classification results compared to previous similar networks, and performs relatively well even on a small amount of training data.

Keywords: convolutional neural network; self-attention; deep learning; spectral–spatial features; hyperspectral image (HSI) classification

Citation: Yang, L.; Yang, Y.; Yang, J.; Zhao, N.; Wu, L.; Wang, L.; Wang, T. FusionNet: A Convolution–Transformer Fusion Network for Hyperspectral Image Classification. *Remote Sens.* **2022**, *14*, 4066. https://doi.org/10.3390/rs14164066

Academic Editors: Salah Bourennane and Xian Sun

Received: 23 May 2022
Accepted: 17 August 2022
Published: 19 August 2022

1. Introduction

Hyperspectral images (HSIs) combine imaging technology with spectral detection technology, in which each pixel contains dozens or even hundreds of spectral bands, recording the spectral profile of the corresponding feature [1,2]. Similar to ordinary images, hyperspectral images also possess spatial information. However, due to the differences in the spectral profiles of different ground covers, hyperspectral images possess richer spectral information compared with normal images, which helps to determine the category of ground cover more accurately. Currently, hyperspectral observation techniques are widely used in many fields, such as water monitoring [3,4], agricultural monitoring [5], and resource exploration [6]. Common tasks of hyperspectral images include unmixing [7], detection [8], and classification [9].

During the past decades, a large number of HSI classification methods have been proposed and dedicated to obtaining accurate classification results. Traditional HSI classification methods, such as support vector machine (SVM) [10], polynomial regression (MLR) [11], random forest [12], and sparse representation [13], are often used in HSI classification tasks. To further improve the classification performance, researchers have conducted extensive exploration into spectral–spatial classification methods, such as using partitional clustering [14], joint sparse representation [15], etc. Spectral–spatial classification methods combine spectral information and spatial contextual information, which can effectively improve the results of HSI classification compared with the classification methods based on spectra only [16]. Although traditional HSI classification methods are capable of

achieving good performance, they are usually highly reliant on manual design with poor generalization ability, which limits their performance in difficult scenarios [16,17].

Recently, deep-learning-based methods have gained attention due to their powerful feature extraction and representation capabilities, and have been widely used in computer vision [18], natural language processing [19], remote sensing image processing [20], and other fields. In the field of HSI classification, deep-learning-based models have also been applied extensively and proved to possess excellent learning ability and feature representation [21]. Chen et al. [22] first introduced deep learning methods to hyperspectral image classification, and proposed an automatic stacking encoder-based HSI classification model. In the following years, many backbone networks in the area of deep learning were successfully applied to HSI classification tasks. For example, Hu et al. [23] designed a deep convolutional neural network (CNN) for hyperspectral image classification; Hang et al. [24] proposed a cascaded recurrent neural network (RNN) model using gated recurrent units, aiming to eliminate redundant information between adjacent spectral bands and learning complementary information between nonadjacent bands; in [25], generative adversarial networks (GANs) were introduced into the HSI classification task, providing a new perspective; based on graph convolutional networks (GCNs) to extract features, a network called nonlocal GCN was proposed in [26]; in [27], a novel end-to-end, pixel-to-pixel, fully convolutional network (FCN) was proposed, consisting of a three-dimensional fully convolutional network and a convolutional spatial propagation network; and finally, a dual multiheaded contextual self-attention network for HSI classification was proposed in [28], which used two submodules for the extraction of spectral–spatial information.

CNN-based classification models have dominated the current deep-learning-based HSI classification models. Benefiting from local connectivity and weight-sharing mechanisms, CNNs can effectively capture spatial structure information and local contextual information while reducing the number of parameters to achieve good classification results. A spectral–spatial CNN classification method was proposed earlier in [29], and achieved better results than traditional methods; a network named context–depth CNN was proposed in [30] to optimize local contextual interactions by exploiting the spectral–spatial information of neighboring pixels; a spectral–spatial 3-D–2-D CNN classification model was introduced in [31], which shows the excellent potential of hybrid networks by mixing two-dimensional convolution and three-dimensional convolution for deep extraction of spectral–spatial features. Recently, more novel CNN-based classification models have been proposed; for example, the consolidated convolutional neural network (C-CNN) [32] and the lightweight spectral–spatial convolution module (LS2CM) [33], etc.

Although convolutional neural networks have been proven to have good capability in HSI classification tasks, the limitations of the convolutional operation itself also hinder the further improvement of its performance. Firstly, in the spectral dimension, CNNs have difficulty in capturing the global properties of sequences well, and perform poorly when facing the extraction of long-range dependencies. Benefiting from the convolution operation in CNN, CNN can achieve powerful spatial feature extraction capability in the spatial dimension by collecting local features hierarchically. However, HSI data are more similar to a sequence in the spectral dimension, and usually have a large spectral dimension. The use of CNN may introduce a partial loss of information [34], which causes difficulties in feature extraction. At the same time, this makes it difficult to fully utilize the dependency information among the spectral channels that are far apart, especially when there are many types of land cover and the similarity of spectral features is significant [35]; this may become a bottleneck that prevents CNNs from achieving more refined land cover classification. Second, CNN may not be the best choice for global contextual relationship establishment in the spatial dimension. When performing classification data selection, a common method of selection is to combine the pixel to be classified and its nearby pixel information as the input of the classification model. The nearby pixels contain rich spatial contextual information, which helps to achieve better classification results. Compared to convolutional neural networks, which are more adept at local feature extraction, the

self-attention mechanism may be a better choice since it is not limited by distance and is inherently better at capturing global contextual relationships. In addition, establishing such global contextual relationships will help classification models to obtain better robustness and be less susceptible to perturbations [36].

Facing the difficulties mentioned above, one possible solution is to design HSI classification models based on attention mechanisms. The attention mechanism is capable of extracting the features that are more important for the classification task, while ignoring less-valuable features, to achieve effective information utilization. Fang et al. proposed a 3D CNN network using the attention mechanism in the spectral channel [37], but did not use the attention mechanism for spatial information. Therefore, Li and Zheng et al. [9] constructed a parallel network using the attention mechanism for spatial and spectral features separately, and achieved better results. Later, a network called Transformer was proposed, which is based on the self-attention mechanism, and exhibits good global feature extraction and long-range relationship capturing capability [38]. Although the self-attention mechanism can extract important spatial and spectral features, it is impossible to obtain the location information between features with the self-attention mechanism only; thus, Transformer adds positional encoding so that the network that uses the attention mechanism can learn the positional information of features. The vision transformer [39] model was the first to propose the use of Transformer in computer vision application, and achieved decent results. Since then, a large number of Transformer-based networks have emerged in the field of computer vision, as well as in HSI classification. For example, a backbone network called SpectralFormer was proposed in [28] to view HSI classification from the perspective of sequences; [40] replaced the traditional convolutional layer with Transformer, and investigated Transformer classification results along spatial and spectral dimensions, proposing a spectral–spatial HSI classification model called DSS-TRM; Sun et al. [41] proposed a spectral–spatial feature tokenization transformer (SSFTT) model to overcome the difficulty of extracting deep semantic features by previous methods, thus making full use of the deep semantic properties of spectral–spatial features; to alleviate the problem of gradient vanishing, a novel Transformer called DenseTransformer was proposed by [34], and applied in their spectral–spatial HSI classification framework. Attention-based and Transformer-based HSI classification models are gaining more attention from researchers, but such models are not a perfect choice; for example, although Transformer has good global feature extraction capabilities and long-range contextual relationship representation, it is usually weak in local fine-grained feature extraction [42] and underutilizes spatial information [41], which is usually what convolution excels at.

Reviewing the CNN-based and attention-based models, it is not difficult to find that their advantages and disadvantages are complementary. By jointly employing CNNs and attention mechanisms, it will be possible to achieve sufficient extraction of local features while further optimizing the full utilization of global contextual relationships. Related explorations have been made in the field of deep learning; for example, Gulati et al. [42] proposed a Transformer–CNN fusion model for automatic speech recognition (ASR), expecting a win–win situation by representing local features and global contextual connections of audio sequences through convolution and Transformer; Peng and Ye [43] et al. also proposed a network with a two-branch CNN and Transformer, thus synthesizing the representation capability of enhanced learning.

Inspired by the above works, this paper proposes a convolution–Transformer fusion network for HSI classification, known as FusionNet. FusionNet contains two branches, one of which is the Local Branch, consisting of residual-connected convolutional modules and mainly focusing on extracting local information in spatial and spectral dimensions. The other branch is the Global Branch, which is serially connected by Transformer and convolutional modules, and is mainly responsible for extracting global information of spatial and spectral dimensions. The purpose of introducing convolutional modules into this branch is to further enhance the extraction capability of Transformer for local features. In addition, a fusion mechanism is added between the Global Branch and Local branch

to enable the communication of local and global information. Hyperspectral images are different from ordinary images, which express a theme as a whole image and possess local features at the spatial–channel level that are more important than hyperspectral images. Thus, CNN is added to Transformer to enhance the ability of the Global Branch to extract local information. The whole network is structured so that the Global Branch and Local Branch are parallel as a whole, while the convolution module and Transformer are serially localized inside the Global Branch. The main contributions of this article are listed as follows:

- **This paper proposes a novel convolution–Transformer fusion network for HSI classification.** Considering the strength of convolution in local feature extraction, and the capability of Transformer in long-range contextual relationship extraction, this paper proposes a fusion solution for HSI classification. Local fine-grained features can be well extracted by the convolution module, while global features in the spectral–spatial dimension and long-distance relationships in the spectral dimension can be better extracted by the Transformer module. The fusion of convolution and Transformer successfully enables them to complement each other to achieve better HSI classification results.
- **In FusionNet, a hybrid convolution–Transformer fusion pattern based on serial arrangement and parallel arrangement for a double-branch fusion mechanism was proposed for HSI classification.** The Local Branch provides local feature extraction based on convolution, while the Global Branch provides global feature extraction based on convolution–Transformer serial arrangement module. The Local Branch and Global Branch are fused through a double-branch fusion mechanism in a parallel arrangement module. Furthermore, the hybrid fusion pattern enables the fusion and exchange of information between two branches to achieve effective extraction of local and global features.
- FusionNet achieves competitive results on the three most commonly used datasets. This suggests that the fusion of convolution and Transformer may be a better solution for HSI classification, and provides a reference for further research in the future.

The remainder of the article is organized as follows: Section 2 presents the materials and methods of FusionNet, including the related works and detailed structure of FusionNet. The comparison results of the network on several datasets are given in Section 3. Section 4 presents a discussion of the experimental results and design of FusionNet to further investigate the effectiveness of the structure. Finally, Section 5 summarizes the entire work and the future directions.

2. Materials and Methods

This Section contains a detailed description of the related work and the structure of FusionNet. First, from Section 2.1 to Section 2.4, the classical structures in FusionNet will be introduced, including the attention mechanism, LayerNorm, residual connectivity, etc. Subsequently, Section 2.5 will provide a detailed description of the proposed FusionNet.

2.1. Attention Mechanism

The attention mechanism draws on the human attention mechanism to select the information that is more important to the task from numerous information sources. Three matrices are obtained by three linear computations of the inputs; namely key, query, and value. Key is the key-value matrix. Query refers to the query matrix, and value is the weight matrix. By calculating the similarity of each pair of query and key, the value corresponding to each key is obtained. Finally, the value is weighted and summed to obtain the final output. In Equation (1), let key be K, query be Q, and value be V.

$$Output(Q, K, V) = softmax(QK^T)V \quad (1)$$

Q and V are calculated as follows, where W_Q, W_K, and W_V are the parameter matrices to be learned.

$$Q = Input \times W_Q \tag{2}$$

$$K = Input \times W_K \tag{3}$$

$$V = Input \times W_V \tag{4}$$

The self-attention mechanism is one of the attention mechanisms, which reduces the dependence on external data and is better at capturing the internal relevance of data or features. The calculation process of the self-attention mechanism is shown in Figure 1. The scaled dot-product attention mechanism is used in Transformer, which differs from normal self-attention mechanisms in terms of scaling the results of the dot-products of Q and K. As shown below, d_k is the input dimension of the key.

$$Output(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{5}$$

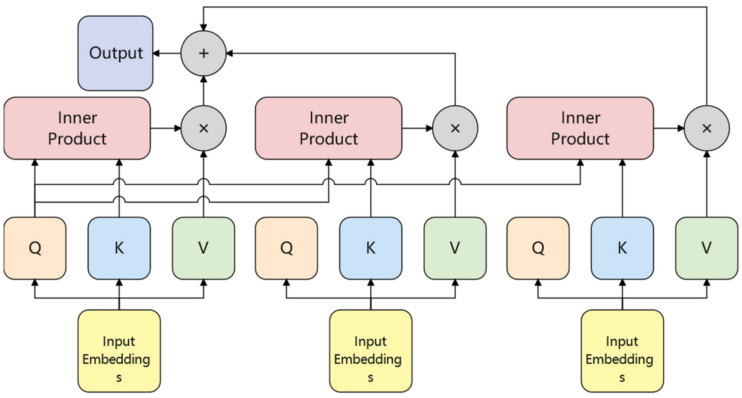


Figure 1. Attention mechanism.

The multihead self-attention mechanism decomposes the dimensions that the self-attention mechanism needs to learn to extract the correct features more easily. Figure 2 shows the process of the multihead self-attention mechanism. First, Q, K, and V are, respectively, fed into multiple linear layers, then the attention mechanism is performed. Finally, the results obtained from each attention mechanism are concatenated and output through a linear layer.

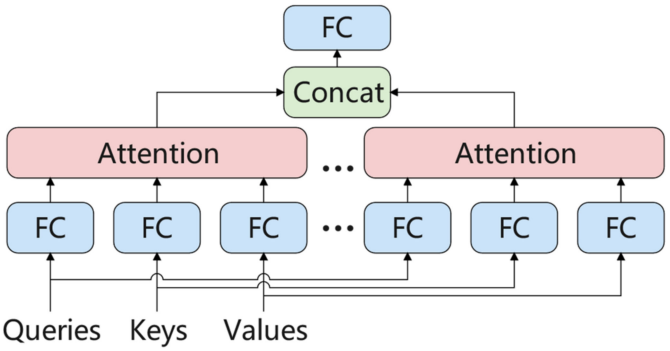


Figure 2. The multihead self-attention mechanism.

2.2. LayerNorm

The normalization layer serves to restrict the data to a certain range to facilitate subsequent processing, prevent gradient vanishing and gradient explosion, and speed up network training. The commonly used mechanisms are batch normalization (BatchNorm) [44], layer normalization (LayerNorm) [45], instance normalization (InstanceNorm) [46], group

normalization (GroupNorm) [47], etc.; the most commonly used is BatchNorm. LayerNorm normalizes on the hidden layer, i.e., it normalizes all neuronal inputs of a particular layer.

Figure 3 shows the difference between LayerNorm and BatchNorm. LayerNorm and BatchNorm are very similar, while their major difference is the normalization dimension. LayerNorm normalizes from the sample dimension, while BatchNorm normalizes from the feature dimension; therefore, LayerNorm has better performance for sequences of uncertain length compared to BatchNorm.

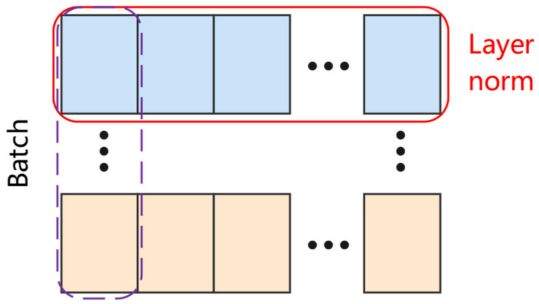


Figure 3. LayerNorm and BatchNorm.

2.3. Residual Connection and Dense Connection

Usually, the deeper a network is, the better the results. However, gradient vanishing and gradient explosion may occur when the network is very deep due to the nonlinearity of the activation function, making it difficult to improve the accuracy or even decrease the accuracy.

Residual connectivity is a connection method used between modules within the FusionNet, and originated from ResNet [48]. Such a connection structure allows shallow features to reach the deeper layers of the network so that networks with many layers can be trained [49].

The proposal of DenseNet [50] was inspired by ResNet, where the input of each module is a merge of all previous modules, enabling a more efficient transfer of features and gradients and making the network easier to train. Although the Dense Connection network is narrower and has fewer parameters, it brings an increase in the number of channels, making the network too complex and consuming a lot of memory during training. In contrast, the residual connections only add up the values, but the number of channels does not change.

2.4. Bottleneck

Bottleneck is a structure often used in ResNet [48], which uses several small 1×1 convolutional kernels to replace part of the work of the large middle convolutional kernel, reducing the number of parameters and reducing the network size significantly while achieving the same effect.

The remaining 3×3 convolutional kernels in the middle can also introduce some spatial information, and play a role similar to positional coding.

2.5. Proposed Method

In this section, the detailed structure of FusionNet is introduced. Figure 4 provides an overview of FusionNet. In general, the proposed FusionNet network is divided into four main parts: a preprocessing module, the Local Branch consisting of a residually connected convolution module, the Global Branch consisting of a Transformer module that incorporates convolution operations, i.e., the Convolution–Transformer Block, and a double-branch fusion mechanism. First, a 3D cube containing the center pixel need to be classified and its neighboring pixels is extracted from the original HSI image as the network input. After being processed by the preprocessing module, the data are fed into the Local Branch and Global Branch, respectively, for parallel extraction. The two branches can both

process data in parallel and fuse with each other: the double-branch fusion mechanism can fuse the information between two different branches to achieve a better utilization of the local and global spatial spectral information. Finally, the classification results obtained from the two branches are given equal weights, and the obtained probabilities are summed and passed through Softmax. The maximum value obtained is the prediction result.

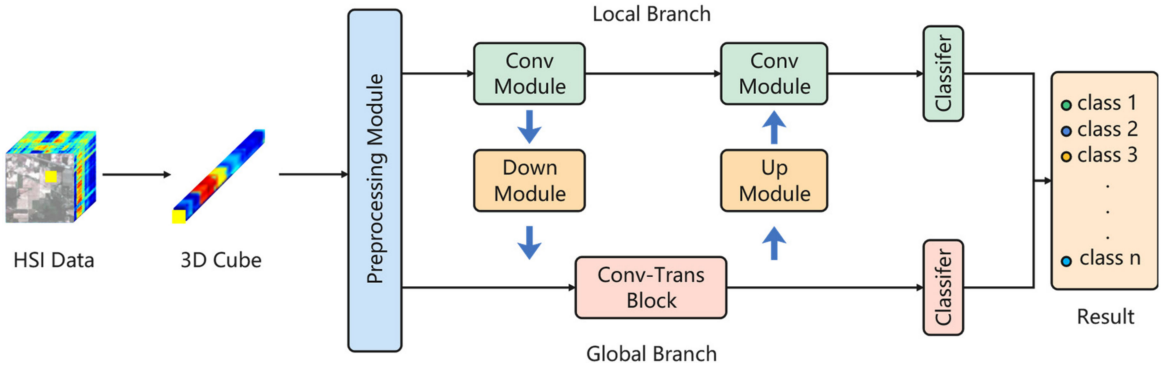


Figure 4. Overall structure of the proposed FusionNet model.

2.5.1. Preprocessing Module

Figure 5 illustrates the structure of the preprocessing module. HSI data are rich in spectral–spatial information, and the adopted data preprocessing method plays a crucial role in determining whether this information can be effectively mined and utilized. In this paper, the method of extracting the 3D cube from the original HSI data is adopted to obtain the input data of the network. Furthermore, a preprocessing module is designed according to the characteristics of each of the two branches. The preprocessing module will provide the two branches with input data suitable for both branches, in order to achieve better classification results. The Local Branch and Global Branch have their own characteristics regarding the way they utilize data. Especially for the Global Branch, the HSI data that are closer to the image may not be suitable to be directly used as inputs for the Global Branch. In the preprocessing module of FusionNet, different data preprocessing methods are designed for these two networks in such a way that the performance of each branch will be better utilized. First, the input 3D cube passes through a convolutional layer with a kernel size of 7 and a stride size of 2, and a pooling layer with a kernel size of 3 and a stride size of 2, to extract the local features of the data. For the Global Branch, an additional learnable class token is used and spliced with the output data of the pooling layer, and then passed through a transformer encoder module to obtain the input data of the Global Branch [39]; this is similar to the approach the vision Transformer model. Since the convolution module itself incorporates position information into the output, position encoding is no longer needed in the subsequent Transformer encoder module. For the Local Branch, the data from the pooling layer are passed through a three-layer convolution module with a bottleneck structure to obtain the input data for the Local Branch. The detailed structure of the convolution module will be described later.

2.5.2. Local Branch

The Local Branch is mainly composed of two consecutive convolution modules. Figure 6 illustrates the structure of the convolution module in the Local Branch. Inspired by [42], a serial convolution–Transformer connection was designed for HSI images. It is called the Local Branch since the convolution operation is better at capturing local features. The same applies to the other branch. Similar to the bottleneck layer in ResNet, each convolution module contains three layers of convolutional networks, i.e., two convolutional layers with kernel size 1×1 and stride 1 at the beginning and end, respectively, and one convolutional layer with kernel size 3×3 and stride 1 at the center, using residual connections between the beginning and the end. The bottleneck structure reduces the

dimensionality first and then raises it. The bottleneck reduces the number of parameters significantly while ensuring the efficient use of spatial features in HSI data, which enhances the efficiency of network training. In addition, the two convolution modules provide an interface for information interaction with the Global Branch. The output of the first convolution module will be utilized by the Global Branch, further enhancing the ability of the Global Branch to utilize local features; the input of the second convolution module will fuse the intermediate results from the Global Branch, further enhancing the ability of the Local Branch to characterize global features.

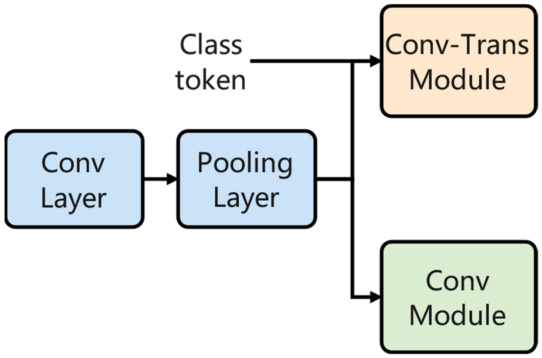


Figure 5. Preprocessing module.

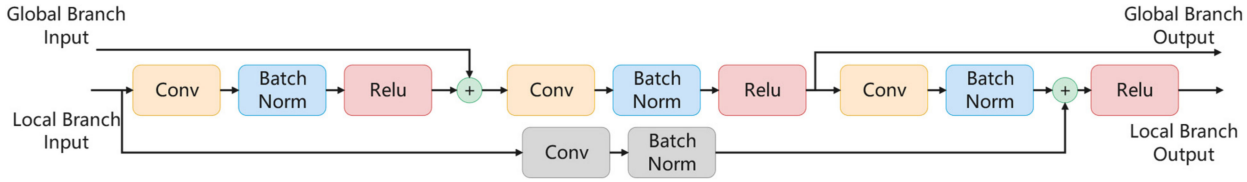


Figure 6. Convolution module in the Local Branch.

2.5.3. Global Branch

The Global Branch mainly consists of a convolution–Transformer module with an additional convolution module and an information interaction interface with the Local Branch. Benefiting from the attention mechanism, the Global Branch possesses a better ability to capture global features. In this module, the convolution–Transformer module and the Transformer module are connected together in a serial manner. Figure 7 illustrates the structure of the convolution–transformer module in the Local Branch. The module consists of four main submodules: MLP module, multihead self-attention module, convolution module, and layer normalization module, each of which is connected by residuals. The Transformer module has a sandwich-like network structure, i.e., the multihead self-attention module and the convolution module are sandwiched between two feedforward modules. The feedforward module mainly consists of two linear layers, and uses Swish as the activation function. The multihead self-attention module mainly consists of a multihead self-attention mechanism with the addition of a relative positional embedding. The convolution module uses depthwise convolution and pointwise convolution, which have lower parameter numbers and operation costs compared with traditional convolution. In addition, the convolution module also uses two activation layers, GLU (gated linear unit) [51] and Swish [52]. Assuming that the input of the convolution–Transformer module is x, and the final output is y, the output of the whole convolution–Transformer module can be shown by the following equation:

$$x_1 = x_0 + \frac{1}{2}MLP(x_0) \tag{6}$$

$$x_2 = x_1 + MHSA(x_1) \tag{7}$$

$$x_3 = x_2 + Convolution(x_2) \tag{8}$$

$$y = Layernorm(x_3 + \frac{1}{2}MLP(x_3)) \quad (9)$$

where *MLP* stands for *MLP* module, *MHSA* stands for multihead self-attention module, *Convolution* stands for convolution module, and *Layernorm* stands for layer normalization module.

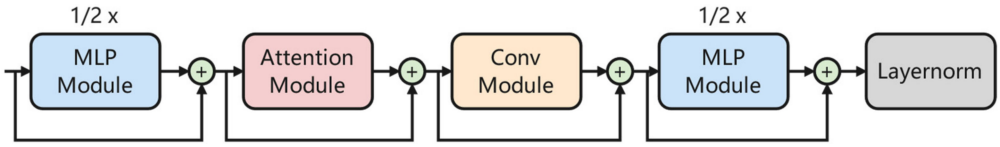


Figure 7. Convolution–Transformer module in the Local Branch.

2.5.4. Double-Branch Fusion Mechanism

The Local Branch has excellent extraction capability for local spatial features, while the Global Branch has the advantage of extracting global spatial and spectral features. Therefore, inspired by [43], the double-branch fusion mechanism is used to help the two branches exchange information so that each branch can take advantage of the other. Figure 8 illustrates the structure of the double-branch fusion mechanism. The double-branch fusion mechanism consists of a down module and an up module. The down module mainly consists of a convolutional layer of kernel size 1, a GELU (Gaussian error linear unit) [53] activation layer, and a mean pooling layer. Layer normalization is added to be responsible for incorporating the information of the Global Branch into the CNN branch. The up module consists of a kernel size of 1, a convolutional layer, a RELU activation layer, and a batch normalization module, and finally interpolates to complete the dimensional alignment with the Global Branch feature map. A convolutional layer of kernel size 1 is introduced to enable cross-channel information interaction and help further exploit the HSI data with rich spectral features. The two-branch fusion mechanism starts from the first convolution module in the Local Branch: the data output from the first convolution module travels through the down module to complete the dimensional alignment with the Global Branch. After that, it is sent to the convolution–Transformer module, together with the original input from the Global Branch. After convolution–Transformer module fusion, data from the Global Branch are returned to the Local Branch through the up module. The fusion mechanism ensures that both branches can recognize the features extracted from the other branch, further improving the network's ability to utilize local fine-grained features and global contextual relationships.

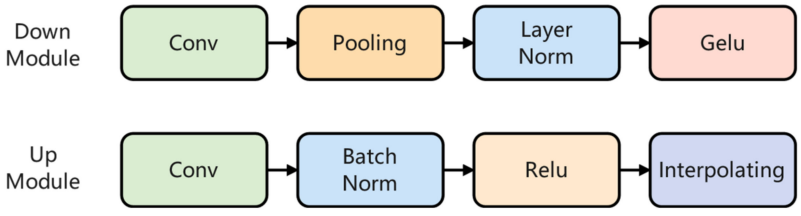


Figure 8. Double-Branch fusion mechanism.

3. Experiments

In this section, three well-known public hyperspectral image datasets are used. Next, the proposed FusionNet network is applied to these three datasets. The performance of the proposed FusionNet network is validated by comparing it with the prevailing network structures in the field of hyperspectral image classification.

3.1. Dataset Description

In experiments, to validate the classification capability of the proposed FusionNet, three most commonly used public datasets of hyperspectral images were selected for comparison: the Indian Pines dataset, the Pavia University dataset, and the recently

available Houston dataset. The basic information regarding the selected datasets is shown in Table 1.

Table 1. Basic information of each HSI dataset.

Dataset	Spatial Size	Spectral Size	Number of Classes	Labeled Samples
Indian Pines	145 × 145	200	16	10,249
Pavia University	610 × 340	103	9	42,776
Houston	349 × 1905	144	15	15,029

Indian Pines is the earliest test dataset for hyperspectral image classification, imaged by AVIRIS in 1992 on an Indian pine tree in Indiana, USA, with a spatial size of 145 × 145 pixels; the number of remaining spectral channels was 200 and the number of land cover classes was 16, after removing 20 of the bands that were interfered by noise. The detailed land cover categories and the corresponding sample numbers are shown in Table 2.

Table 2. Ground truth classes and samples of the Indian Pines dataset.

Order	Class	Labeled Samples
1	Alfalfa	46
2	Corn-notill	1428
3	Corn-mintill	830
4	Corn	237
5	Grass-pasture	483
6	Grass-trees	730
7	Grass-pasture-mowed	28
8	Hay-windrowed	478
9	Oats	20
10	Soybean-notill	972
11	Soybean-mintill	2455
12	Soybean-clean	593
13	Wheat	205
14	Woods	1265
15	Buildings-Grass-Trees-Drives	386
16	Stone-Steel-Towers	93

The false-color map and ground truth map of Indian Pines dataset are shown in Figure 9.

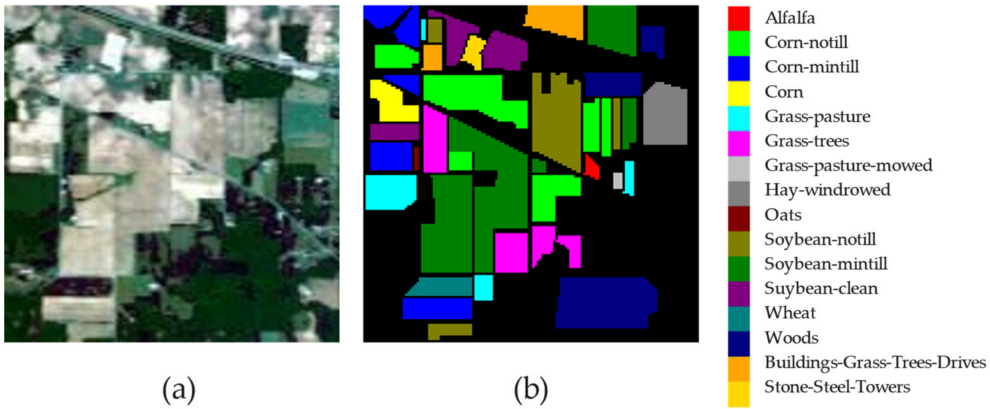


Figure 9. Indian Pines dataset: (**a**) false-color map, (**b**) ground truth map.

The Pavia University dataset was obtained from the German airborne Reflective Optics Spectrographic Imaging System (ROSIS), taken in 2003 at the University of Pavia in Italy.

The spatial size is 610 × 340 pixels. In the classification, 12 bands affected by noise were removed. The number of spectral channels used for the actual classification is 103, and the number of land cover categories is 9. The details of the land cover categories and the corresponding sample numbers are shown in Table 3.

Table 3. Ground truth classes and samples of the Pavia University dataset.

Order	Class	Labeled Samples
1	Asphalt	6631
2	Meadows	18,649
3	Gravel	2099
4	Trees	3064
5	Sheets	1345
6	Bare Soil	5029
7	Bitumen	1330
8	Bricks	3682
9	Shadows	947

The false-color map and ground truth map of the Pavia University dataset are shown in Figure 10.

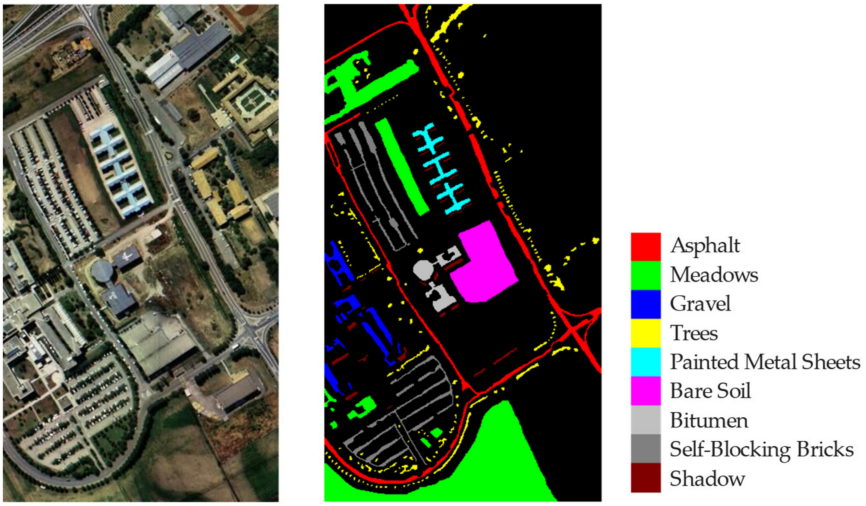


Figure 10. Pavia University dataset: (**a**) false-color map, (**b**) ground truth map.

The Houston dataset was sourced from the Hyperspectral Image Analysis Group at the University of Houston and the NSF-funded Center for Airborne Laser Mapping (NCALM). It was initially used in the IEEE GRSS data fusion competition in 2013. The spatial size is 349 × 1905 pixels. The dataset has a spatial size of 349 × 1905, with 144 spectral channels, and contains 15 classes. The details of the land cover categories and the corresponding sample numbers are shown in Table 4.

The false-color map and ground truth map of the Houston dataset are shown in Figure 11.

3.2. Experimental Settings

To evaluate the classification performance of the proposed FusionNet network, three widely adopted metrics were used, overall accuracy (OA), average accuracy (AA), and the Kappa coefficient, as the evaluation metrics. OA is the ratio of the number of correct predictions made by the network over all test sets to the overall number. AA is the average of the ratio of the number of correct predictions made in each category to the overall

number of that category, and the Kappa coefficient reflects the degree of consistency, and takes values in the range $[-1, 1]$. The higher the value achieved by the network on these three metrics, the better the classification of the network.

Table 4. Ground truth classes and samples of the Houston dataset.

Order	Class	Labeled Samples
1	Healthy grass	1251
2	Stressed grass	1254
3	Synthetic grass	697
4	Trees	1244
5	Soil	1242
6	Water	325
7	Residential	1268
8	Commercial	1244
9	Road	1252
10	Highway	1227
11	Railway	1235
12	Parking Lot 1	1233
13	Parking Lot 2	469
14	Tennis Court	428
15	Running Track	660

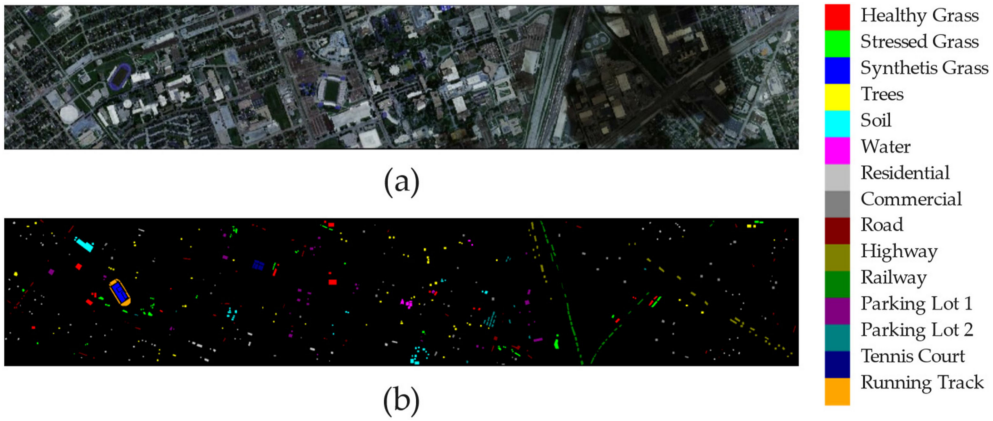


Figure 11. Houston dataset: (**a**) false-color map, (**b**) ground truth map.

The proposed FusionNet uses AdamW as the optimizer of the network. The learning rate is set to 0.0003, and the number of epochs is set to 50. The spatial size of the HSI cube is set to 15×15, and the batch size is set to 16. The experiment uses the Pytorch 1.9.0 deep learning framework, Python 3.7 development language, the Ubuntu 18.04.4 operating system, and NVIDIA GeForce RTX 2080 Ti graphics card to accelerate the training process of the proposed FusionNet network.

3.3. Experimental Results

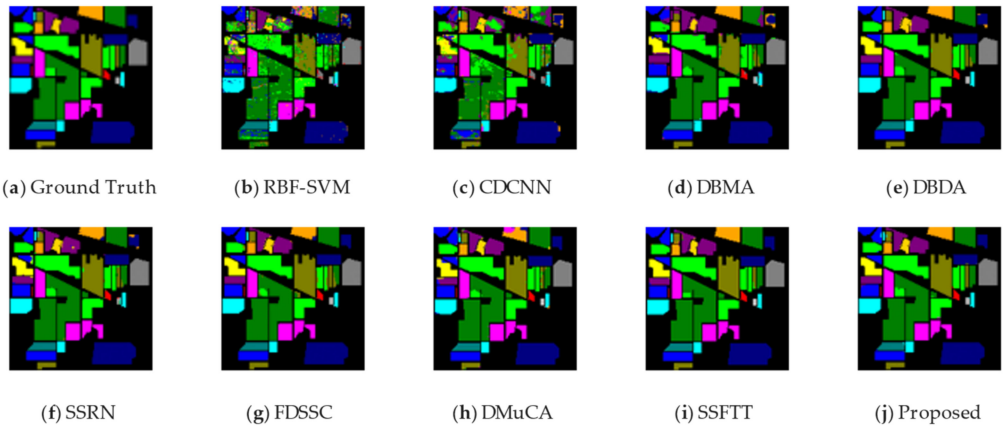
To demonstrate the effectiveness of the proposed model, several representative models were selected for comparison experiments with the proposed model: the RBF-SVM [10], CDCNN [30], DBMA [54], DBDA [9], SSRN [55], FDSSC [56], DMuCA [28], and SSFTT [41]. RBF-SVM is one of the most classical traditional HSI classification methods; CDCNN, SSRN, and FDSSC are representative convolution-based HSI classification methods; DBMA and DBDA are representative attention mechanism-based HSI classification methods; and DMuCA and SSFTT are recently proposed state-of-the-art HSI classification methods.

3.3.1. Classification Results of the Indian Pines Dataset

In total, 10% of the samples were randomly selected from the Indian Pines dataset as the training set to train the network, 10% as the validation set, and 80% as the test set. The average classification results of the 10 algorithms after 10 independent runs are given in Table 5, where the first 16 rows correspond to the classification results of each class, and the last three rows are the overall OA, AA, and Kappa coefficient, with the highest accuracy for a specific class shown in bold. Figure 12 shows the classification images obtained for the different methods, where Figure 12a indicates ground truth and Figure 12b–h are the results obtained using different classification networks.

Table 5. Classification results of the Indian Pines dataset using 10% training samples.

Class	RBF-SVM	DBMA	DBDA	CDCNN	SSRN	FDSSC	DMuCA	SSFTT	Proposed
1	55.07	58.53	94.82	98.37	93.42	97.33	87.80	**100.00**	97.63
2	71.34	80.88	97.13	98.73	97.94	99.09	94.39	95.95	**99.14**
3	75.53	73.48	98.60	99.03	99.13	99.22	96.52	98.39	**99.49**
4	61.18	72.95	97.50	99.24	98.32	98.38	83.89	99.53	**99.54**
5	88.76	95.37	97.40	98.73	99.28	99.05	98.16	**99.77**	97.05
6	89.16	94.32	98.74	99.38	99.80	99.41	99.54	**99.84**	99.27
7	85.05	60.33	84.65	94.52	94.90	98.37	92.97	**100.00**	96.60
8	90.32	91.46	99.39	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	99.95
9	71.14	79.94	93.52	97.25	**98.89**	90.94	86.27	94.44	97.69
10	75.74	74.93	96.04	97.65	97.23	96.96	97.94	99.42	**99.71**
11	77.97	84.11	97.14	99.17	98.77	98.35	99.50	99.19	**99.68**
12	73.24	61.32	93.98	98.93	98.44	**99.24**	94.57	96.82	97.69
13	90.80	96.24	98.83	99.64	99.82	99.09	94.59	**100.00**	99.74
14	91.74	93.37	98.78	99.38	99.36	99.22	99.29	**100.00**	99.96
15	74.41	81.90	93.98	98.17	98.38	98.37	92.35	98.56	**99.65**
16	**98.16**	97.00	96.02	92.97	94.07	96.31	97.98	88.10	95.53
OA (%)	80.01	81.76	97.10	98.87	98.63	98.68	96.62	98.67	**99.30**
AA (%)	79.35	81.01	96.03	98.20	97.98	98.08	94.74	98.13	**98.65**
Kappa (%)	77.09	79.27	96.69	98.71	98.44	98.49	96.29	98.48	**99.21**



(**a**) Ground Truth (**b**) RBF-SVM (**c**) CDCNN (**d**) DBMA (**e**) DBDA

(**f**) SSRN (**g**) FDSSC (**h**) DMuCA (**i**) SSFTT (**j**) Proposed

Figure 12. Classification maps of the Indian Pines dataset with 10% training samples.

From Table 5, it can be seen that the proposed FusionNet achieved the best classification results with OA of 99.30%, AA of 98.65%, and Kappa coefficient of 99.21%. The traditional classification method, i.e., RBF-SVM classification method, achieved the worst results with the lowest OA, AA, and Kappa coefficient metrics among all the networks used for comparison. CDCNN introduced a convolutional neural network and jointly used the local spectral relationships of adjacent individual pixel vectors for classification, thus achieving better classification results compared to RBF-SVM. However, it can be also observed that the CDCNN results are less stable: the standard deviation of its OA is as high

as 8.73% in 10 independent runs. The DBMA and DBDA networks introduced an attention mechanism for classification, and thus achieved more satisfactory classification results: the OA of DBMA improved by about 15% compared to the CDCNN. The DBDA results further improved compared to DBMA. The SSRN and FDSSC can effectively use spatial information and spectral features, and possess similar performances to DBMA, achieving OAs of 98.63% and 98.68% respectively, with good performance in the classification of several categories. DMuCA and SSFTT are more novel HSI classification methods. DMuCA captures spectral–spatial context dependencies through deep convolution and contextual attention mechanisms, and achieved 96.62% OA. SSFTT, which includes Transformer encoders in its network structure, achieved 98.67% OA and performed well on several categories. However, compared to all other networks, the proposed FusionNet achieved a superior performance in terms of overall results. Although the classification results of FusionNet are lower than the others in a few classes, FusionNet achieves the best performance in all three metrics, OA, AA, and the Kappa coefficient, revealing the excellent classification ability of FusionNet. Among the classification result images, it is found that the results of RBF-SVM showed the most noise, a large number of misclassified regions, and significantly worse results than the deep-learning-based classification networks. Among these methods, FusionNet exhibited the least noise and hardly any misclassified regions were visible from the classification result images. This corresponds to the results in Table 5.

3.3.2. Classification Results of the Pavia University Dataset

Compared with the Indian Pines dataset, the Pavia University dataset contains a larger amount of data, so it is advisable to use a smaller proportion of the training set to test the performance of FusionNet. In the Pavia University dataset, 5% of the data were randomly selected as the training set to train the network, 5% of the samples were selected as the validation set, and 90% of the samples were selected as the test set. The classification results of different networks on the Pavia University dataset are listed in Table 6, and the highest accuracies for specific categories are shown in bold. The visualization of the results obtained from different networks is shown in Figure 13. As can be seen from Table 6, the SVM still performs poorly in terms of OA, and its classification result images also show a large degree of mislabeling. This is significantly improved in the subsequent networks: DBMA, DBDA, CDCNN, SSRN, FDSSC, DMuCA, and SSFTT. The best classification results can be achieved using the proposed FusionNet network: the classification graph obtained using the proposed FusionNet network performs well on the classification maps, and the classification results are accurate and almost identical to the ground truth. Similarly, the proposed method achieved the best performance in the three metrics of OA, AA, and the Kappa coefficient, reaching 99.92%, 99.63%, and 98.64%, respectively.

Table 6. Classification results of the Pavia University dataset using 5% training samples.

Class	RBF-SVM	CDCNN	DBMA	DBDA	SSRN	FDSSC	DMuCA	SSFTT	Proposed
1	93.17	98.77	99.81	95.06	99.91	99.87	**99.92**	**99.92**	99.78
2	95.43	99.72	99.90	98.58	99.92	99.95	99.95	**99.99**	99.65
3	83.81	98.01	99.19	84.86	98.25	99.23	99.03	98.34	**99.71**
4	96.42	98.60	99.26	97.17	**99.93**	99.81	99.47	99.42	99.62
5	98.69	99.57	99.64	99.92	**100.00**	99.94	**100.00**	99.69	99.88
6	92.10	99.95	99.90	96.14	99.61	99.95	99.81	**100.00**	99.76
7	86.71	99.88	99.96	94.74	99.88	99.91	99.75	**100.00**	99.66
8	84.58	97.69	94.35	91.64	97.25	98.09	99.35	99.23	**99.75**
9	99.99	98.08	98.59	98.51	99.98	99.28	99.47	97.78	**100.00**
OA (%)	93.20	99.21	99.24	96.09	99.57	99.71	99.78	99.73	**99.92**
AA (%)	92.32	98.92	98.96	95.18	99.41	99.56	99.64	99.37	**99.76**
Kappa (%)	90.94	98.95	98.99	94.82	99.43	99.62	**99.85**	99.65	98.64

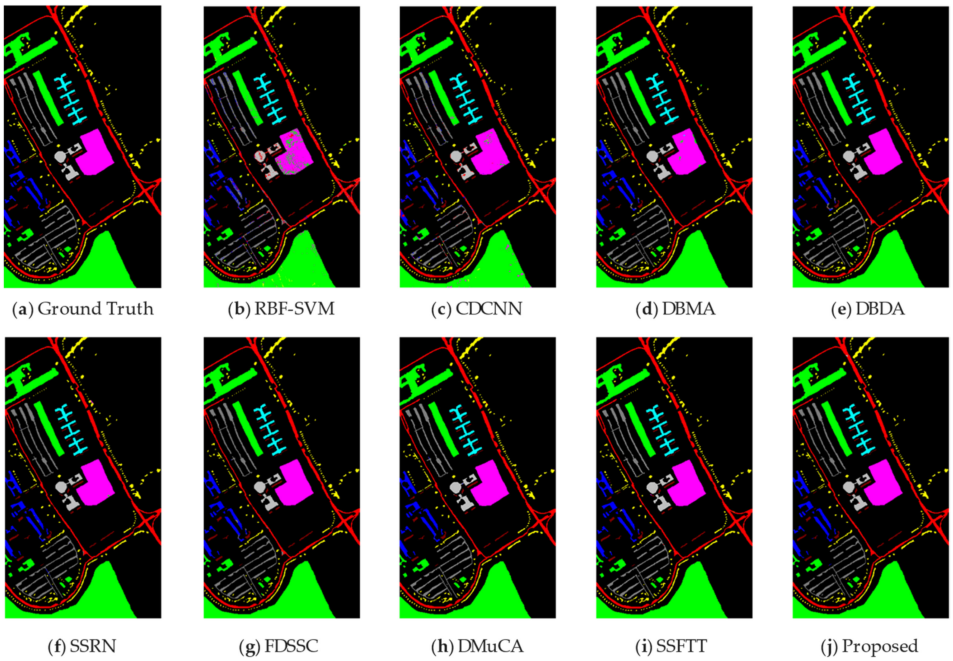


Figure 13. Classification maps of the Pavia University dataset with 5% training samples.

3.3.3. Classification Results of the Houston Dataset

The same ratio of dataset partitioning was adopted in the Houston dataset as in the Indian Pines dataset: 10% of the randomly selected feature samples were used as the training set to train the network, 10% as the validation set, and 80% as the test set. The classification results of different networks on the Houston dataset are listed in Table 7, and the highest accuracy for specific categories is shown in bold. The classification plots for the different methods are shown in Figure 14. As can be seen, although the proposed FusionNet network did not achieve the best results in all individual category classification accuracies, the proposed FusionNet network performed well in terms of overall classification results: accurate classification was achieved in the results of almost all categories, with individual category classification accuracies above 98%, and ranked first among all networks with an overall accuracy of 99.28% and an average accuracy of 99.33%.

Table 7. Classification results of the Houston dataset using 10% training samples.

Class	RBF-SVM	CDCNN	DBMA	DBDA	SSRN	FDSSC	DMuCA	SSFTT	Proposed
1	97.38	90.94	97.38	98.98	98.51	**99.54**	97.23	98.49	99.27
2	98.26	95.13	98.91	**99.91**	99.82	99.57	98.44	**99.91**	99.63
3	99.56	98.72	99.89	**100.00**	**100.00**	**100.00**	99.88	**100.00**	99.66
4	99.03	97.74	99.13	99.75	99.52	99.77	99.25	99.11	99.29
5	97.15	99.11	99.53	99.46	99.40	99.74	**100.00**	**100.00**	99.68
6	99.79	99.12	99.89	**100.00**	**100.00**	**100.00**	95.66	**100.00**	99.60
7	91.83	89.88	97.70	98.90	**99.32**	97.45	98.84	94.22	99.13
8	87.36	94.66	98.99	99.60	99.54	98.61	98.40	97.32	**99.69**
9	87.53	90.66	96.89	97.80	98.62	98.56	96.40	**99.29**	98.44
10	92.45	84.58	97.93	98.58	98.41	98.80	97.40	**100.00**	98.59
11	90.95	86.20	98.59	98.85	99.26	99.00	97.87	**100.00**	99.57
12	88.03	89.71	97.48	99.13	98.85	99.39	96.16	**99.64**	99.33
13	81.86	94.68	98.28	98.09	**99.41**	97.98	99.12	98.10	99.07
14	97.99	97.19	98.65	99.13	**100.00**	99.13	**100.00**	**100.00**	**100.00**
15	99.32	97.49	98.86	99.02	99.38	99.25	99.92	**100.00**	99.02
OA (%)	93.60	92.42	98.36	99.11	99.22	99.04	98.21	98.94	**99.28**
AA (%)	93.90	93.72	98.54	99.15	99.33	99.12	98.30	99.07	**99.33**
Kappa (%)	93.08	91.81	98.22	99.04	99.15	98.96	98.48	98.85	**99.23**

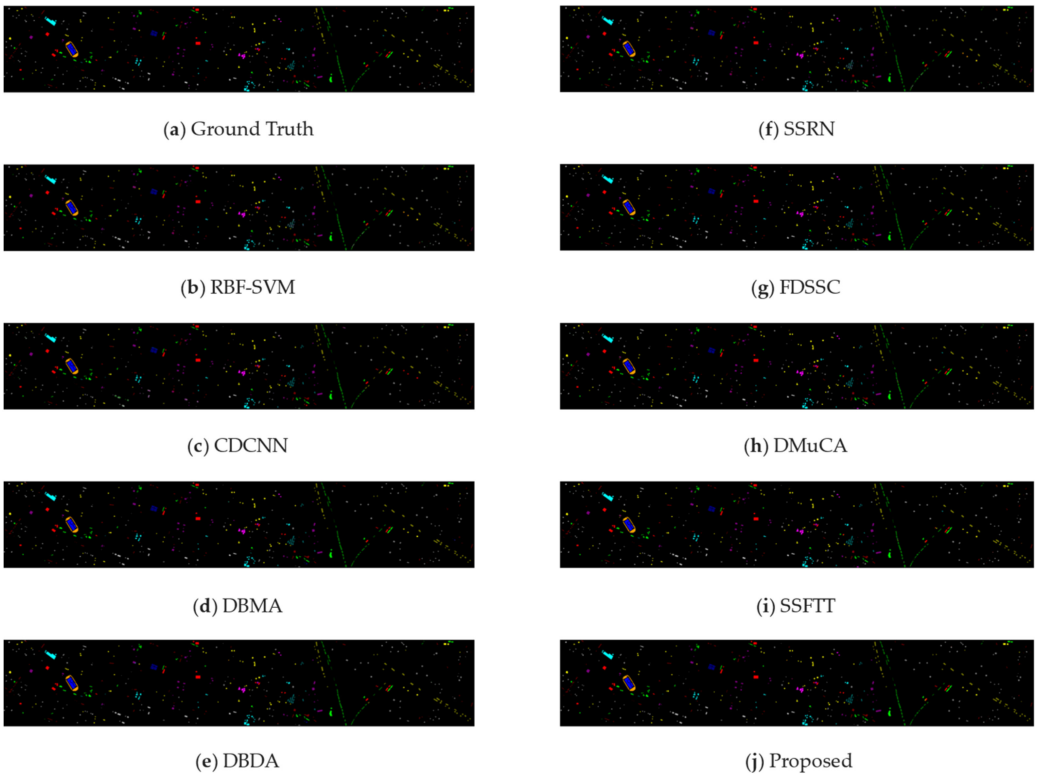


Figure 14. Classification maps of the Houston dataset with 10% training samples.

4. Discussion

The experimental results of FusionNet on three datasets from Indian Pines, Pavia University, and Houston demonstrate the power of the fusion of convolution and Transformer. This chapter will discuss the network structure and experimental results of FusionNet to further demonstrate the effectiveness and possible value of proposed structure for future research.

4.1. Ablation Study

This paper proposed a hybrid convolution–Transformer fusion pattern. The FusionNet structure with convolution operations and Transformer should be arranged in parallel overall and serially arranged locally (i.e., within the Global Branch), helping to further extract features from HSI data and further enhance the effectiveness of HSI classification methods. To prove the proposed point, two comparison models are designed based on the original FusionNet: (a) serial arrangement of convolutional and attention operations, i.e., using only the Convolution–Transformer Block as the core network structure for classifying and keeping the other modules unchanged; (b) parallel arrangement of convolutional and attention operations, i.e., the Convolution–Transformer Block in FusionNet is replaced by a normal Transformer encoder and keeps the other modules unchanged. Tables 8 and 9 show the comparison of the experimental results of (a), (b), and the original FusionNet for the three datasets. It can be observed that among the three networks, (a), which uses only serial alignment, performed the worst overall on the three datasets, especially when the amount of available training data is limited. For example, on the Indian Pines dataset, (a) achieves an overall accuracy of only 83.09%, which is not far from both (b) and the full FusionNet. In comparison, (b), which uses only parallel alignment, achieved better results relative to (a), but its results still fall short of those of FusionNet. The best results were obtained when fusing serial alignment with parallel alignment, i.e., using FusionNet for classification, with more significant improvements in both individual category classification accuracy, overall accuracy, average accuracy, and the Kappa coefficient, proving the effectiveness of the proposed method.

Table 8. Comparison results of (a), (b), and the original FusionNet on the Indian Pines dataset.

Class	(a)	(b)	FusionNet
1	77.54 ± 10.71	93.20 ± 12.20	**97.63 ± 3.40**
2	83.13 ± 3.74	93.74 ± 2.74	**99.14 ± 0.23**
3	80.97 ± 5.57	95.35 ± 1.65	**99.49 ± 0.29**
4	79.03 ± 4.71	96.43 ± 3.44	**99.54 ± 0.28**
5	86.49 ± 2.94	**99.24 ± 0.89**	97.95 ± 1.16
6	86.33 ± 3.92	98.83 ± 0.88	**99.27 ± 0.46**
7	76.84 ± 22.24	90.24 ± 13.54	**96.60 ± 2.23**
8	91.82 ± 4.07	98.27 ± 1.33	**99.95 ± 0.10**
9	89.14 ± 12.41	95.55 ± 8.88	**97.69 ± 3.53**
10	82.93 ± 2.05	94.14 ± 1.49	**99.71 ± 0.14**
11	81.00 ± 9.62	96.50 ± 1.07	**99.68 ± 0.27**
12	74.51 ± 5.11	90.71 ± 2.32	**97.69 ± 0.85**
13	90.44 ± 4.13	**99.75 ± 0.48**	99.74 ± 0.42
14	91.95 ± 5.10	97.35 ± 1.30	**99.96 ± 0.07**
15	76.65 ± 4.99	94.48 ± 2.27	**99.65 ± 0.43**
16	94.68 ± 4.78	**96.86 ± 1.79**	95.53 ± 2.83
OA (%)	83.09 ± 4.72	95.85 ± 0.47	**99.30 ± 0.20**
AA (%)	83.96 ± 2.59	95.67 ± 1.23	**98.65 ± 0.57**
Kappa (%)	80.56 ± 5.63	95.27 ± 0.54	**99.21 ± 0.22**

Table 9. Comparison results of (a), (b), and the original FusionNet on the Pavia University dataset.

Class	(a)	(a)	FusionNet
1	87.04 ± 7.66	96.03 ± 1.01	**99.78 ± 0.10**
2	97.21 ± 0.53	98.93 ± 0.34	**99.65 ± 0.22**
3	57.81 ± 35.19	94.18 ± 1.99	**99.71 ± 0.13**
4	98.38 ± 1.53	99.04 ± 0.69	**99.62 ± 0.21**
5	99.63 ± 0.35	**99.90 ± 0.10**	99.88 ± 0.04
6	82.27 ± 15.75	97.27 ± 0.72	**99.76 ± 0.21**
7	53.67 ± 43.86	93.64 ± 1.95	**99.66 ± 0.22**
8	80.63 ± 14.53	92.82 ± 0.96	**99.75 ± 0.30**
9	96.61 ± 5.46	96.03 ± 1.01	**100.00 ± 0.00**
OA (%)	89.40 ± 7.25	97.41 ± 0.18	**99.92 ± 0.13**
AA (%)	83.70 ± 13.05	96.84 ± 0.40	**99.63 ± 0.27**
Kappa (%)	86.14 ± 9.33	96.57 ± 0.24	**98.64 ± 1.16**

4.2. Parameter Analysis

4.2.1. Effect of Patch Size on Classification Results

To take full advantage of the spectral–spatial information of HSI, 3D cubes of a certain size around the central pixel were selected from the original data to be input into the network. The size of the 3D cubes selected in this way directly affects the amount of information used for classification and, to some extent, the classification accuracy. To evaluate the effect of the size of the cube on the classification results, the effect on the training results was explored by continuously increasing the patch size, i.e., by increasing the size of the cube in the spatial dimension. In this analysis, DMuCA, SSFTT, SSRN, FDSSC, DBMA, DBDA, and the proposed method were selected for comparison. The patch size is set to 5, 7, 9, 11, 13, 15, and 17, respectively, and the results obtained with the Indian Pines dataset are shown in Figure 15.

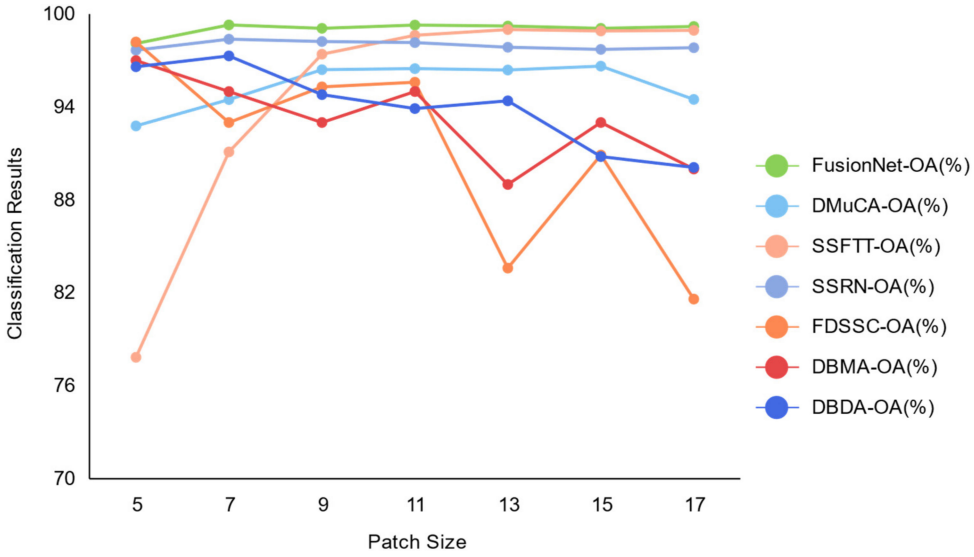


Figure 15. Effect of patch size on classification results.

From Figure 15, it can be seen that increasing the patch size will introduce more spectral–spatial information, which contributes to the classification accuracy. However, a larger patch size does not necessarily lead to better results. Larger patch size will bring higher computational cost and may also introduce more irrelevant information, thus causing a loss of accuracy. It can be seen that for a given different patch size, FusionNet is consistently able to exhibit high classification accuracy, reflecting good spectral–spatial information utilization. For FusionNet, it can be found that when the patch size is between 5 and 7, the overall accuracy, average accuracy, and kappa indicators all increase significantly with the increase in the spatial dimension input size. Additionally, when the patch size is between 7 and 11, the classification result indicators increase slowly, and when the patch size is larger than 11, the classification results no longer continue to grow with the increase in patch size. Considering the performance and cost, a patch size of 7 was chosen as the parameter used in the experiments, i.e., the size of the HSI cube is 15×15 in the spatial dimension.

4.2.2. Effect of Training Set Size on Classification Results

In practice, accurate manual labeling of pixels in large amounts of HSI data is relatively difficult, thus, the number of labeled samples available for training HSI networks may not be sufficient. Therefore, the proposed classification network should demonstrate good classification results under different training set sizes. To analyze the performance of the proposed FusionNet with different training set sizes, FusionNet was tested against the Indian Pines dataset, and was compared with other models, namely RBF-SVM, CDCNN, DBMA, DBDA, SSRN, FDSSC, DMuCA, and SSFTT. All models were tested using training set proportions of 3%, 5%, 10%, 15%, and 20%, respectively, and the results obtained are shown in Figure 16. From the results, it can be seen that the classification results achieved by FusionNet improve significantly as the proportion of the training set increases. It is also observed that FusionNet, SSFTT, FDSSC, and DBDA achieved good classification results with a small percentage of the training set (3%). Meanwhile, FusionNet still achieved competitive classification results compared to other models, i.e., an OA of 93.36%. When the training set percentage reached 10%, the OA of FusionNet had already reached 99.30%. The experimental results show that FusionNet has good adaptability to different training set sizes.

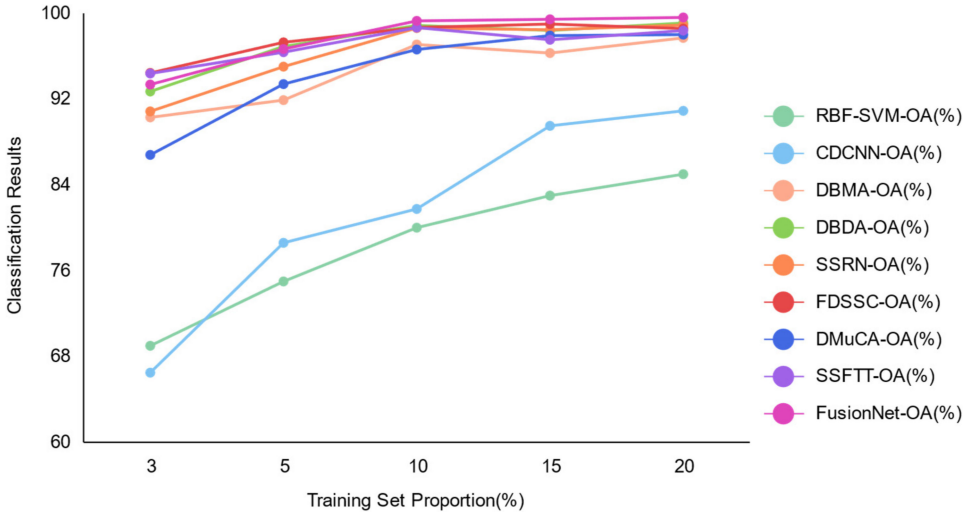


Figure 16. Effect of training set size on classification results.

5. Conclusions

In this paper, the FusionNet and a two-branch convolution–Transformer fusion network for HSI classification, and innovatively designed parallel and serial convolution–Transformer fusion methods are proposed. The proposed network has two branches, the Local Branch that focuses on local information extraction in spatial and spectral dimensions, and the Global Branch, which is responsible for extracting global information. Subsequently, the features of the two branches are fused to form spatial–spectral features for the final classification results. Overall, the convolution and Transformer modules are fused in parallel, while the convolution and Transformer modules are serially arranged to further enhance the extraction capability of the local features within the Global Branch. Experimental results for three genuine HSI datasets show that the proposed FusionNet network exhibits excellent classification results with a significant advantage over previous works, especially when the labeled training data are very limited. This work provides an optimization scheme for the processing and analysis of HSI classification. Future research will focus on optimizing the network structure to further improve the classification accuracy and further reduce the dependence on the amount of training data.

Author Contributions: Investigation, L.Y., Y.Y. and J.Y.; Methodology, L.Y., Y.Y., N.Z. and J.Y.; Writing, L.Y. and Y.Y.; Editing, J.Y., T.W. and L.W. (Ling Wu); Valuable advice, L.W. (Liguo Wang). All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (Nos. 62001434, 62071084).

Data Availability Statement: Publicly available datasets were analyzed in this study, which can be found here: http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes (accessed on 16 March 2022) and https://hyperspectral.ee.uh.edu/?page_id=459 (accessed on 17 March 2022).

Acknowledgments: The authors would like to thank the authors of all references used in the paper, the editors, and the anonymous reviewers for their detailed comments and suggestions.

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

References

1. Landgrebe, D. Hyperspectral image data analysis. *IEEE Signal Process. Mag.* **2002**, *19*, 17–28. [CrossRef]
2. Ahmed, N.; Khan, S.H.; Anjum, M.A.; Rehman, A. A cost effective preparative thin layer chromatography cleanup method for high performance liquid chromatography analysis of aflatoxins B1, B2 and G2. *Adv. Life Sci.* **2014**, *2*, 1–4.
3. Pipitone, C.; Maltese, A.; Dardanelli, G.; Lo Brutto, M.; La Loggia, G. Monitoring water surface and level of a reservoir using different remote sensing approaches and comparison with dam displacements evaluated via GNSS. *Remote Sens.* **2018**, *10*, 71. [CrossRef]

4. Awad, M.; Jomaa, I.; Arab, F. Improved capability in stone pine forest mapping and management in Lebanon using hyperspectral CHRIS-Proba data relative to Landsat ETM+. *Photogramm. Eng. Remote Sens.* **2014**, *80*, 725–731. [CrossRef]
5. Luo, B.; Yang, C.; Chanussot, J.; Zhang, L. Crop yield estimation based on unsupervised linear unmixing of multidate hyperspectral imagery. *IEEE Trans. Geosci. Remote Sens.* **2012**, *51*, 162–173. [CrossRef]
6. Bishop, C.A.; Liu, J.G.; Mason, P.J. Hyperspectral remote sensing for mineral exploration in Pulang, Yunnan Province, China. *Int. J. Remote Sens.* **2011**, *32*, 2409–2426. [CrossRef]
7. Bhatt, J.S.; Joshi, M.V. Deep learning in hyperspectral unmixing: A review. In Proceedings of the IGARSS 2020—2020 IEEE International Geoscience and Remote Sensing Symposium, Waikoloa, HI, USA, 26 September–2 October 2020; pp. 2189–2192.
8. Zhao, X.; Li, W.; Shan, T.; Li, L.; Tao, R. Hyperspectral target detection by fractional Fourier transform. In Proceedings of the IGARSS 2020—2020 IEEE International Geoscience and Remote Sensing Symposium, Waikoloa, HI, USA, 26 September–2 October 2020; pp. 1655–1658.
9. 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]
10. Melgani, F.; Bruzzone, L. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. Geosci. Remote Sens.* **2004**, *42*, 1778–1790. [CrossRef]
11. Aswathy, C.; Sowmya, V.; Gandhiraj, R.; Soman, K. Hyperspectral image denoising using legendre Fenchel Transformation for improved Multinomial Logistic Regression based classification. In Proceedings of the 2015 International Conference on Communications and Signal Processing (ICCSP), Melmaruvathur, India, 2–4 April 2015; pp. 1670–1674.
12. Belgiu, M.; Drăguţ, L. Random forest in remote sensing: A review of applications and future directions. *ISPRS J. Photogramm. Remote Sens.* **2016**, *114*, 24–31. [CrossRef]
13. Chen, Y.; Nasrabadi, N.M.; Tran, T.D. Hyperspectral image classification via kernel sparse representation. *IEEE Trans. Geosci. Remote Sens.* **2012**, *51*, 217–231. [CrossRef]
14. Tarabalka, Y.; Chanussot, J.; Benediktsson, J.A. Segmentation and classification of hyperspectral images using watershed transformation. *Pattern Recognit.* **2010**, *43*, 2367–2379. [CrossRef]
15. Chen, Y.; Nasrabadi, N.M.; Tran, T.D. Hyperspectral image classification using dictionary-based sparse representation. *IEEE Trans. Geosci. Remote Sens.* **2011**, *49*, 3973–3985. [CrossRef]
16. Ahmad, M.; Shabbir, S.; Roy, S.K.; Hong, D.; Wu, X.; Yao, J.; Khan, A.M.; Mazzara, M.; Distefano, S.; Chanussot, J. Hyperspectral image classification—Traditional to deep models: A survey for future prospects. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *15*, 968–999. [CrossRef]
17. 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]
18. Voulodimos, A.; Doulamis, N.; Doulamis, A.; Protopapadakis, E. Deep learning for computer vision: A brief review. *Comput. Intell. Neurosci.* **2018**, *2018*, 7068349. [CrossRef]
19. Otter, D.W.; Medina, J.R.; Kalita, J.K. A survey of the usages of deep learning for natural language processing. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 604–624. [CrossRef]
20. Sun, L.; Cheng, S.; Zheng, Y.; Wu, Z.; Zhang, J. SPANet: Successive Pooling Attention Network for Semantic Segmentation of Remote Sensing Images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 4045–4057. [CrossRef]
21. Wang, J.; Song, X.; Sun, L.; Huang, W.; Wang, J. A novel cubic convolutional neural network for hyperspectral image classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 4133–4148. [CrossRef]
22. 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]
23. Hu, W.; Huang, Y.; Wei, L.; Zhang, F.; Li, H. Deep Convolutional Neural Networks for Hyperspectral Image Classification. *J. Sens.* **2015**, *2015*, 258–619. [CrossRef]
24. 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]
25. Zhu, L.; Chen, Y.; Ghamisi, P.; Benediktsson, J.A. Generative adversarial networks for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 5046–5063. [CrossRef]
26. Mou, L.; Lu, X.; Li, X.; Zhu, X.X. Nonlocal graph convolutional networks for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 8246–8257. [CrossRef]
27. Jiang, Y.; Li, Y.; Zou, S.; Zhang, H.; Bai, Y. Hyperspectral image classification with spatial consistence using fully convolutional spatial propagation network. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 10425–10437. [CrossRef]
28. Liang, M.; He, Q.; Yu, X.; Wang, H.; Meng, Z.; Jiao, L. A Dual Multi-Head Contextual Attention Network for Hyperspectral Image Classification. *Remote Sens.* **2022**, *14*, 3091. [CrossRef]
29. Yue, J.; Zhao, W.; Mao, S.; Liu, H. Spectral–spatial classification of hyperspectral images using deep convolutional neural networks. *Remote Sens. Lett.* **2015**, *6*, 468–477. [CrossRef]
30. Lee, H.; Kwon, H. Going deeper with contextual CNN for hyperspectral image classification. *IEEE Trans. Image Process.* **2017**, *26*, 4843–4855. [CrossRef]
31. 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.* **2019**, *17*, 277–281. [CrossRef]

32. Chang, Y.-L.; Tan, T.-H.; Lee, W.-H.; Chang, L.; Chen, Y.-N.; Fan, K.-C.; Alkhaleefah, M. Consolidated Convolutional Neural Network for Hyperspectral Image Classification. *Remote Sens.* **2022**, *14*, 1571. [CrossRef]
33. Meng, Z.; Jiao, L.; Liang, M.; Zhao, F. A lightweight spectral-spatial convolution module for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 1–5. [CrossRef]
34. He, X.; Chen, Y.; Lin, Z. Spatial-spectral transformer for hyperspectral image classification. *Remote Sens.* **2021**, *13*, 498. [CrossRef]
35. 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.* **2021**, *60*, 1–15. [CrossRef]
36. Xu, Y.; Du, B.; Zhang, L. Self-attention context network: Addressing the threat of adversarial attacks for hyperspectral image classification. *IEEE Trans. Image Process.* **2021**, *30*, 8671–8685. [CrossRef]
37. Fang, B.; Li, Y.; Zhang, H.; Chan, J.C.-W. Hyperspectral images classification based on dense convolutional networks with spectral-wise attention mechanism. *Remote Sens.* **2019**, *11*, 159. [CrossRef]
38. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *arXiv* **2017**, arXiv:1706.03762.
39. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
40. Liu, B.; Yu, A.; Gao, K.; Tan, X.; Sun, Y.; Yu, X. DSS-TRM: Deep spatial–spectral transformer for hyperspectral image classification. *Eur. J. Remote Sens.* **2022**, *55*, 103–114. [CrossRef]
41. 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. [CrossRef]
42. Gulati, A.; Qin, J.; Chiu, C.-C.; Parmar, N.; Zhang, Y.; Yu, J.; Han, W.; Wang, S.; Zhang, Z.; Wu, Y. Conformer: Convolution-augmented transformer for speech recognition. *arXiv* **2020**, arXiv:2005.08100.
43. Peng, Z.; Huang, W.; Gu, S.; Xie, L.; Wang, Y.; Jiao, J.; Ye, Q. Conformer: Local Features Coupling Global Representations for Visual Recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 367–376.
44. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *Proc. Mach. Learn. Res.* **2015**, *37*, 448–456.
45. Ba, J.L.; Kiros, J.R.; Hinton, G.E. Layer normalization. *arXiv* **2016**, arXiv:1607.06450.
46. Ulyanov, D.; Vedaldi, A.; Lempitsky, V. Instance normalization: The missing ingredient for fast stylization. *arXiv* **2016**, arXiv:1607.08022.
47. Wu, Y.; He, K. Group normalization. In Proceedings of the European Conference on Computer Vision (ECCV) 15th European Conference, Munich, Germany, 8–14 September 2018; pp. 3–19.
48. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
49. Srivastava, R.K.; Greff, K.; Schmidhuber, J. Highway networks. *arXiv* **2015**, arXiv:1505.00387.
50. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
51. Dauphin, Y.N.; Fan, A.; Auli, M.; Grangier, D. Language modeling with gated convolutional networks. *Proc. Mach. Learn. Res.* **2017**, *70*, 933–941.
52. Ramachandran, P.; Zoph, B.; Le, Q.V. Searching for activation functions. *arXiv* **2017**, arXiv:1710.05941.
53. Hendrycks, D.; Gimpel, K. Gaussian error linear units (gelus). *arXiv* **2016**, arXiv:1606.08415.
54. Ma, W.; Yang, Q.; Wu, Y.; Zhao, W.; Zhang, X. Double-branch multi-attention mechanism network for hyperspectral image classification. *Remote Sens.* **2019**, *11*, 1307. [CrossRef]
55. 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.* **2017**, *56*, 847–858. [CrossRef]
56. Wang, W.; Dou, S.; Jiang, Z.; Sun, L. A fast dense spectral–spatial convolution network framework for hyperspectral images classification. *Remote Sens.* **2018**, *10*, 1068. [CrossRef]