remote sensing

Article

A Hyperspectral Image Classification Method Based on Adaptive Spectral Spatial Kernel Combined with Improved Vision Transformer

Aili Wang [1], Shuang Xing [1], Yan Zhao [2], Haibin Wu [1,*] and Yuji Iwahori [3]

[1] Heilongjiang Province Key Laboratory of Laser Spectroscopy Technology and Application, Harbin University of Science and Technology, Harbin 150080, China; aili925@hrbust.edu.cn (A.W.); 2020600033@stu.hrbust.edu.cn (S.X.)
[2] Communication Construction Operation and Maintenance Center, State Grid Heilongjiang Electric Power Co., Ltd. Information and Communication Company, Harbin 150010, China; q_peach@126.com
[3] Department of Computer Science, Chubu University, Aichi 487-8501, Japan; iwahori@isc.chubu.ac.jp
* Correspondence: woo@hrbust.edu.cn

Abstract: In recent years, methods based on deep convolutional neural networks (CNNs) have dominated the classification task of hyperspectral images. Although CNN-based HSI classification methods have the advantages of spatial feature extraction, HSI images are characterized by approximately continuous spectral information, usually containing hundreds of spectral bands. CNN cannot mine and represent the sequence properties of spectral features well, and the transformer model of attention mechanism proves its advantages in processing sequence data. This study proposes a new spectral spatial kernel combined with the improved Vision Transformer (ViT) to jointly extract spatial spectral features to complete classification task. First, the hyperspectral data are dimensionally reduced by PCA; then, the shallow features are extracted with an spectral spatial kernel, and the extracted features are input into the improved ViT model. The improved ViT introduces a re-attention mechanism and a local mechanism based on the original ViT. The re-attention mechanism can increase the diversity of attention maps at different levels. The local mechanism is introduced into ViT to make full use of the local and global information of the data to improve the classification accuracy. Finally, a multi-layer perceptron is used to obtain the classification result. Among them, the Focal Loss function is used to increase the loss weight of small-class samples and difficult-to-classify samples in HSI data samples and reduce the loss weight of easy-to-classify samples, so that the network can learn more useful hyperspectral image information. In addition, using the Apollo optimizer to train the HSI classification model to better update and compute network parameters that affect model training and model output, thereby minimizing the loss function. We evaluated the classification performance of the proposed method on four different datasets, and achieved good classification results on urban land object classification, crop classification and mineral classification, respectively. Compared with the state-of-the-art backbone network, the method achieves a significant improvement and achieves very good classification accuracy.

Keywords: hyperspectral image (HSI); image classification; feature extraction; vision transformer

check for updates

Citation: Wang, A.; Xing, S.; Zhao, Y.; Wu, H.; Iwahori, Y. A Hyperspectral Image Classification Method Based on Adaptive Spectral Spatial Kernel Combined with Improved Vision Transformer. *Remote Sens.* **2022**, *14*, 3705. https://doi.org/10.3390/rs14153705

Academic Editors: Liguo Wang, Yanfeng Gu and Peng Wang

Received: 20 June 2022
Accepted: 30 July 2022
Published: 2 August 2022

1. Introduction

Hyperspectral imagery (Hyperspectral Imagery, HSI) is an image acquired by a hyperspectral imager, and its spatial and spectral information is very rich. Compared with ordinary images, hyperspectral remote sensing images also have more bands and extremely high resolution. The application of hyperspectral remote sensing to earth observation technology is very common, such as precision agriculture [1], land cover analysis [2], marine hydrology detection [3], geological exploration [4] and other fields.

Hyperspectral Imagery (HSI) classification is an important task in hyperspectral image processing and application. In the early research, many traditional machine learning methods have been applied to hyperspectral image classification, such as K-nearest neighbor method [5], support vector machine [6], random forest [7], naive Bayes [8] and decision trees [9], etc. Although these traditional methods have achieved good performance, they are all based on shallow features for learning classification and rely on manual design of classification features, which is difficult to learn more complex information in hyperspectral images [10].

The hyperspectral image classification algorithm based on deep learning can automatically obtain the advanced features of the image, so that the classification model can better express the characteristic of the remote sensing image to improve classification accuracy. Chen [11] applied deep learning theory to hyperspectral image classification for the first time, which used stacked autoencoders to extract spatial spectral features from hyperspectral images and achieved good results. Yu [12] applied convolutional neural networks (CNN) to hyperspectral image classification, which only used spectral information, without taking into account the relationship between adjacent cells. Chen et al. [13] proposed a three-dimensional convolutional neural network (3D-CNN) feature extraction model to directly extract spectral spatial features and achieve better classification results from hyperspectral images end-to-end, which has higher inter-class distinguishability compared to two-dimensional convolutional neural networks (2D-CNN). Roy [14] proposed the HybridSN framework, which is a spectral–spatial 3D-CNN followed by spatial 2D-CNN to further learn a more abstract spatial representation. Zhong et al. [15] proposed the SSRN network, where a spectral residual block and a spatial residual block sequentially learn discriminative features from the rich spectral features and spatial context in hyperspectral images. The selection of informative spectral–spatial kernel features presents challenges due to the presence of noise and band correlations, which is usually solved by using a convolutional neural network with a fixed size receptive field (RF). Roy et al. [16] proposed an attention-based adaptive spectral spatial kernel modified residual network (A2S2K-ResNet) with spectral attention to capture discriminative spectral spatial features in an end-to-end training manner, using an improved 3D ResBlocks to jointly extract spectral–spatial features for HSI classification. T. Alipour-Fard et al. [17] proposed a new multi-branch selection kernel network (MSKNet), which uses different receptive field sizes to convolve the input image to generate multiple branches, so as to adjust each branch according to the input contrast through the attention mechanism effect of the channel. Automatically adjusting the size of neuron receptive field and enhance the cross-channel relationship between features improves the problem of using fixed size receptive field in convolutional neural network, so as to limit the learning weight of the model. Although CNN-based methods have the advantages of spatial feature extraction, they are difficult to handle continuous data, and CNNs are not good at modeling long-range dependencies.

Recently, the application of transformers in the visual direction has become a hot topic. The spectrum of HSI is a kind of sequence data, which usually contains hundreds of spectral bands. Attention-based transformer models have demonstrated their advantages in handling sequential data, and the transformer framework can represent high-level semantic features well. Although CNN has good local perception ability, due to the limitation of the inherent network backbone, CNN cannot mine and represent the sequence attributes of spectral features well, while the transformer model based on an attention mechanism enables the model to be trained in parallel and has global information. CNN methods have limited ability to acquire deep semantic features. As the depth increases, the traditional CNN will increase the channel dimension and reduce the spatial dimension, and the computational cost will increase significantly. However, the transformer does not have this problem, and the channels and spatial dimensions of different layers do not change. This strategy of reducing the spatial dimension and increasing the channel dimension is also beneficial to improve the performance of the transformer structure.

He et al. [18] used CNN to extract spatial features and a transformer to capture spectral sequence relationships. Hong et al. [19] proposed the SpectralFormer network architecture to learn spectral local sequence information from adjacent bands of HSI images to generate intra-group spectral embeddings. Ji et al. [20] proposed the bidirectional encoder representations from transformers (BERT), which has a global receptive field and can directly capture the global dependencies between pixels without considering their spatial distances. Han et al. [21] proposed that Transformer iN Transformer (TNT) block uses an outer transformer block to model the relationship between patches and an inner Transformer block to model the relationship between pixels. The model not only retains the information extraction at the patch level but also achieves the information extraction at the pixel level, which can significantly improve the model's ability to model local structures. Hugo et al. [22] used distillation to enable the transformer-based model to learn some inductive biases based on the CNN model, thereby improving the processing capability of image. Although the global interaction between token embeddings can be well modeled by the transformer's self-attention mechanism, the locality mechanism for information exchange within local regions is lacking. Li et al. [23] introduced locality into the transformer by introducing depthwise convolutions in a feedforward network.

In order to capture the spectral relationship of HSI sequences over long distances, obtain deep semantic features and make full use of the local and global information of the data, this paper proposes a new classification framework, that is, attention-based adaptive spectral spatial kernel combined with ViT. The contributions of this article are summarized as follows:

1. This study proposes a novel HSI classification architecture, the attention-based adaptive spectral spatial kernel combined with improved ViT, which systematically combines bands from shallow to deep, enables neurons to adaptively adjust the receptive field size and successfully handles the long-range dependence of the spectrum making full use of the spectral spatial information and local global information in HSI to improve the classification performance of HSI.

2. This study proposes an improved ViT model that introduces the re-attention mechanism and the local mechanism. We use the re-attention mechanism to increase the diversity of attention maps at different levels. The local mechanism is introduced into ViT, the improved attention mechanism of ViT global relation modeling and the locality mechanism of local information aggregation are combined to make full use of the local and global information of the data and improve the classification accuracy.

3. In order to train the model better, the Focal Loss function is used to increase the loss weight of small-class samples and hard-to-classify samples in HSI data samples and reduce the loss weight of easy-to-classify samples, so that the network can learn more useful hyperspectral image information. In addition, using the Apollo optimizer to train the HSI classification model resulted in better updating and computing network parameters that affect model training and model output, thereby minimizing the loss function. The smaller the loss function, the better the model, thus improving the classification model's performance.

4. The effectiveness of the method is verified on the challenging HSI four public datasets, the urban ground object classification is realized in the Pavia University dataset, the mineral classification is realized on the Xuzhou dataset, and the classification of crops is implemented on the Indian Pines and WHU-Hi-LongKou datasets. The effectiveness of the method is demonstrated on public datasets in different application domains. Compared with other representative methods, the classification results accuracy of the proposed method is improved.

The remaining part of this paper is organized as follows. Section 2 describes the details of the proposed classification method in detail. Section 3 describes the experimental datasets, experimental results and related analyses. Section 4 gives conclusions and suggestions for future work.

2. Related Works

The framework proposed in this paper for HSI image classification is shown in Figure 1. First, the principal component analysis (PCA) method is used to remove redundant spectra and reduce the time and space complexity of image processing. Considering that in order to effectively adjust the receptive field size of neurons and cross-channel dependencies, we proposed an attention-based adaptive spectral spatial residual method. Since CNN is good for capturing local information but has difficulty processing HSI's continuous data, the extracted features are sent to the modified Vision Transformer model. The original ViT model is improved by combining it with Re-Attention mechanism to increase the diversity of the attention graph at different levels. Then, the local mechanism is introduced into the ViT, locality is added to the ViT by introducing a depthwise convolution in the feedforward network, and the transformed features are fed into the transformer encoder modules to perform feature representation and learning. The following part is divided into four parts: attention-based adaptive spectral spatial residual module, improved ViT, Apollo optimizer and Focal loss function.

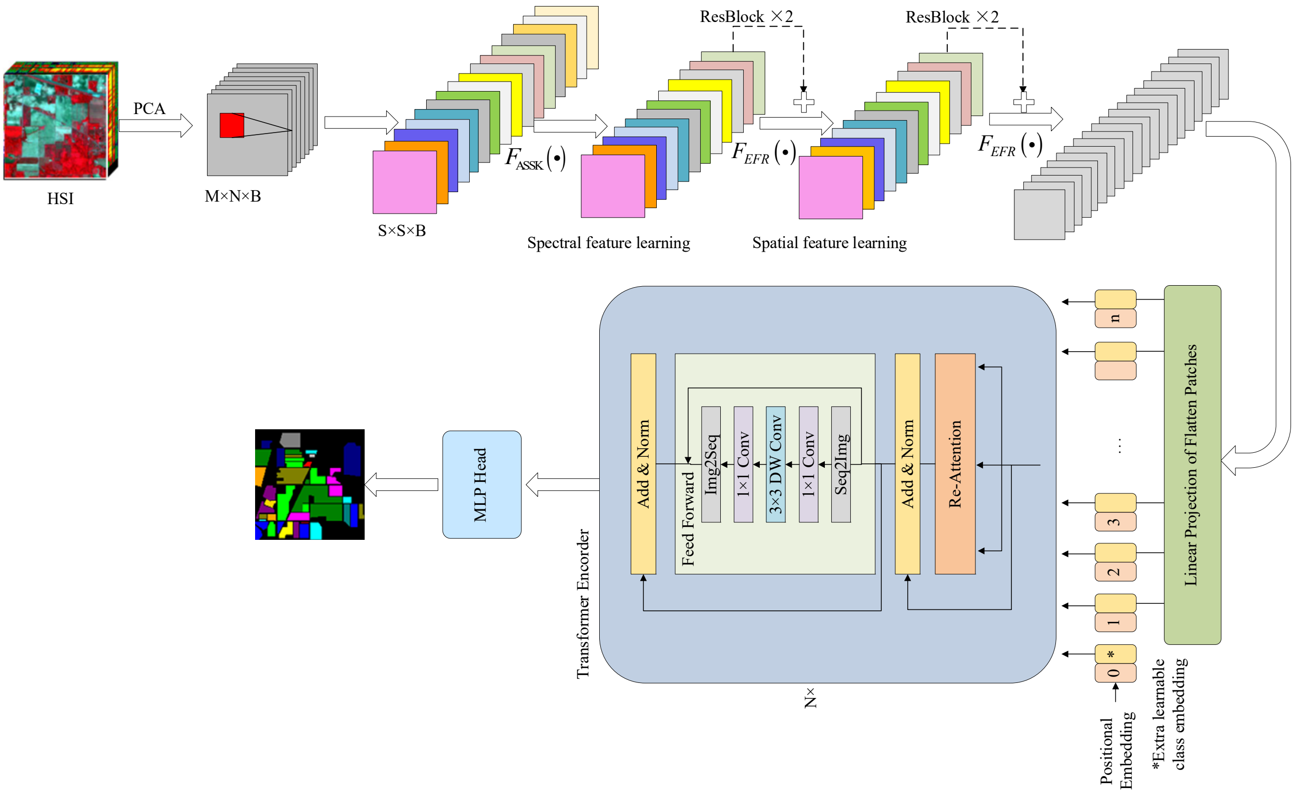


Figure 1. The proposed network structure for HSI Classification.

2.1. Spectral–Spatial Feature Extraction

Let the hyperspectral data cube be $I \in \mathbb{R}^{M\times N\times D}$, where I is the original input, M is the width, N is the height, and D is the number of spectral bands. Every HSI pixel in I contains D spectral measures and forms a one-hot vector $Y = (Y_1, Y_2, \cdots, Y_C) \in \mathbb{R}^{1\times 1\times C}$, where C represents the land cover category. To remove spectral redundancy, a principal component analysis was first performed on the raw input HSI data to reduce the number of spectral bands from D to B while maintaining the same spatial dimension. Let $X \in \mathbb{R}^{M\times N\times B}$ be the data cube after PCA processing and B be the number of spectral bands after PCA [24]. Thus, spectral bands are reduced, and spectral information is preserved. Using the combined spectral and spatial information, a region size of $S \times S$ centered on the pixel (i, j) is superimposed into X, defined as a spectral–spatial vector

$X_{i,j} = [X_{i,j,1}, \cdots, X_{i,j,B}] \in \mathbb{R}^{S\times S\times B}$. Taking the HSI digital cube $X \in \mathbb{R}^{S\times S\times B}$ as input, the adaptive spectral space kernel feature map $V \in \mathbb{R}^{S\times S\times B}$ is generated as output [16]:

$$V = F_{ASSK}(X; \theta_a) \tag{1}$$

where θ_a is the trainable parameter in ASSK. By automatically adjusting the receptive field size, neurons can jointly learn spectral spatial features and amplify multi-scale information of neurons in the next layer.

In order to enable neurons to adaptively adjust the size of the receptive field, we use selective kernel convolution to learn the selection of the spectral–spatial kernel attention feature maps between different receptive fields through FASSK, as shown in Figure 2. Selective kernel convolutions between multiple kernels have different kernel sizes. The basic idea is to use gates to control the flow of information from two branches carrying information of different scales into the neurons of the next layer. To achieve this, the gate needs to integrate information from branch offices, where multiple branches with different kernel sizes are fused using softmax function attention guided by information in these branches. Different attention to these branches produces different sizes of the effective receptive fields of neurons in the fusion layer. $\hat{F}^{(l+1)}_{spectral} : X^l \rightarrow \hat{U}^{(l+1)} \in \mathbb{R}^{S\times S\times B}$ and $\widetilde{F}^{(l+1)}_{spatial} : X^l \rightarrow \widetilde{U}^{(l+1)} \in \mathbb{R}^{S\times S\times B}$ are the transformations of the $(l+1)^{th}$ layer, where X_l is the input to the $(l+1)^{th}$ layer spectral and spatial kernel selection transformation. The output feature maps $\hat{U}^{(l+1)}$ and $\widetilde{U}^{(l+1)}$ are defined as:

$$\hat{U}^{(l+1)} = \hat{F}^{(l+1)}_{spectral}\left(X^l\right) = X^l * W^{(l+1)}_{(1\times1\times7)} + b^{(l+1)} \tag{2}$$

$$\widetilde{U}^{(l+1)} = \widetilde{F}^{(l+1)}_{spatial}\left(X^l\right) = X^l * W^{(l+1)}_{(3\times3\times7)} + b^{(l+1)} \tag{3}$$

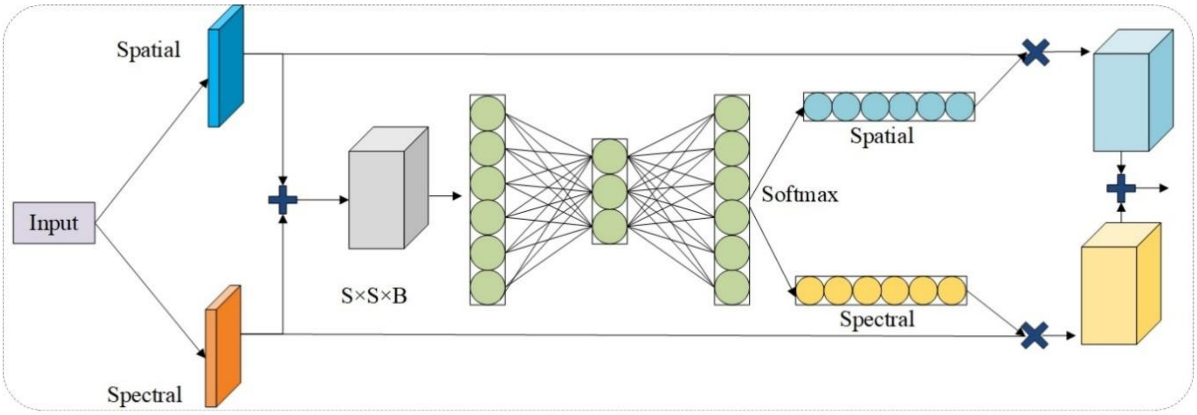


Figure 2. The structure of selective kernel convolution.

Among them, $*$ is the three-dimensional convolution operation, W^{l+1} is the weight of the $(l+1)^{th}$ convolution layer, $b^{(l+1)}$ is the bias, and two three-dimensional convolution kernels with receptive field sizes (1 × 1 × 7) and (3 × 3 × 7) are used to extract the spectral and spatial feature maps. $\hat{F}_{spectral}$ extracts spectral features, and $\hat{F}_{spatial}$ extracts spatial features.

By automatically adjusting the size of the receptive field of neurons, the neurons jointly learn the spectral–spatial features and amplify the multi-scale information flow of the neurons in the next layer. Firstly, element-level summation is used to fuse the results of the two branches:

$$U^{(l+1)} = \widetilde{U}^{(l+1)} + \hat{U}^{(l+1)} \tag{4}$$

Secondly, global information is embedded by using global average pooling (GAP) to generate feature response vectors (FRVs) with channel statistics of the data. Specifically, the spatial dimension of $U^{(l+1)} \in \mathbb{R}^{S\times S\times B}$ is reduced to $s_b^{(l+1)} \in \mathbb{R}^{1\times 1\times B}$ along the b^{th} feature map direction by averaging the spatial elements of $S \times S$ at each channel:

$$s_b^{(l+1)} = \frac{1}{S\times S}\sum_{i=1}^{S}\sum_{j=1}^{S} u_b^{(l+1)}(i,j) \tag{5}$$

Furthermore, to obtain neural activations of different channel features enabling adaptive kernel selection, a compact feature $z^{(l+1)} \in \mathbb{R}^{d\times 1}$ was created to enable guidance for precise and adaptive selection. This is achieved by a simple fully connected layer, which reduces the dimensionality to improve efficiency, and the feature weight vector is defined as:

$$z^{(l+1)} = \mathrm{ReLu}\left(\mathrm{BN}\left(\mathrm{W}^{(l+1)} \cdot s_b^{l+1}\right)\right) \tag{6}$$

ReLu is the activation function, and BN is the batch normalization process. d is used to achieve model convergence, and the compression ratio r is used to control $z^{(l+1)}$ for compressing dimension:

$$d = \max\left(\frac{C}{r}, L\right) \tag{7}$$

where L is the minimum value of d (L = 32 in our experiment).

Guided by the channel descriptor $z^{(l+1)}$, a discriminative spectral-spatial kernel feature map is automatically selected. Specifically, apply $z^{(l+1)}$ to the softmax function:

$$a_{spectral}^{l+1} = \frac{e^{A_b^{(l+1)} z^{(l+1)}}}{e^{A_b^{(l+1)} z^{(l+1)}} + e^{B_b^{(l+1)} z^{(l+1)}}} \tag{8}$$

$$b_{spatial}^{l+1} = \frac{e^{B_b^{(l+1)} z^{(l+1)}}}{e^{A_b^{(l+1)} z^{(l+1)}} + e^{B_b^{(l+1)} z^{(l+1)}}} \tag{9}$$

Among them, $a_{spectral}^{l+1}$ and $b_{spatial}^{l+1}$ denote the soft attention vector for $\hat{U}^{(l+1)}$ and $\widetilde{U}^{(l+1)}$, respectively. $A_b^{(l+1)} \in \mathbb{R}^{1\times d}$ and $B_b^{(l+1)} \in \mathbb{R}^{1\times d}$ are the b^{th} row of $A^{(l+1)} \in \mathbb{R}^{B\times d}$ and $B^{(l+1)} \in \mathbb{R}^{B\times d}$, the final feature map V is obtained through the attention weights on each kernel function:

$$V = a_{spectral}^{l+1} \times \hat{U}^{(l+1)} + b_{spatial}^{l+1} \times \widetilde{U}^{(l+1)} \tag{10}$$

Among them, $a_{spectral}^{l+1} + b_{spatial}^{l+1} = 1$, $V = [V_1, V_2, \ldots, V_B]$ and $V_i \in \mathbb{R}^{S\times S}, \forall i = 1, \ldots, B$.

The kernel feature map is made up of four ResBlocks in order to extract more robust and discriminative spectral–spatial characteristics. Each ResBlock is made up of 24 kernels that are separated into spectral characteristics based on the learning of distinct kernel shapes and spatial features. The first two ResBlocks extract spatially focused spectral characteristics, whereas the latter two extract spatially focused spectral features. As a result, combining spectral and spatial data increase the model's identification capabilities. A GAP layer is utilized after re-blocking to transform 3D feature maps of size 7 × 7 × 24 into feature vectors of size 1 × 1 × 24.

Efficient Feature Recalibration is recalibrated by residual and spectral spatial channels. Among them, $F_{EFR}(\cdot)$ takes the transformed feature map of the l^{th} layer $X^l \in \mathbb{R}^{S\times S\times B}$ as the input, and generates the feature map recalibrated by the channel $\hat{X}^{l+1} \in \mathbb{R}^{S\times S\times B}$ as the output, that is:

$$\hat{X}^{l+1} = F_{EFR}\left(X^l; \theta_b\right) \tag{11}$$

where θ_b is the trainable parameter in the EFR module.

2.2. Improved Vision Transformer

Transformer networks were developed to simulate long-term relationships between sequence parts in machine translation. Although Transformer's self-attention mechanism can mimic the global interaction between token embeddings, it lacks a local method for information sharing among small areas. We provide a locality mechanism to ViT by incorporating depth-wise convolution because locality is critical for HSI pictures. The upgraded ViT blends global relation modeling's attention mechanism with local information aggregation's locality mechanism. Locality is added to the ViT by introducing depthwise convolutions in a feedforward network, and the Re-Attention mechanism based on the original ViT is used to increase the diversity of attention maps at different levels.

When compared to standard convolution, depthwise convolution uses just channels for calculation. That is, just one input feature map is convolved to obtain one channel of the output feature map. As a result, depth-wise convolution is both parameter and computation efficient. The patch is input to the Embedding layer, that is, the Linear Projection of Flattened Patches in the Figure 1, a lot of vectors called tokens can be obtained. Then, a new token is added in front of a series of tokens, and Positional Encoding will be added to the patch embedding to retain the position. The closer the information is located, the more similarly it tends to be encoded. In addition, the location information needs to be added, corresponding to $0 \sim n$. Then, it is input into the transformer encoder to repeatedly stack the block N times. The output of the transformer is classified by the MLP Head which consists of LayerNorm and two fully connected layers, and the GELU activation function is used for classification to obtain the final classification result [24].

Figure 3 depicts the transformer encoder, which consists of N stacks of the same layer. Each layer consists of the re-attention mechanism and position-wise fully connected feed-forward network. Around each of these two sublayers, we utilize a residual connection [25] and a normalization layer [26]. That is, $\text{LayerNorm}(x + \text{Sublayer}(x))$ is the output of each sublayer, where $\text{Sublayer}(x)$ is an implementation function of the sublayer.

1. Re-Attention

Re-Attention successfully overcomes the problem of attention collapse and allows for more in-depth ViT training, which collects complementing information from multiple attention heads through interactions to promote the variety of attention maps. Specifically, we use dynamic aggregation to create a new set of attention maps based on the head's attention maps. A learnable transition matrix $\Theta \in \mathbb{R}^{H \times H}$ is defined and used to combine the multi-head attention maps into a new regenerated map before multiplying by V. Re-Attention is accomplished by the following manner [27]:

$$\text{Re-Attention}(Q, K, V) = \text{Norm}\left(\Theta^{\mathrm{T}}\left(\text{softmax}\left(\frac{QK^{\mathrm{T}}}{\sqrt{d}}\right)\right)\right)V \tag{12}$$

The transformation matrix Θ is multiplied by the self-attention map of the head dimension. Norm is the normalizing function used to decrease hierarchical variance. The softmax function is applied to rows of comparable matrices, and d is used to normalize the result. The three learnable weight matrices include query (Q), key (K), and value (V). Relationships between tokens are modeled by projecting the similarity between query key pairs, resulting in attention scores.

2. Feed forward

After the Re-Attention layer, a feedforward network is attached. A token sequence is first reshaped into a feature map on a 2D lattice. Then, two 1 × 1 convolutions and one depthwise convolution are applied to the feature map. Then, the feature map is reshaped into a sequence of tokens, which is used as self-attention in the transformer layer of the network. The specific description is as follows.

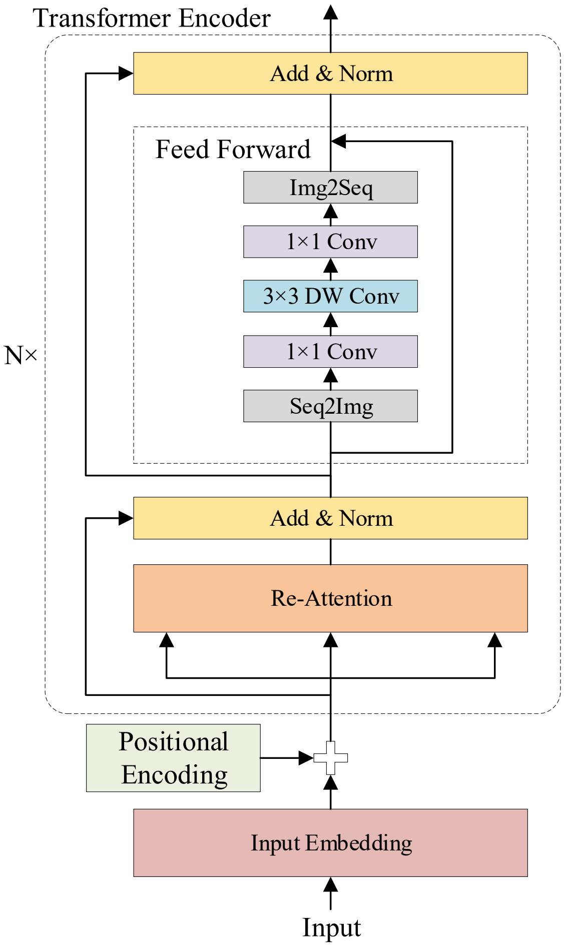


Figure 3. Transformer encoder.

The feedforward network consists of two input convolutions of size 1 × 1 and transforms features along the embedding dimension. The hidden dimension between the two convolutional layers is expanded to learn richer feature representations. Since the feedforward network is applied to the sequence of tokens $Z \in \mathbb{R}^{N \times d}$ by position, the reshaped features of the sequence of tokens are represented as:

$$Z^r = \text{Seq2Img}(Z), Z^r \in \mathbb{R}^{h \times w \times d} \tag{13}$$

The sequence is converted into a 2D feature map using Seq2Img. To re-establish token closeness, each token is placed at the pixel position of the feature map, offering a chance to reinstate locality into the network.

There is no information exchange between neighboring pixels since the feature map just performs the 1 × 1 convolution. Furthermore, the transformer's attention section only captures the global interdependence between all tokens. In the inverted residual block, there is a depthwise convolution. Each channel is given k × k (k > 1) convolution kernels by depthwise convolution. To calculate a new feature, the features from k × k kernels are combined. Therefore, depthwise convolution is a good approach to bring locality into the network. The depthwise convolution is introduced into the transformer feedforward network, and the calculation formula is [23]:

$$Y^r = f(f(Z^r * W_1^r) * W_d) * W_2^r \tag{14}$$

$$Y = \text{Img2Seq}(Y^r) \tag{15}$$

$f(\cdot)$ is the nonlinear activation function. The bias phrase has been eliminated for clarity. In most cases, the dimensional expansion ratio γ is set to 4. $W_1^r \in \mathbb{R}^{d\times\gamma d\times 1\times 1}$ is reshaped from W_1 and represents the convolution kernel. $W_d \in \mathbb{R}^{\gamma d\times 1\times k\times k}$ is the kernel of depthwise convolution. The Img2Seq function returns the image feature map to a series of tokens which is used in the following self-attention layer.

2.3. Apollo Optimizer

The optimizer is used to update and compute network parameters that influence model training and output in order to approach or attain the optimal value, reducing (or maximizing) the loss function. This work employs Apollo, a non-convex quasi-Newtonian stochastic optimization technique that is both simple and computationally efficient. This approach is useful for large-scale optimization issues involving big data sets or high-dimensional parameter spaces, such as deep neural network machine learning, and using the Apollo optimizer improves HSI data categorization accuracy. The method approximates the Hessian through a diagonal matrix, dynamically introduces the curvature of the loss function, and the update and storage of the Hessian diagonal is as efficient as the adaptive first-order optimization method of linear complexity. The Hessian is replaced with its adjusted absolute value to handle non-convexity and ensure that it is positive definite.

The Apollo optimizer formula is as follows [28]:

$$\theta_{t+1} = \theta_t - H_t^{-1} g_t \tag{16}$$

where $g_t = \nabla f(\theta_t)$ is the gradient at θ_t, and $H_t = \nabla^2 f(\theta_t)$ is the Hessian matrix:

$$\theta_{t+1} = \theta_t - \eta_t B_t^{-1} g_t \tag{17}$$

where η_t is the step size, and B_t is the approximation of the Hessian matrix for each parameter update. Exponential moving averages (EMVs) are applied to g_t, and bias correction is initialized:

$$m_{t+1} = \frac{\beta(1-\beta^t)}{1-\beta^{t+1}} m_t + \frac{1-\beta}{1-\beta^{t+1}} g_{t+1} \tag{18}$$

where $0 < \beta < 1$ is the decay rate of the EMV. For each parameter B_t, the update formula is as follows:

$$\Lambda \triangleq B_{t+1} - B_t = \frac{s_t^T y_t - s_t^T B_t s_t}{\|s_t\|_4^4} \mathrm{Diag}\left(s_t^2\right) \tag{19}$$

Among them, $y_t = g_{t+1} - g_t$, $s_t = \theta_{t+1} - \theta_t$, and s_t^2 is the element-wise squared vector of s_t, $\mathrm{Diag}(s_t^2)$ is a diagonal matrix consisting of the vector's diagonal elements s_t^2, and $\|\cdot\|_4$ is the 4-norm of the vector.

Replace the step size bias in the stochastic gradient g_t with the modified gradient $g_t{}' = \eta_t g_t$. Combined with the corresponding corrected $y_t{}' = g_{t+1}{}' - g_t{}' = \eta_t y_t$, modify the update term Λ in formula (19) and replace y_t with $y_t{}'$:

$$\Lambda' = \frac{s_t^T y_t{}' - s_t^T B_t s_t}{\|s_t\|_4^4} \mathrm{Diag}\left(s_t^2\right) = -\frac{d_t^T y_t + d_t^T B_t d_t}{\|d_t\|_4^4} \mathrm{Diag}\left(d_t^2\right) \tag{20}$$

where $d_t = -\frac{s_t}{\eta_t} = B_t^{-1} g_t$ is the updated direction after correction.

When calculating the update direction with B_t as preprocessing, we use its absolute value:

$$|B_t| = \sqrt{B_t^T B_t} \tag{21}$$

where $\sqrt{\cdot}$ is the positive definite square root of the matrix. Apollo uses a diagonal matrix to represent B_t. In order to deal with the non-convexity of the objective function, the absolute value of B_t is corrected with the convexity hyperparameter σ:

$$D_t = rectify(B_t, \sigma) = \max(|B_t|, \sigma) \tag{22}$$

Among them, the rectify $(\cdot, \sigma)$ function is similar to the corrected linear unit (ReLu), and the threshold is set to σ.

2.4. Focal Loss

Various samples have the same amount of loss in cross-entropy measures prediction; however, in the real HSI classification job, the quantity of samples of different categories varies substantially, as does the classification difficulty of the same category of samples. The categorization complexity of distinct samples varies according to the differences between them. If the same weight is utilized to optimize each instance's prediction results, the prediction results for difficult-to-classify data will be relatively bad. Furthermore, the categorization findings of certain instances are not optimal due to the effect of mixed pixels. The model must adaptively alter the proportion of each instance in the loss according to the classification difficulty in order to enhance classification performance and pay attention to small-class samples and difficult-to-classify samples at the same time. More "optimization resources" should be allocated to challenging classification samples [29]. Focal Loss [30], an improved variant of cross entropy loss, is used in this research, which is defined as:

$$\mathrm{CE}(p, y) = -\frac{1}{N}\sum_{i=0}^{N-1}\sum_{k=0}^{K-1} y_{i,k} \log p_{i,k} \tag{23}$$

Assuming there are K label values, y is the real label, $p_{i,k}$ is the probability of predicting the k^{th} label value for the i^{th} sample, and N represents the number of samples. A common way to address class imbalance is to introduce a weighting factor $\alpha \in \mathbb{R}^{1\times N}$, which is between $[0, 1]$:

$$\mathrm{CE}(p, y) = -\frac{1}{N}\sum_{i=0}^{N-1}\sum_{k=0}^{K-1} \alpha_i y_{i,k} \log p_{i,k} \tag{24}$$

A more formal approach is to add a tuning factor $(1 - p_t)^\gamma$ to the cross-entropy loss function, with tunable focusing parameter $\gamma \geq 0$.

$$\mathrm{CE}(p, y) = -\frac{1}{N}\sum_{i=0}^{N-1}\sum_{k=0}^{K-1} (1 - p_{i,k})^\gamma y_{i,k} \log p_{i,k} \tag{25}$$

Combining the above two formulas, the Focal Loss is obtained:

$$\mathrm{FL}(p, y) = -\frac{1}{N}\sum_{i=0}^{N-1}\sum_{k=0}^{K-1} \alpha_i (1 - p_{i,k})^\gamma y_{i,k} \log(p_{i,k}) \tag{26}$$

3. Experimental Results

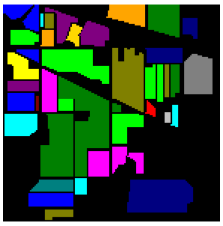
The experiments were carried out on the Windows 10 operating system, and the classification methods were implemented using the Python language and PyTorch library. The experimental environment is an Intel(R) Core(TM) i7-10750H CPU @ 2.60 GHz 2.59 GHz processor, 16 GB memory, and a GeForce GTX 1650Ti graphics card. In order to minimize the experimental error and chance, all the experimental data in this paper are the average results of 10 iterations. In order to adapt to hardware resources and reduce the amount of computation per batch during network training, the size of the input data is set to 32 × 32. All experimental networks can reach a stable convergence state after training up to 100 epochs. In order to ensure that all methods can achieve the best classification effect, this paper sets the maximum number of training epochs to 200 and adopts the early stopping method to avoid the overfitting problem. We use the Apollo optimizer to learn the mixing operation parameters, where the learning rate is set to 0.0004. Three indicators of comprehensive accuracy (OA), average accuracy (AA) and the Kappa coefficient (K) are used to quantitatively evaluate the experimental results.

3.1. Hyperspectral Datasets Description

In this study, we conduct experiments on four different HSI datasets, including Indian Pines datasets, Pavia University datasets, Xuzhou datasets and WHU-Hi-LongKou datasets. The datasets used are described in detail below. The number of samples per class, a false color map and a ground truth map of the datasets are shown in Tables 1–4.

1. Data in the Indian Pines dataset were obtained by the AVIRIS sensor over the Indian Pines Agricultural Proving Ground in northwestern Indiana, USA. The original data have a total of 224 bands, 4 zero bands and 20 water absorption bands (104–108, 150–163 and 220) are removed, and the remaining 200 bands are for experimental study, ranging from 0.4 to 2.5 µm. the space size is 145 × 145 pixels with 16 different types of plants.

Table 1. Indian Pines Dataset Labeled Sample Counts.

No.	Class	Color	Sample Numbers	False-Color Map	Ground-Truth Map
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		
Total			10,249		

2. Data in the Pavia University dataset were obtained by ROSIS-03 sensors over the University of Pavia, Pavia, Italy. The size of the dataset is 610 × 340 pixels, and the spatial resolution is 1.3 m. The original data have 115 bands with spectral coverage ranging from 0.43 to 0.86 µm. Twelve noise bands are removed, and the remaining 103 bands are available for experiments with 9 categories.

3. Data in the Xuzhou dataset [31,32] were acquired by HySpex SWIR-384 and HySpex VNIR-1600 imaging spectrometers in Xuzhou, Jiangsu Province, China, in November 2014, and the experimental area is located near a coal mining area. The size of the dataset is 500 × 260 pixels, the noise bands from 415 to 2508 nm are removed, and there are 436 bands for experiments with 9 categories.

Table 2. Pavia University Dataset Labeled Sample Counts.

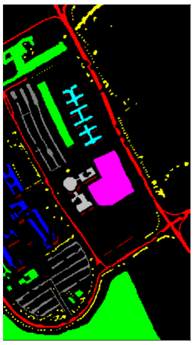
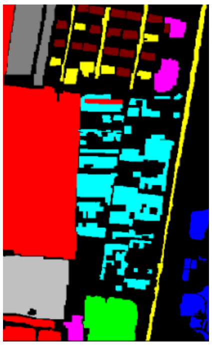
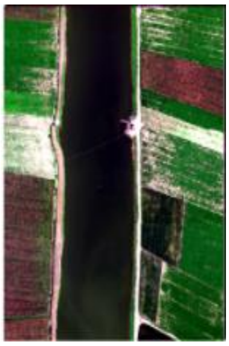
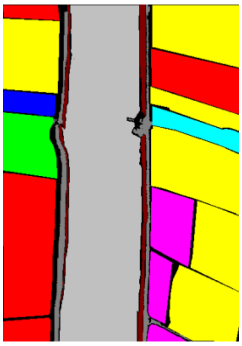
No.	Class	Color	Sample Numbers	False-Color Map	Ground-Truth Map
1	Asphalt	🟥	6631		
2	Meadows	🟩	18,649		
3	Gravel	🟦	2099		
4	Trees	🟨	3064		
5	Painted metal sheets	🟦	1345		
6	Bare Soil	🟪	5029		
7	Bitumen	⬜	1330		
8	Self-Blocking Bricks	⬛	3682		
9	Shadows	🟫	947		
	Total		42,776		

Table 3. Xuzhou Dataset Labeled Sample Counts.

No.	Class	Color	Sample Numbers	False-Color Map	Ground-Truth Map
1	Bareland-1	🟥	26,396		
2	Lakes	🟩	4027		
3	Coals	🟦	2783		
4	Cement	🟨	5214		
5	Crops-1	🟦	13,184		
6	Trees	🟪	2436		
7	Bareland-2	⬜	6990		
8	Crops-2	⬛	4777		
9	Red-tiles	🟫	3070		
	Total		68,877		

4. The WHU-Hi-LongKou dataset [33,34] consists of an 8 mm focal length head-wall nano-Hyperspec imaging sensor, mounted on the DJI Matrice 600 Pro (DJI M600 Pro) drone platform in 2018 Obtained in Longkou Town, Hubei Province, China, on 7 July 2008. The study area is a simple agricultural scenario containing six crops: corn, cotton, sesame, broad-leaf soybean, narrow-leaf soybean, and rice, with a total of nine categories. The image size is 550 × 400 pixels with 270 bands between 0.4~1 µm, and the spatial resolution of the hyperspectral image carried by the UAV is about 0.463 m.

Table 4. WHU-Hi-LongKou Dataset Labeled Sample Counts.

No.	Class	Color	Sample Numbers	False-Color Map	Ground-Truth Map
1	Corn	🟥	34,511		
2	Cotton	🟩	8374		
3	Sesame	🟦	3031		
4	Broad-leaf soybean	🟨	63,212		
5	Narrow-leaf soybean	🟦	4151		
6	Rice	🟪	11,854		
7	Water	⬜	67,056		
8	Roads and houses	⬛	7124		
9	Mixed weed	🟫	5229		
	Total		204,542		

3.2. Comparison of the Proposed Methods with the State-of-the-Art Methods

In this section, to evaluate the classification performance of our proposed method, the proposed method is validated by using several comparative experiments, including the traditional method RBF-SVM [35] and the deep-learning-related methods CNN [36], HybirdSN [14], PyResNet [37], SSRN [15], SSFTT [38] and A2S2KResNet [16]. For RBF-SVM, the radial basis function is used as the kernel, and the grid search method is used to find the exponential growing sequence. In each dataset, the number of training samples is 10% of the total number of samples. The experimental results of the proposed method are shown in Tables 5–8. It can be seen that the OA, AA and Kappa values achieved by the proposed method are the best, with OA reaching 98.81%, 99.76%, 99.80% and 99.89% on the Indian Pines, Pavia University, Xuzhou and WHU-Hi-LongKou datasets, respectively.

Table 5. Classification results of all methods on Indian Pines dataset.

Method Class	RBF-SVM	CNN	HybirdSN	PyResNet	SSRN	SSFTT	A2S2KResNet	Proposed
1	65.07 ± 9.65	81.93 ± 7.74	82.99 ± 25.02	91.31 ± 3.43	85.38 ± 2.10	99.76 ± 0.73	91.62 ± 2.05	98.70 ± 3.16
2	71.34 ± 2.01	74.70 ± 9.07	89.21 ± 4.71	87.74 ± 7.82	98.07 ± 1.64	94.85 ± 1.15	98.67 ± 1.05	98.57 ± 1.13
3	75.53 ± 2.58	59.56 ± 6.96	89.13 ± 3.73	80.27 ± 8.06	97.09 ± 1.29	99.24 ± 0.47	98.81 ± 0.65	98.96 ± 0.67
4	61.18 ± 6.84	45.37 ± 5.60	89.50 ± 6.70	82.60 ± 5.87	97.95 ± 1.97	99.30 ± 1.08	99.37 ± 0.87	98.87 ± 1.00
5	88.76 ± 2.97	89.37 ± 4.67	96.73 ± 4.55	96.82 ± 3.32	96.68 ± 1.29	98.78 ± 0.94	98.97 ± 0.80	98.77 ± 0.83
6	89.16 ± 1.77	94.88 ± 3.44	97.49 ± 2.12	92.57 ± 5.47	97.46 ± 4.53	99.37 ± 0.39	98.86 ± 1.01	99.35 ± 0.67
7	85.05 ± 9.27	81.91 ± 13.10	82.27 ± 9.19	93.94 ± 6.50	70.00 ± 5.83	98.40 ± 3.20	97.93 ± 6.21	97.02 ± 7.70
8	90.32 ± 1.50	97.83 ± 1.22	95.70 ± 3.99	92.27 ± 3.57	98.50 ± 1.97	99.79 ± 0.45	100.00 ± 0.00	99.92 ± 0.12
9	71.14 ± 13.65	57.26 ± 12.84	70.23 ± 9.42	92.85 ± 1.53	74.27 ± 10.90	67.22 ± 7.29	81.70 ± 10.54	96.02 ± 5.45
10	75.74 ± 2.60	68.54 ± 4.98	88.03 ± 5.51	85.21 ± 7.20	96.94 ± 1.54	97.54 ± 0.89	97.90 ± 1.24	97.51 ± 1.09
11	77.97 ± 1.29	88.46 ± 3.55	91.62 ± 2.26	89.28 ± 6.57	99.07 ± 0.61	99.22 ± 0.35	99.16 ± 0.42	99.18 ± 0.50
12	73.24 ± 3.75	64.95 ± 10.02	87.51 ± 5.96	86.97 ± 8.25	98.23 ± 1.62	95.96 ± 1.09	98.33 ± 1.25	98.59 ± 1.41
13	90.80 ± 4.36	98.47 ± 1.05	97.01 ± 3.26	98.36 ± 1.52	98.03 ± 2.21	98.86 ± 0.71	99.17 ± 1.15	99.46 ± 0.86
14	91.74 ± 0.88	98.20 ± 0.35	97.76 ± 0.91	94.17 ± 4.02	99.27 ± 0.65	99.38 ± 0.88	99.25 ± 0.51	99.15 ± 0.60
15	74.41 ± 6.25	53.50 ± 5.01	94.66 ± 3.60	91.22 ± 3.92	98.85 ± 1.10	98.01 ± 1.21	98.58 ± 1.06	98.65 ± 1.13
16	98.16 ± 2.27	93.67 ± 4.80	92.23 ± 7.19	95.80 ± 4.88	88.71 ± 14.42	91.69 ± 5.79	94.46 ± 4.95	94.33 ± 3.67
OA (%)	80.01 ± 0.66	78.31 ± 2.82	92.08 ± 1.72	87.27 ± 4.82	97.97 ± 0.58	98.07 ± 0.39	98.51 ± 0.26	98.81 ± 0.32
AA (%)	79.35 ± 2.40	78.04 ± 1.78	90.13 ± 5.38	90.71 ± 3.47	86.72 ± 7.45	96.08 ± 1.44	97.05 ± 1.35	98.83 ± 0.63
K × 100	77.09 ± 0.77	75.04 ± 3.11	90.96 ± 1.97	85.49 ± 5.37	97.69 ± 0.67	97.85 ± 0.50	98.58 ± 0.30	98.65 ± 0.36

Table 6. Classification results of all methods on Pavia University dataset.

Method Class	RBF-SVM	CNN	HybirdSN	PyResNet	SSRN	SSFTT	A2S2KResNet	Proposed
1	81.26 ± 5.08	96.14 ± 1.60	87.42 ± 10.30	94.44 ± 3.69	99.40 ± 1.31	99.72 ± 0.21	98.95 ± 1.52	99.72 ± 0.14
2	84.53 ± 3.81	96.67 ± 0.99	99.57 ± 0.28	96.49 ± 2.42	99.97 ± 0.03	99.98 ± 0.01	99.98 ± 0.03	99.98 ± 0.02
3	56.56 ± 16.17	73.84 ± 11.76	72.11 ± 16.92	88.58 ± 11.05	98.96 ± 2.16	98.97 ± 0.85	99.05 ± 1.40	99.34 ± 0.85
4	94.34 ± 3.50	75.32 ± 13.62	81.59 ± 11.63	98.86 ± 1.67	99.82 ± 0.26	98.70 ± 0.42	99.66 ± 0.72	99.71 ± 0.22
5	95.38 ± 3.40	99.75 ± 0.18	79.47 ± 8.35	98.77 ± 1.83	99.89 ± 0.13	99.85 ± 0.26	99.94 ± 0.11	99.76 ± 0.25
6	80.66 ± 7.54	79.30 ± 5.72	98.88 ± 0.75	92.77 ± 7.60	99.97 ± 0.04	99.99 ± 0.01	99.92 ± 0.09	99.85 ± 0.22
7	69.13 ± 11.04	68.22 ± 14.45	72.33 ± 15.73	95.61 ± 4.39	99.98 ± 0.06	99.55 ± 0.38	99.90 ± 0.31	99.53 ± 0.52
8	71.16 ± 6.24	80.58 ± 3.89	78.22 ± 9.17	89.20 ± 3.55	98.72 ± 0.87	99.02 ± 0.63	98.72 ± 0.83	99.13 ± 0.60
9	99.94 ± 0.07	97.25 ± 5.10	66.95 ± 16.17	99.12 ± 0.55	99.87 ± 0.18	96.63 ± 1.33	99.95 ± 0.09	99.31 ± 0.54
OA(%)	82.06 ± 2.78	87.95 ± 3.47	91.71 ± 8.31	93.80 ± 5.35	99.70 ± 0.32	99.62 ± 0.07	99.62 ± 0.33	99.76 ± 0.06
AA(%)	79.22 ± 5.87	85.23 ± 4.22	76.28 ± 2.29	94.87 ± 1.89	99.62 ± 0.35	99.15 ± 0.19	99.56 ± 0.34	99.60 ± 0.09
K × 100	75.44 ± 4.26	84.19 ± 4.28	88.83 ± 11.41	91.70 ± 7.23	99.61 ± 0.42	99.50 ± 0.10	99.50 ± 0.44	99.69 ± 0.08

Table 7. Classification results of all methods on Xuzhou dataset.

Method Class	RBF-SVM	CNN	HybirdSN	PyResNet	SSRN	SSFTT	A2S2KResNet	Proposed
1	96.38 ± 0.32	97.16 ± 1.20	99.36 ± 0.25	94.92 ± 1.27	99.83 ± 0.09	99.59 ± 0.21	99.47 ± 0.51	99.84 ± 0.03
2	99.81 ± 0.15	99.06 ± 0.89	99.49 ± 0.46	99.99 ± 1.37	99.99 ± 0.01	99.98 ± 0.03	100.00 ± 0.00	99.98 ± 0.03
3	93.71 ± 0.69	87.15 ± 2.51	96.94 ± 1.41	95.55 ± 4.04	99.71 ± 0.16	99.54 ± 0.25	99.16 ± 0.58	99.49 ± 0.22
4	97.31 ± 0.47	85.76 ± 8.60	98.39 ± 0.61	93.85 ± 1.24	99.70 ± 0.54	99.92 ± 0.08	99.86 ± 0.02	99.94 ± 0.05
5	94.64 ± 0.49	94.03 ± 1.37	98.92 ± 0.35	97.34 ± 4.50	99.52 ± 0.36	99.56 ± 0.20	99.64 ± 0.29	99.75 ± 0.10
6	88.72 ± 1.11	62.30 ± 3.70	96.47 ± 0.99	84.85 ± 2.88	99.20 ± 0.49	99.64 ± 0.17	99.03 ± 0.37	99.46 ± 0.22
7	87.00 ± 0.82	73.31 ± 5.18	98.43 ± 0.56	88.63 ± 2.96	99.59 ± 0.42	99.90 ± 0.05	99.64 ± 0.13	99.77 ± 0.14
8	98.18 ± 0.27	93.97 ± 2.32	99.28 ± 0.44	98.36 ± 2.55	99.77 ± 0.15	99.94 ± 0.13	99.70 ± 0.13	99.77 ± 0.23
9	97.67 ± 0.61	98.63 ± 0.36	99.38 ± 0.35	98.58 ± 2.81	99.89 ± 0.12	99.62 ± 0.16	99.88 ± 0.17	99.88 ± 0.12
OA (%)	95.16 ± 0.13	90.56 ± 1.10	98.91 ± 0.16	95.44 ± 0.09	99.71 ± 0.13	99.68 ± 0.06	99.58 ± 0.23	99.80 ± 0.02
AA (%)	94.82 ± 0.20	87.93 ± 1.16	98.52 ± 0.29	94.67 ± 0.11	99.69 ± 0.16	99.72 ± 0.04	99.60 ± 0.18	99.77 ± 0.04
K × 100	93.84 ± 0.16	88.04 ± 1.36	98.61 ± 0.21	93.89 ± 1.13	99.64 ± 0.17	99.60 ± 0.09	99.47 ± 0.30	99.75 ± 0.02

Table 8. Classification results of all methods on WHU-Hi-LongKou dataset.

Method / Class	RBF-SVM	CNN	HybirdSN	PyResNet	SSRN	SSFTT	A2S2KResNet	Proposed
1	99.52 ± 0.68	96.74 ± 1.64	89.91 ± 9.97	99.97 ± 0.02	99.97 ± 0.04	99.96 ± 0.02	99.92 ± 0.12	99.96 ± 0.04
2	94.13 ± 3.99	71.28 ± 3.89	80.25 ± 10.79	99.70 ± 0.07	99.79 ± 0.15	99.92 ± 0.08	99.79 ± 0.24	99.82 ± 0.12
3	99.09 ± 0.29	42.95 ± 2.54	78.29 ± 15.58	99.81 ± 0.10	99.91 ± 0.15	99.93 ± 0.14	99.98 ± 0.03	99.99 ± 0.02
4	98.87 ± 0.09	98.19 ± 0.46	89.79 ± 9.93	99.84 ± 0.06	99.84 ± 0.06	99.89 ± 0.04	99.86 ± 0.22	99.95 ± 0.03
5	92.46 ± 0.09	54.82 ± 2.50	72.38 ± 11.83	99.21 ± 0.40	99.32 ± 0.53	99.56 ± 0.30	96.43 ± 9.03	99.42 ± 0.29
6	99.76 ± 0.89	92.92 ± 3.53	87.66 ± 11.52	99.84 ± 0.06	99.96 ± 0.04	99.86 ± 0.13	99.17 ± 2.36	99.92 ± 0.07
7	99.98 ± 0.01	99.89 ± 0.10	93.20 ± 0.14	99.98 ± 0.01	99.99 ± 0.01	99.97 ± 0.01	99.98 ± 0.01	99.98 ± 0.02
8	97.40 ± 0.34	83.06 ± 9.43	78.82 ± 10.47	95.11 ± 0.39	98.92 ± 0.06	98.41 ± 0.53	98.45 ± 1.21	98.92 ± 0.55
9	97.63 ± 0.50	66.60 ± 4.38	74.76 ± 6.03	98.91 ± 0.31	98.30 ± 0.08	98.07 ± 0.69	99.03 ± 0.48	98.32 ± 0.89
OA (%)	98.95 ± 0.03	91.42 ± 0.94	90.88 ± 9.58	99.63 ± 0.57	99.83 ± 0.03	99.81 ± 0.02	99.69 ± 0.47	99.89 ± 0.03
AA (%)	97.65 ± 0.01	77.16 ± 1.57	81.67 ± 7.95	99.15 ± 1.14	99.55 ± 0.11	99.50 ± 0.08	99.18 ± 1.28	99.67 ± 0.11
K × 100	98.68 ± 0.04	88.91 ± 1.18	86.85 ± 9.20	99.52 ± 0.75	99.78 ± 0.04	99.76 ± 0.03	99.60 ± 0.62	99.86 ± 0.04

To show the classification results more clearly, we present the classification results of eight methods on the four hyperspectral datasets, as shown in Figures 4–7. Obviously, our proposed method has more accurate classification results compared to other methods. Compared with deep-learning-based methods on the four datasets, there are more noise scatters in the classification graph of EMP-SVM, CNN, Hybird, PyResNet, SSRN, SSFTT, and A2S2KResNet classification methods still have some misclassifications. Compared with ground truth, it can be seen that the proposed method can obtain more accurate classification results, which further proves the effectiveness of the proposed method in the classification of hyperspectral data.

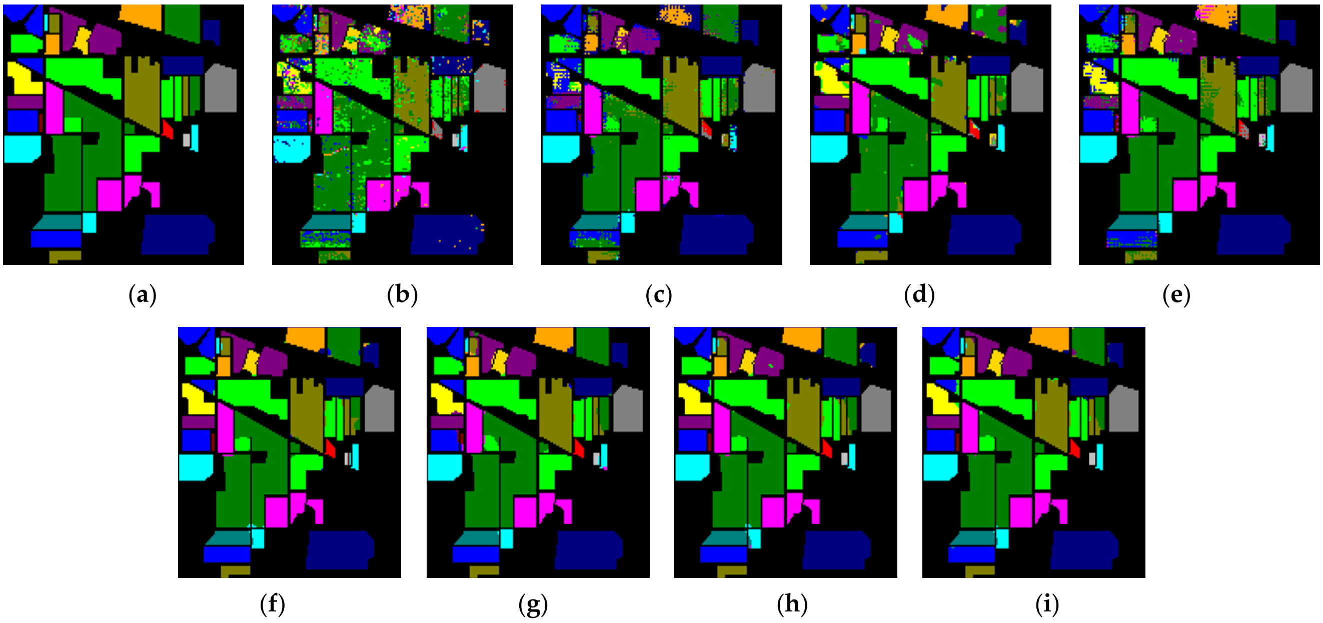


Figure 4. The classification results of Indian Pines dataset. (**a**) Ground-truth map; (**b**) RBF-SVM; (**c**) CNN; (**d**) HybirdSN; (**e**) PyResNet; (**f**) SSRN; (**g**) SSFTT; (**h**) A2S2KResNet; (**i**) Proposed.

3.3. Ablation Experiments

Among them, we performed ablation experiments on the Indian Pines dataset to verify the effectiveness of the proposed method. The experimental results are shown in Table 9.

When we only use the A2S2kResNet model to classify hyperspectral data, its OA on Indian pines dataset is only 98.51%. When A2S2KResNet is combined with ViT (A2S2KResNet + ViT), the OA is 98.61%, which proves that the ViT model can slightly improve the classification performance of the model. When the A2S2KResNet + ViT model is combined with Focal Loss function or Apollo optimizer, the OAs are 98.63% and 98.75%, respectively. It is proven that Focal Loss function and Apollo optimizer are slightly helpful to A2S2kResNet + ViT model. When A2S2KResNet + ViT is combined with Focal Loss function and Apollo optimizer, which is the HSI classification model proposed by us in this paper, it achieves the highest classification accuracy on Indian Pines dataset, which further proves the effectiveness of our method in improving the classification performance of HSI.

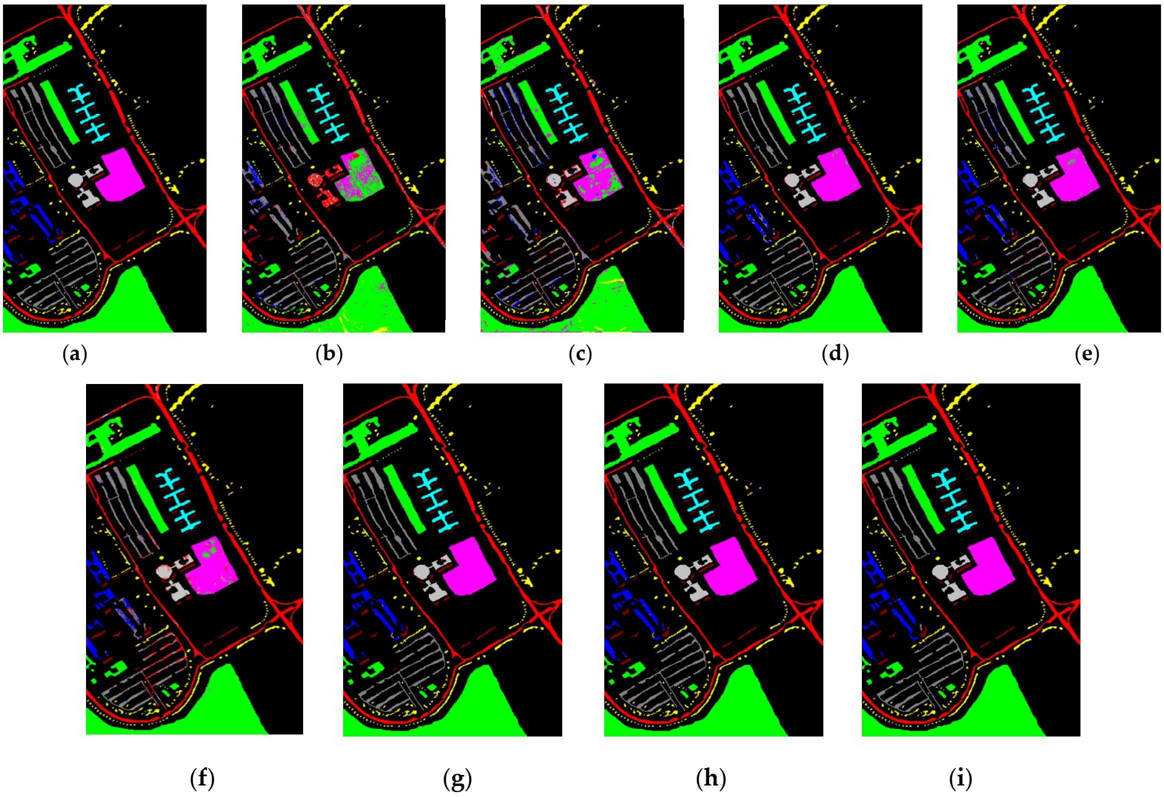


Figure 5. The classification results of Pavia University dataset. (**a**) Ground-truth map; (**b**) RBF-SVM; (**c**) CNN; (**d**) HybirdSN; (**e**) PyResNet; (**f**) SSRN; (**g**) SSFTT; (**h**) A2S2KResNet; (**i**) Proposed.

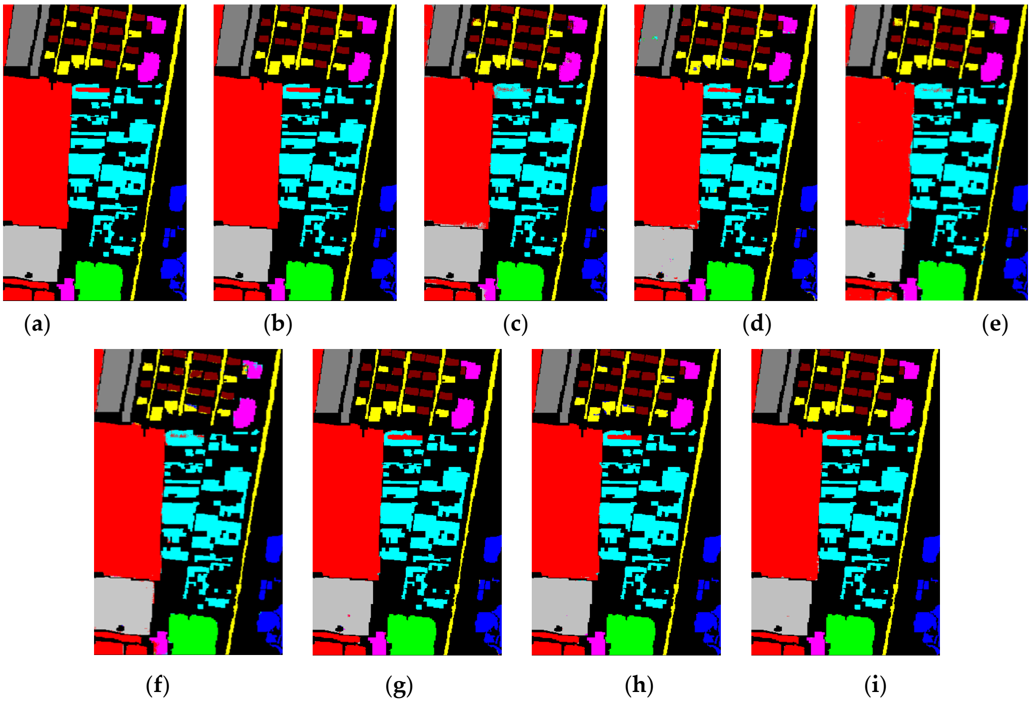


Figure 6. The classification results of Xuzhou dataset. (**a**) Ground-truth map; (**b**) RBF-SVM; (**c**) CNN; (**d**) HybirdSN; (**e**) PyResNet; (**f**) SSRN; (**g**) SSFTT; (**h**) A2S2KResNet; (**i**) Proposed.

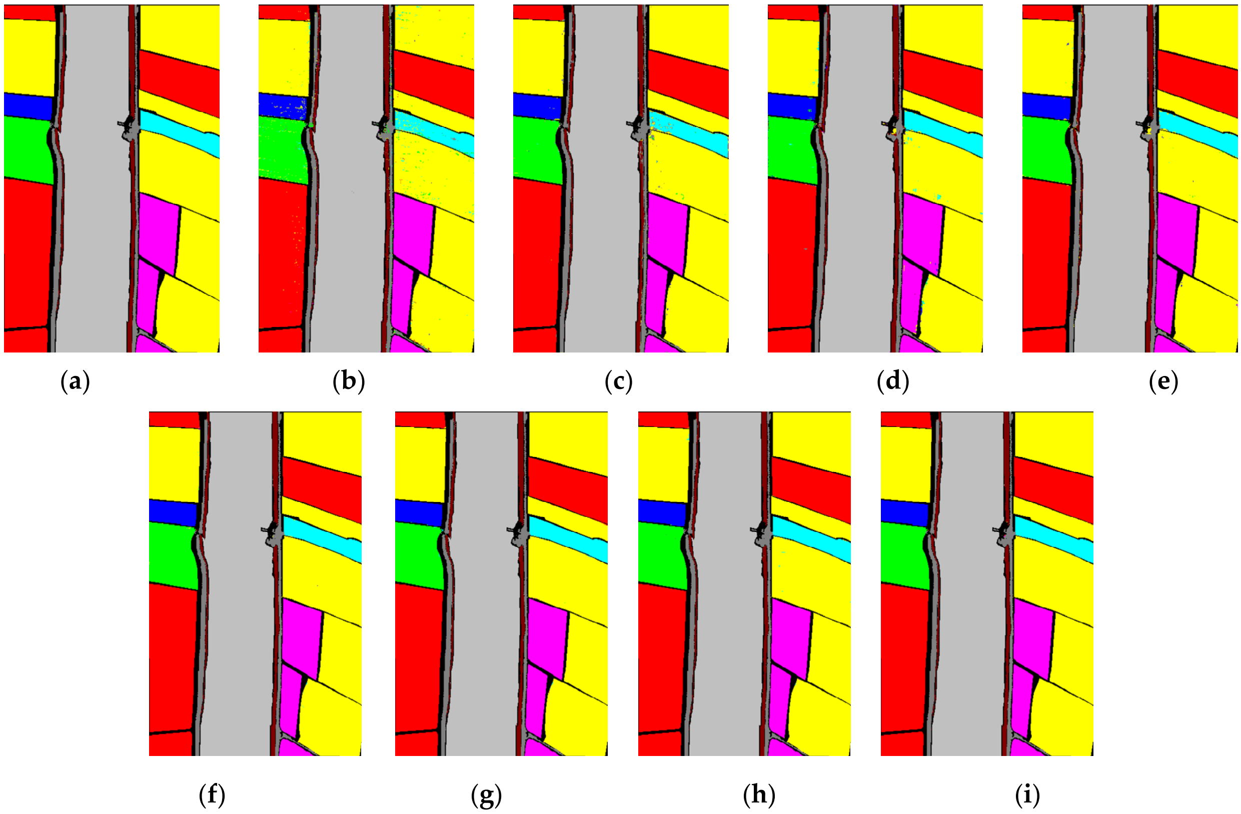


Figure 7. The classification results of WHU-Hi-LongKou dataset. (**a**) Ground-truth map; (**b**) RBF-SVM; (**c**) CNN; (**d**) HybirdSN; (**e**) PyResNet; (**f**) SSRN; (**g**) SSFTT; (**h**) A2S2KResNet; (**i**) Proposed.

Table 9. Comparison of ablation experiments on Indian Pines dataset.

Support Surface	A2S2KResNet	A2S2KResNet + ViT	A2S2KResNet + ViT + Loss	A2S2KResNet + ViT + Apollo	Proposed
1	91.62 ± 2.05	98.21 ± 3.30	98.74 ± 2.20	97.66 ± 3.43	98.70 ± 3.16
2	98.67 ± 1.05	98.61 ± 1.02	98.60 ± 0.67	98.36 ± 7.82	98.57 ± 1.13
3	98.81 ± 0.65	98.52 ± 1.24	98.97 ± 0.86	98.74 ± 8.06	98.96 ± 0.67
4	99.37 ± 0.87	98.97 ± 1.23	97.62 ± 2.16	98.71 ± 5.87	98.87 ± 1.00
5	98.97 ± 0.80	98.75 ± 1.58	98.29 ± 1.37	99.00 ± 3.32	98.77 ± 0.83
6	98.86 ± 1.01	98.87 ± 0.71	99.04 ± 0.77	99.13 ± 5.47	99.35 ± 0.67
7	97.93 ± 6.21	94.65 ± 7.23	96.91 ± 5.59	93.24 ± 6.50	97.02 ± 7.70
8	100.00 ± 0.00	99.82 ± 0.26	99.14 ± 2.33	99.30 ± 3.57	99.92 ± 0.12
9	81.70 ± 10.54	83.00 ± 12.20	84.24 ± 16.59	92.18 ± 1.53	96.02 ± 5.45
10	97.90 ± 1.24	97.4 ± 1.21	97.60 ± 1.55	97.88 ± 7.20	97.51 ± 1.09
11	99.16 ± 0.42	99.17 ± 0.39	99.16 ± 0.35	98.93 ± 6.57	99.18 ± 0.50
12	98.33 ± 1.25	97.86 ± 1.29	98.11 ± 1.16	98.02 ± 8.25	98.59 ± 1.41
13	99.17 ± 1.15	99.34 ± 1.06	98.52 ± 1.36	99.16 ± 1.52	99.46 ± 0.86
14	99.25 ± 0.51	99.36 ± 0.73	99.00 ± 0.67	98.99 ± 4.02	99.15 ± 0.60
15	98.58 ± 1.06	98.23 ± 2.01	98.12 ± 0.82	98.54 ± 3.92	98.65 ± 1.13
16	94.46 ± 4.95	95.5 ± 4.17	96.16 ± 3.64	95.34 ± 4.88	94.33 ± 3.67
OA (%)	98.51 ± 0.26	98.61 ± 0.40	98.63 ± 0.33	98.75 ± 0.33	98.81 ± 0.32
AA (%)	97.05 ± 1.35	96.62 ± 1.37	97.39 ± 1.37	97.70 ± 1.03	98.83 ± 0.63
K × 100	98.58 ± 0.30	98.30 ± 0.46	98.42 ± 0.37	98.43 ± 0.38	98.65 ± 0.36

4. Discussions

This paper made some modifications and designed an HSI classification method. This study proposes an improved ViT model that introduces a re-attention mechanism and a local mechanism. Then, the improved ViT model is combined with the attention-based adaptive spectral–spatial kernel, which systematically combines bands from shallow to deep, enables neurons to adaptively adjust the receptive field size, and successfully handles the long-range dependence of the spectrum, making full use of the spectral–spatial information and local global information in HSI to improve the classification performance. The Focal Loss function is used to increase the loss weight of small-class samples and hard-to-classify samples in HSI data samples and Apollo. Furthermore, a quasi-Newton method for nonconvex stochastic optimization is introduced to dynamically incorporate the curvature of the loss function by approximating the Hessian via a diagonal matrix.

It can be seen from Tables 5–8 that the classical RBF-SVM method and several deep-learning-based methods including CNN, HybirdSN, PyResNet, SSRN, SSFTT, and A2S2KResNet are considered for comparison. All experimental results show that the proposed method achieves the best performance on all datasets. The suggested technique obtained superior performance in terms of classification accuracy on the four popular HSI datasets, according to all experimental results. Taking the Indian Pines dataset as an example, the OA, AA and K of the proposed method are improved by 18.8%, 19.48% and 21.56%, respectively, compared with RBF-SVM. Furthermore, compared with CNN, the OA of the proposed method is improved by 20.5%, 15.5%, 9.24% and 8.47% on the Indian, Pavia, Xuzhou and WHU-Hi-Longkou datasets, respectively. For the Pavia University dataset, the proposed method improves the OA by 8.05%, 5.96%, 0.06%, 0.14% and 0.22% compared with HybirdSN, PyresNet, SSRN, SSFTT and A2S2KResNet, respectively.

Furthermore, to verify the effectiveness of the proposed method for different HSI dataset, Figures 8–11 show the classification results of different methods for each class.

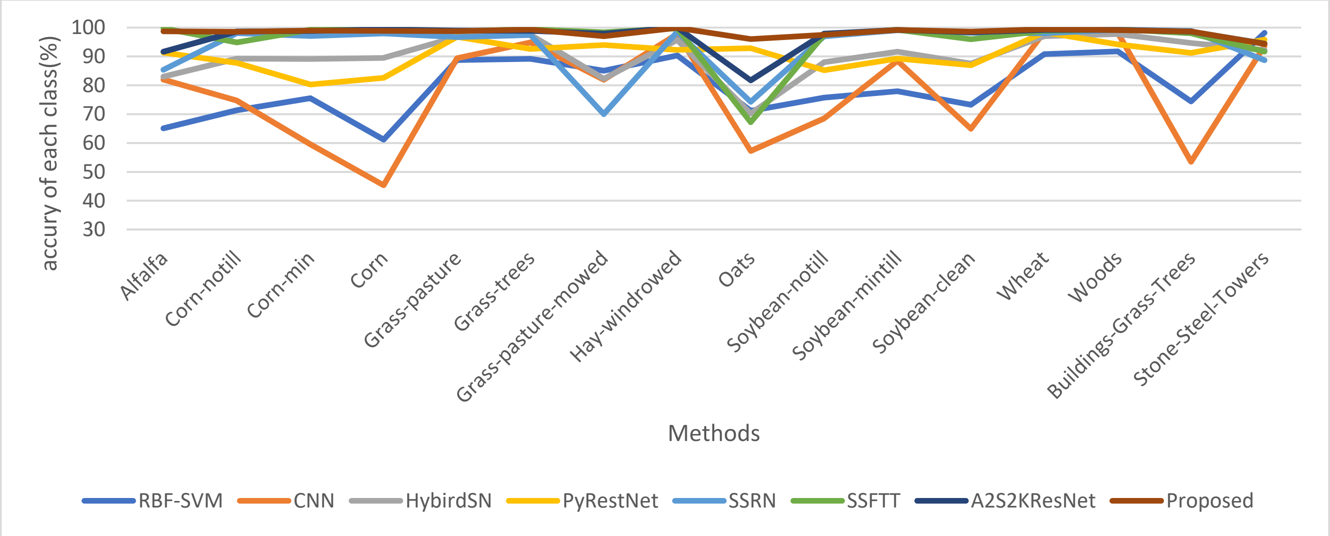


Figure 8. Classification results comparison for each class on the Indian Pines dataset.

We can see that our method achieves the highest classification accuracy for almost every class for four different datasets. For example, for the Oats category in the Indian Pines dataset, our method improves OA by 3.17% over the state-of-the-art among other methods. The effectiveness of the method is demonstrated in different datasets in different application domains. The urban land feature classification is realized on the Pavia University dataset, the mineral classification is realized on the Xuzhou dataset and the Indian Pines dataset and WHU-Hi-LongKou datasets are implemented for fine crop classification.

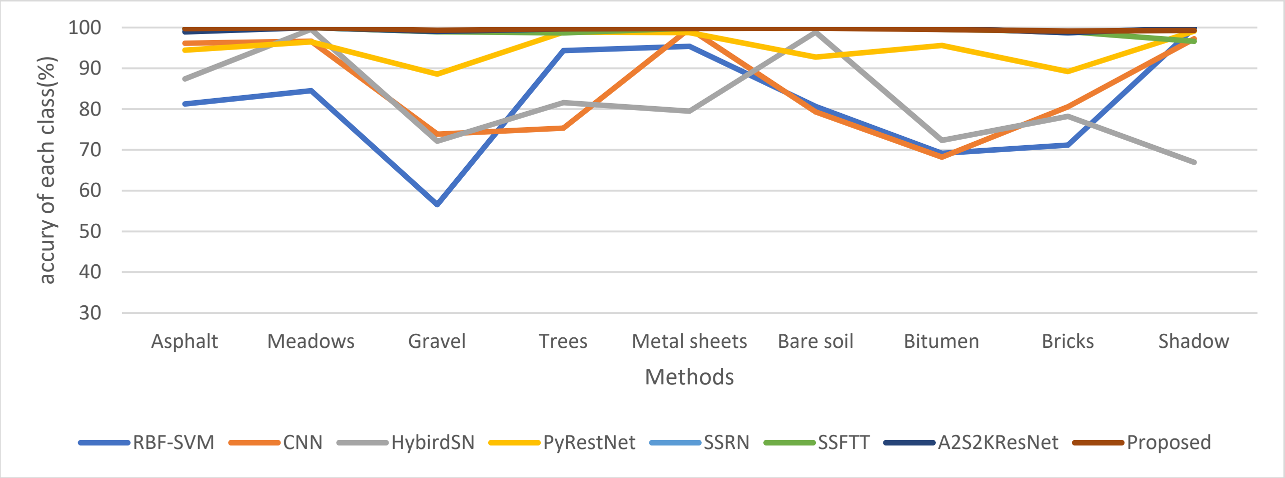


Figure 9. Classification results comparison for each class on the Pavia University dataset.

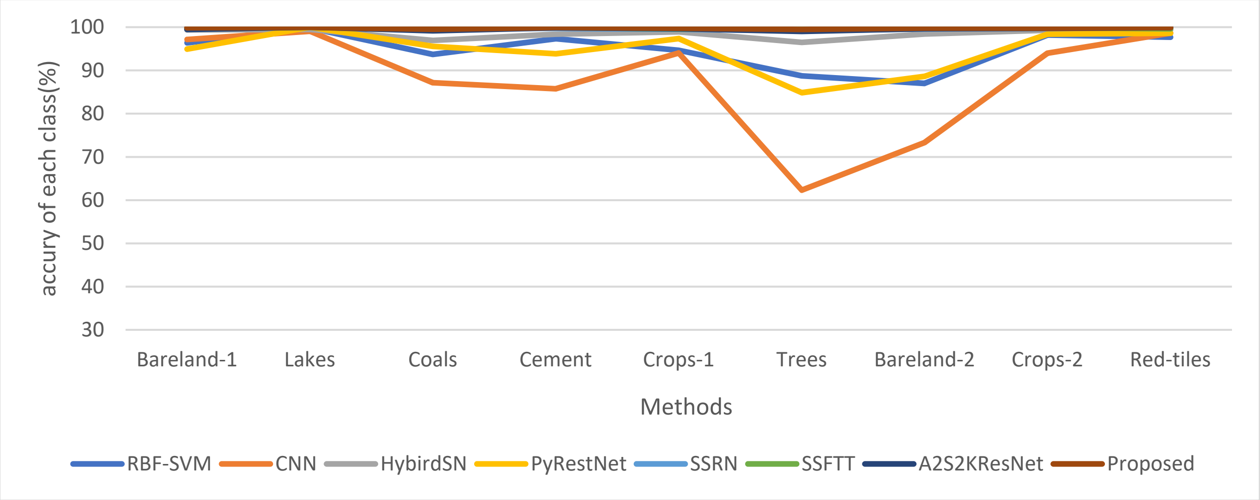


Figure 10. Classification results comparison for each class on the Xuzhou dataset.

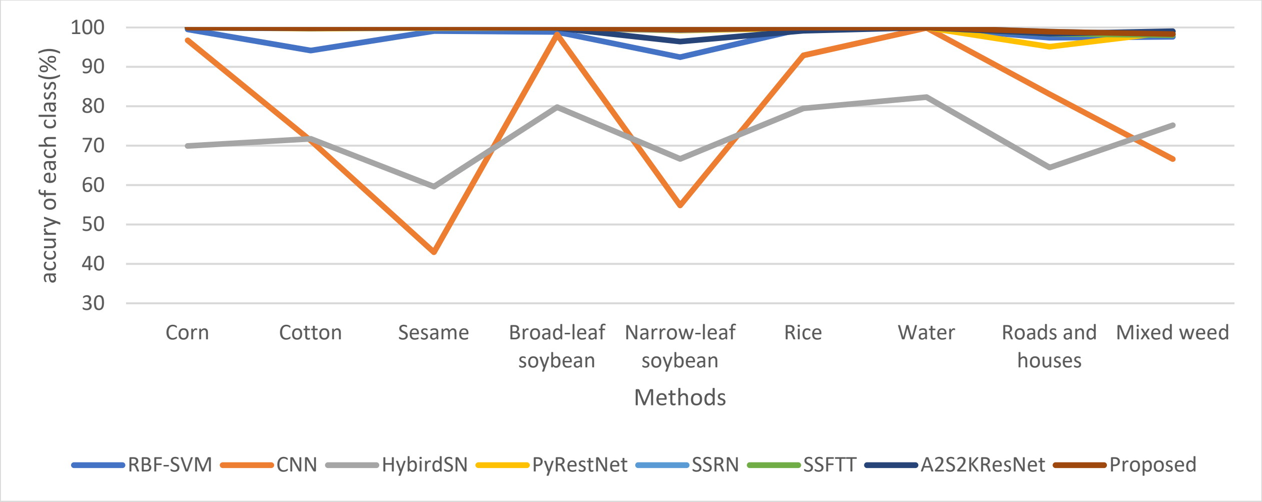


Figure 11. Classification results comparison for each class on the WHU-Hi-LongKou dataset.

5. Conclusions

In this study, an attention-based adaptive spectral–spatial kernel combined with an improved ViT network architecture is proposed to classify HSI images. For the spectra of HSI images that are approximately continuous, the proposed method fully utilized the local and global information of the data. Compared with classical methods and some deep-learning-based methods, the proposed method achieves excellent performance on four different datasets in urban land classification, crop classification and mineral classification. In the future research process, we will study strategies to improve the transformer's architecture to make it more suitable for HSI classification, build lightweight networks and reduce network complexity while ensuring the network's working performance.

Author Contributions: Conceptualization, A.W., S.X. and H.W.; methodology, S.X., H.W., A.W. and Y.I.; software, validation, S.X. and Y.Z.; writing—review and editing, H.W., A.W. and Y.I.; supervision, Y.I. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the National Natural Science Foundation of China under Grant NSFC-61671190 and Reserved Leaders of Heilongjiang Provincial Leading Talent Echelon 2021.

Data Availability Statement: The data are available at https://www.ehu.eus/ccwintco/index.php?%20title=Hyperspectral-Remote-Sensing-Scenes; http://rsidea.whu.edu.cn/resource_WHUHi_sharing.htm.

Acknowledgments: We thank for Kaiyuan Jiang for his valuable comments and discussion. Iwahori's research is supported by JSPS Grant-in-Aid for Scientific Research (C) (20K11873) and Chubu University Grant.

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

References

1. Ibrahim, A.; Franz, B.; Ahmad, Z.; Healy, R.; Knobelspiesse, K. Atmospheric correction for hyperspectral ocean color retrieval with application to the Hyperspectral Imager for the Coastal Ocean (HICO). *Remote Sens. Environ. Interdiscip. J.* **2018**, *204*, 60–75. [CrossRef]
2. Bhosle, K.; Musande, V. Evaluation of deep learning CNN model for land use land cover classification and crop identification using hyperspectral remote sensing images. *J. Indian Soc. Remote Sens.* **2019**, *47*, 1949–1958. [CrossRef]
3. Wang, B.; Shao, Q.; Song, D. A Spectral-Spatial Features Integrated Network for Hyperspectral Detection of Marine Oil Spill. *Remote Sens.* **2021**, *13*, 1568. [CrossRef]
4. Gao, A.F.; Rasmussen, B.; Kulits, P.; Scheller, E.L.; Greenberger, R.; Ehlmann, B.L. Generalized Unsupervised Clustering of Hyperspectral Images of Geological Targets in the Near Infrared. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Nashville, TN, USA, 19–25 June 2021; pp. 4289–4298.
5. Bo, C.J.; Lu, H.C.; Wang, D. Spectral-spatial K-Nearest Neighbor approach for hyperspectral image classification. *Multimed. Tools Appl.* **2018**, *77*, 10419–10436. [CrossRef]
6. Samadzadegan, F.; Hasani, H.; Schenk, T. Simultaneous feature selection and SVM parameter determination in classification of hyperspectral imagery using Ant Colony Optimization. *Remote Sens.* **2012**, *38*, 139–156. [CrossRef]
7. Li, M.; Zhang, N.; Pan, B.; Xie, S.; Wu, X.; Shi, Z. Hyperspectral Image Classification Based on Deep Forest and Spectral-Spatial Cooperative Feature and Deep Forest. In Proceedings of the International Conference on Image and Graphics, Shanghai, China, 13–15 September 2017; Volume 10668, pp. 325–336.
8. Kayabol, K. Bayesian Gaussian mixture model for spatial-spectral classification of hyperspectral images. In Proceedings of the 2015 23rd European Signal Processing Conference (EUSIPCO), Nice, France, 31 August–4 September 2015; pp. 1805–1809.
9. Wang, M.; Gao, K.; Wang, L.; Miu, X. A Novel Hyperspectral Classification Method Based on C5.0 Decision Tree of Multiple Combined Classifiers. In Proceedings of the 2012 Fourth International Conference on Computational and Information Sciences, Chongqing, China, 17–19 August 2012; pp. 373–376.
10. Lu, Y.; Wang, L.; Shi, Y. Classification of hyperspectral image with small-sized samples based on spatial-spectral feature enhancement. *J. Harbin Eng. Univ.* **2022**, *43*, 436–443.
11. 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]
12. Yu, S.; Jia, S.; Xu, C. Convolutional neural networks for hyperspectral image classification. *Neurocomputing* **2017**, *219*, 88–98. [CrossRef]
13. 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]

14. 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]
15. 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]
16. Roy, S.K.; Manna, S.; Song, T.; Bruzzone, L. Attention-Based Adaptive Spectral–Spatial Kernel ResNet for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 7831–7843. [CrossRef]
17. Alipour-Fard, T.; Paoletti, M.E.; Haut, J.M.; Arefi, H.; Plaza, J.; Plaza, A. Multibranch Selective Kernel Networks for Hyperspectral Image Classification. *IEEE Geosci. Remote Sens. Lett.* **2021**, *18*, 1089–1093. [CrossRef]
18. He, X.; Chen, Y.; Lin, Z. Spatial-Spectral Transformer for Hyperspectral Image Classification. *Remote Sens.* **2021**, *13*, 498. [CrossRef]
19. Hong, D. SpectralFormer: Rethinking Hyperspectral Image Classification with Transformers. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–15. [CrossRef]
20. He, J.; Zhao, L.; Yang, H.; Zhang, M.; Li, W. HSI-BERT: Hyperspectral Image Classification Using the Bidirectional Encoder Representation from Transformers. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 165–178. [CrossRef]
21. Han, K.; Xiao, A.; Wu, E.; Guo, J.; Xu, C.; Wang, Y. Transformer in Transformer. *arXiv* **2021**, arXiv:2103.00112.
22. Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; Jegou, H. Training data-efficient image transformers & distillation through attention. *arXiv* **2020**, arXiv:2012.12877.
23. Li, Y.; Zhang, K.; Cao, J.; Radu, T.; Luc, V.G. LocalViT: Bringing Locality to Vision Transformers. *arXiv* **2021**, arXiv:2104.05707.
24. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929.
25. 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 (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
26. Jimmy, L.B.; Jamie, R.K.; Geoffrey, E.H. Layer normalization. *arXiv* **2016**, arXiv:1607.06450.
27. Zhou, D.; Kang, B.; Jin, X.; Yang, L.; Lian, X.; Jiang, Z.; Hou, Q.; Feng, J. DeepViT: Towards Deeper Vision Transformer. *arXiv* **2021**, arXiv:2103.11886.
28. Ma, X. Apollo: An Adaptive Parameter-wise Diagonal Quasi-Newton Method for Nonconvex Stochastic Optimization. *arXiv* **2020**, arXiv:2009.13586.
29. Cui, Y.; Xia, J.; Wang, Z.; Gao, S.; Wang, L. Lightweight Spectral–Spatial Attention Network for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–14. [CrossRef]
30. Liang, Y.; Zhao, Z.; Wang, H. Unbalanced Geologic Body Classification of Hyperspectral Data Based on Squeeze and Excitation Networks at Tianshan Area. In Proceedings of the IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium, Waikoloa, HI, USA, 26 September–2 October 2020; pp. 6981–6984.
31. Tan, K.; Wu, F.; Du, Q.; Du, P.; Chen, Y. A Parallel Gaussian-Bernoulli Restricted Boltzmann Machine for Mining Area Classification with Hyperspectral Imagery. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2019**, *12*, 627–636. [CrossRef]
32. Wang, X.; Tan, K.; Du, Q.; Chen, Y.; Du, P. Caps-TripleGAN: GAN-Assisted CapsNet for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 7232–7245. [CrossRef]
33. Zhong, Y.; Hu, X.; Luo, C.; Wang, X.; Zhao, J.; Zhang, L. WHU-Hi: UAV-borne hyperspectral with high spatial resolution (H2) benchmark datasets and classifier for precise crop identification based on deep convolutional neural network with CRF. *Remote Sens. Environ.* **2020**, *250*, 112012. [CrossRef]
34. Zhong, Y.; Wang, X.; Xu, Y.; Wang, S.; Jia, T.; Hu, X.; Zhao, J.; Wei, L.; Zhang, L. Mini-UAV-borne hyperspectral remote sensing: From observation and processing to applications. *IEEE Geosci. Remote Sens. Mag.* **2018**, *6*, 46–62. [CrossRef]
35. Benediktsson, J.A.; Palmason, J.; Sveinsson, J.R. Classification of hyperspectral data from urban areas based on extended morphological profiles. *IEEE Trans. Geosci. Remote Sens.* **2005**, *43*, 480–491. [CrossRef]
36. Chen, Y.; Zhu, L.; Ghamisi, P.; Jia, X.; Li, G.; Tang, L. Hyperspectral Images Classification with Gabor Filtering and Convolutional Neural Network. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 2355–2359. [CrossRef]
37. Paoletti, M.E.; Haut, J.M.; Fernandez-Beltran, R. Deep pyramidal residual networks for spectral-spatial hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 740–754. [CrossRef]
38. Sun, L.; Zhao, G.; Zheng, Y.; Wu, Z. Spectral-Spatial Feature Tokenization Transformer for Hyperspectral Image Classifiction. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5522214. [CrossRef]