International Journal of Computer Science and Information Technology
ISSN: 3005-9682 (Print), ISSN: 3005-7140 (Online) | Volume 5, Number 2, Year 2025
DOI: https://doi.org/10.62051/ijcsit.v5n2.04
Journal homepage: https://wepub.org/index.php/IJCSIT/index

Hyperspectral Image Classification Based on the Improved Spectral Former

Tianyu Geng *, Hanwen Fan

College of Computer Science, Nanjing Audit University, Nanjing, China
* Corresponding Author: Tianyu Geng

ABSTRACT

Hyperspectral imaging has become a powerful tool in remote sensing, enabling fine-grained material identification and revealing the chemical and physical properties of materials. Its applications span urban land-use mapping, object recognition, crop classification, and agricultural yield prediction. The typical hyperspectral image classification workflow includes image loading, correction, noise reduction, feature extraction, classifier selection, training, classification, and result output. Feature extraction plays a critical role, but traditional methods such as SIFT, PCA, and LDA are limited in efficiency and accuracy, especially with large-scale datasets. Deep learning, particularly Convolutional Neural Networks (CNNs), has significantly improved classification performance by extracting hierarchical features from raw data. However, challenges remain in capturing both spectral and spatial information effectively. Transformer models, such as SpectralFormer, have been proposed to address these issues by leveraging attention mechanisms to capture long-range dependencies. Yet, they struggle with preserving spatial structures in hyperspectral images. The integration of Atrous Spatial Pyramid Pooling (ASPP) into SpectralFormer offers a promising solution to this problem, enhancing spatial feature extraction and improving overall classification performance. This paper discusses these advancements and highlights the potential of combining deep learning and spatial feature extraction techniques to address the unique challenges of hyperspectral image classification.

KEYWORDS

Hyperspectral Image; Transformer; Atrous Spatial Pyramid Pooling; Deep Learning.

1. INTRODUCTION

Hyperspectral imaging technology has become a powerful tool in modern remote sensing. By capturing hundreds of narrow wavelength bands at each pixel, hyperspectral imaging generates rich spectral data. These data not only enable fine-grained material identification but also reveal the chemical and physical properties of materials, providing robust support for diverse Earth observation and identification tasks such as urban land-use mapping, object recognition, crop classification, and agricultural yield prediction. This capability allows precise classification and detection of surface materials.

The general workflow for hyperspectral image classification typically includes steps such as image loading, image correction [1],[2], noise reduction, feature extraction, classifier selection, classifier training, classification, and result output. Among these steps, feature extraction [3][4] plays a pivotal role and has garnered significant research attention. Traditional hyperspectral classification methods mainly rely on manual feature extraction combined with classifiers. Common feature extraction techniques include SIFT, SURF, Principal Component Analysis (PCA), which projects hyperspectral

data from a high-dimensional space to a low-dimensional space while preserving maximum variance, and Linear Discriminant Analysis (LDA), which leverages probabilistic and statistical principles. While effective for small-scale classification problems, these methods exhibit significant limitations in computational efficiency and accuracy when handling large-scale training samples and highly complex datasets, ultimately reducing classification performance. These challenges stem primarily from the limited data fitting and representation capabilities of traditional methods.

The advent of artificial intelligence, especially the rise of deep learning, has profoundly influenced hyperspectral image classification. With its powerful nonlinear learning capabilities, deep learning autonomously extracts hierarchical and discriminative features from raw data, significantly enhancing classification accuracy [5]. Researchers have since focused on designing advanced network modules and integrating effective mechanisms to derive more valuable features from hyperspectral data, thereby improving the data representation and fitting capabilities of traditional methods. For instance, Zhao et al. [6] proposed a novel joint classification approach combining hierarchical random walks with a deep CNN architecture to extract features by fusing hyperspectral and LiDAR data. Experimental results demonstrated the superior classification performance of this method. In recent years, widely recognized backbone networks have been increasingly applied to hyperspectral image classification tasks. These include CNN-based backbone networks, such as 3D-CNN, 2D-CNN, and HybridSN, which integrates the advantages of three-dimensional (3D) and two-dimensional (2D) CNNs, as well as other deep learning-based networks like Recurrent Neural Networks (RNN), Generative Adversarial Networks (GAN), and attention mechanism-based architectures. Chen et al. [7] utilized autoencoder networks for deep feature extraction after PCA-based dimensionality reduction [8]. Ahmad integrated active learning with 3D-CNNs to mitigate performance drops due to limited sample sizes [9], while Yang [10] employed convolutional operations and self-attention mechanisms to separately extract spatial and spectral features.

Despite the significant advances brought by these foundational network architectures and their derivatives, certain limitations persist, particularly in representing spectral sequence information and capturing subtle changes along the spectral dimension. For example, Convolutional Neural Networks (CNNs) can effectively capture local features of images, reduce computational complexity, and alleviate overfitting risks; however, they struggle to capture global contextual information, which is crucial for tasks requiring a broader perspective. Similarly, 2D-CNNs primarily extract two-dimensional spatial information while underutilizing spectral dimensions, and 3D-CNNs, though capable of leveraging contextual information, may suffer interference from mixed-category pixels during feature extraction. Moreover, hyperspectral images often exhibit "same spectrum, different objects" and "different spectra, same object" phenomena. Classification approaches relying solely on spectral information, such as 1D-CNN or 2D-CNN, struggle to achieve high accuracy due to their neglect of spatial information. Meanwhile, RNNs, which process sequential data, face challenges with long-term dependencies, potentially overlooking subtle spectral variations when processing long spectral sequences.

To address the limitations of traditional sequence models, such as RNN and LSTM, Vaswani et al. proposed the Transformer architecture [11]. This non-recursive model relies entirely on an attention mechanism to capture global dependencies in sequence inputs and outputs, enabling parallelized computation. Transformers can attend to all positions in a sequence, effectively capturing contextual information [12], and have achieved remarkable breakthroughs in natural language processing. In hyperspectral data processing, Transformers have also demonstrated superior performance in modeling long-term dependencies within spectral features. However, they exhibit deficiencies in capturing local contextual features, where CNNs and RNNs excel. Additionally, the typical skip connections in Transformer architectures primarily operate within individual blocks, which limits information flow and connectivity across layers. To overcome these shortcomings, Hong et al. [13] introduced **SpectralFormer**, a Transformer-based architecture designed specifically for hyperspectral image classification. SpectralFormer incorporates two carefully designed modules:

Global Spectral Embedding (GSE) and Contextual Attention Fusion (CAF), which enhance its ability to extract spectral features and improve classification performance. Quantitative evaluations and ablation studies have demonstrated the superiority of SpectralFormer in hyperspectral classification tasks. However, due to its reliance on slicing hyperspectral images into small patches for sequential input, the original spatial structure is disrupted, leading to information loss.

The **Atrous Spatial Pyramid Pooling (ASPP)**[14] module, widely used in tasks like object detection and semantic segmentation, offers a promising solution to this issue. By employing atrous/dilated convolutions with varying dilation rates, the ASPP module expands the network's receptive field while maintaining image resolution. This allows it to capture features at multiple scales, improving the model's ability to handle objects of different sizes and shapes. Integrating the ASPP module into the SpectralFormer architecture can address the disruption of spatial features caused by patch-based input. By leveraging the ASPP module's ability to extract multi-scale spatial features, the enhanced architecture can better preserve the original spatial structure and achieve richer feature representation, ultimately improving overall classification performance.

2. METHODS

2.1. SPECTRALFORMER+

2.1.1. Overview

To address the challenges faced by deep learning models in hyperspectral image classification, we propose SpectralFormer+, an enhanced model that integrates the Atrous Spatial Pyramid Pooling (ASPP) module with the SpectralFormer architecture.

SpectralFormer+ consists of two primary components: SpectralFormer and ASPP. The ASPP module is designed to extract spatial features from hyperspectral image patches. These spatial features are then fed into SpectralFormer to extract spectral features. Finally, the extracted spectral features are processed by the classification system to generate the final hyperspectral classification map. The following sections provide detailed introductions to the SpectralFormer and ASPP modules.

2.1.2. SpectralFormer

Existing classification models are predominantly based on Convolutional Neural Networks (CNNs). Although these backbone networks and their variants have achieved remarkable classification results, they still fall short in representing spectral sequence information, particularly in capturing subtle spectral differences along the spectral dimension. To address these limitations, we build on the SpectralFormer model proposed by Hong et al., which effectively compensates for the deficiencies of CNN-based classification models. We further enhance this approach by integrating the Atrous Spatial Pyramid Pooling (ASPP) module with SpectralFormer, enabling the simultaneous learning of spatial and spectral features from hyperspectral images.

The SpectralFormer model is a novel Transformer-based architecture, primarily composed of two key modules: the Global Spectral Embedding (GSE) module and the Cross-Attention Filter (CAF) module. This architecture is designed to handle both pixel-level and patch-level inputs.

The Transformer architecture consists of two main components: the encoder and the decoder. These components are interconnected through self-attention mechanisms and feed-forward neural networks. The encoder processes the input sequence by stacking multiple identical layers. Each token in the input sequence undergoes self-attention calculations in each layer to progressively generate richer contextual representations. The decoder, tasked with generating the target sequence, is also composed of multiple layers. The output vectors from the decoder are transformed into specific classes through a softmax layer, producing the final output sequence. The self-attention mechanism is primarily used to capture the internal correlations within the data features. The self-attention mechanism can be

implemented through the following steps: Input Sequence: First, an input sequence of length n is provided, denoted as x_i (where $i = 1, \dots, n$), representing either scalars or vectors. Feature Embedding: Each element mi is embedded into a higher-dimensional space to produce the embedding a_i computed using a shared weight matrix Query, Key, and Value Vectors: Each embedding a_i is then multiplied by three distinct transformation matrices W_q, W_k, and W_v to generate three vectors, $i.e.,$: the query vector $\mathrm{Q} = [q_1, \dots, q_m]$, the key vector $\mathrm{K} = [k_1, \dots, k_m]$, and the value vector$V = [v_1, \dots, v_m]$. Attention Scores: The attention scores s are computed as the dot product between each query vector and each key vector. To stabilize the gradients, the scores are scaled by the square root of the dimension d of q_i or k_j, resulting in the normalized score $i.e. S_{i,j} = q_i \cdot \frac{k_j}{\sqrt{d}}$. SoftMax Activation: The softmax function is applied to the scaled scores s to produce the normalized attention weights. For example, at position 1, the normalized attention weight is given by $\hat{s}_{1,i} = \frac{e^{s_{1,i}}}{\sum_j e^{s_{1,i}}}$. Attention Representation: The final attention representation $\mathrm{Z} = [z_1, \dots, z_m]$ is computed by taking a weighted sum of the value vectors v_i using the normalized attention weights. For instance,$z_1 = \sum_i \hat{s}_{1,i} v_i$. In summary, the self-attention (SA) layer can be formulated as follows:

$$\mathrm{z} = \mathrm{Attention}(\mathrm{Q}, \mathrm{K}, \mathrm{V}) = \mathrm{softmax}(\frac{\mathrm{QKT}}{\sqrt{\mathrm{d}}})\mathrm{V}. \quad (1)$$

Utilizing Equation (1), multiple distinct self-attention (SA) layers can be assembled into a multi-head attention mechanism. Specifically, we first obtain multiple attention representations ($e.g., h = 8$), denoted as z^h where $h = 1, \dots, 8$. These representations are concatenated to form a larger feature matrix. Finally, a linear transformation matrix ($e.g., W_0$) is applied to match the feature dimensionality with that of the input data.

Assuming the input to SpectralFormer is y (where y is the output from the ASPP module), y is first divided into image patches. Subsequently, the grouped spectral embedding (GSE) is learned and fed into the Transformer model. To fully leverage features from different layers of the Transformer, the Cross-Attention Filter (CAF) is employed to adaptively fuse these features. The output from the Transformer is then passed through a Multi-Layer Perceptron (MLP) to generate the probability vector P for the input hyperspectral image patch, where the dimension of P corresponds to the number of classes in the hyperspectral image classification task.

2.1.3. Atrous Spatial Pyramid Pooling

In remote sensing scenes, objects vary significantly in scale, such as cars and buildings. The spatial features of these objects are crucial for accurate identification, necessitating a model capable of extracting features across different scales. However, the SpectralFormer model lacks a dedicated learning process for spatial features. It divides the original image into small patches and inputs them into the model in sequence form. This approach disrupts the spatial structure of the image, leading to a loss of spatial information. As a result, the classification model struggles to learn effective spatial features and contextual information, further complicating the identification of objects in the scene.

To address these challenges, we propose integrating the Atrous Spatial Pyramid Pooling (ASPP) module into the SpectralFormer architecture. The ASPP module, which employs dilated convolutions with different dilation rates, can capture contextual information at multiple scales. By stacking contextual information from different scales, the ASPP module effectively addresses the limitations of the original SpectralFormer model. This integration enhances the model's ability to extract spatial features and contextual information, thereby improving the overall classification performance.

The Atrous Spatial Pyramid Pooling (ASPP) module consists of five branches: a 1×1 convolution layer, three dilated convolution layers with dilation rates of 6, 12, and 18 (each using a 3×3 convolution kernel), and an adaptive average pooling branch followed by a 1×1 convolution and upsampling. For change detection, the input image patches are denoted as $X_1 \in R^{H \times W \times B}$ and $X_2 \in R^{H \times W \times B}$, where $H \times W$ represents the spatial dimensions of the patches, and B denotes the number

of spectral bands in the hyperspectral data. When these image patches are fed into the ASPP module, each branch generates a feature map. Assuming each branch produces 200 feature channels, the resulting feature maps from the five branches are $F1, F2, F3, F4, F5 \in R^{H\times W\times 200}$.To fully leverage the features captured at different spatial scales, these feature maps are concatenated along the channel dimension, resulting in a combined feature $F \in R^{H\times W\times 1000}$. This concatenated feature is then passed through a 1×1 convolution layer to compress the channel dimension, yielding the final feature $F' \in R^{H\times W\times 200}$. Figure 1 illustrates the SpectralFormer+ proposed for the HS image classification task.

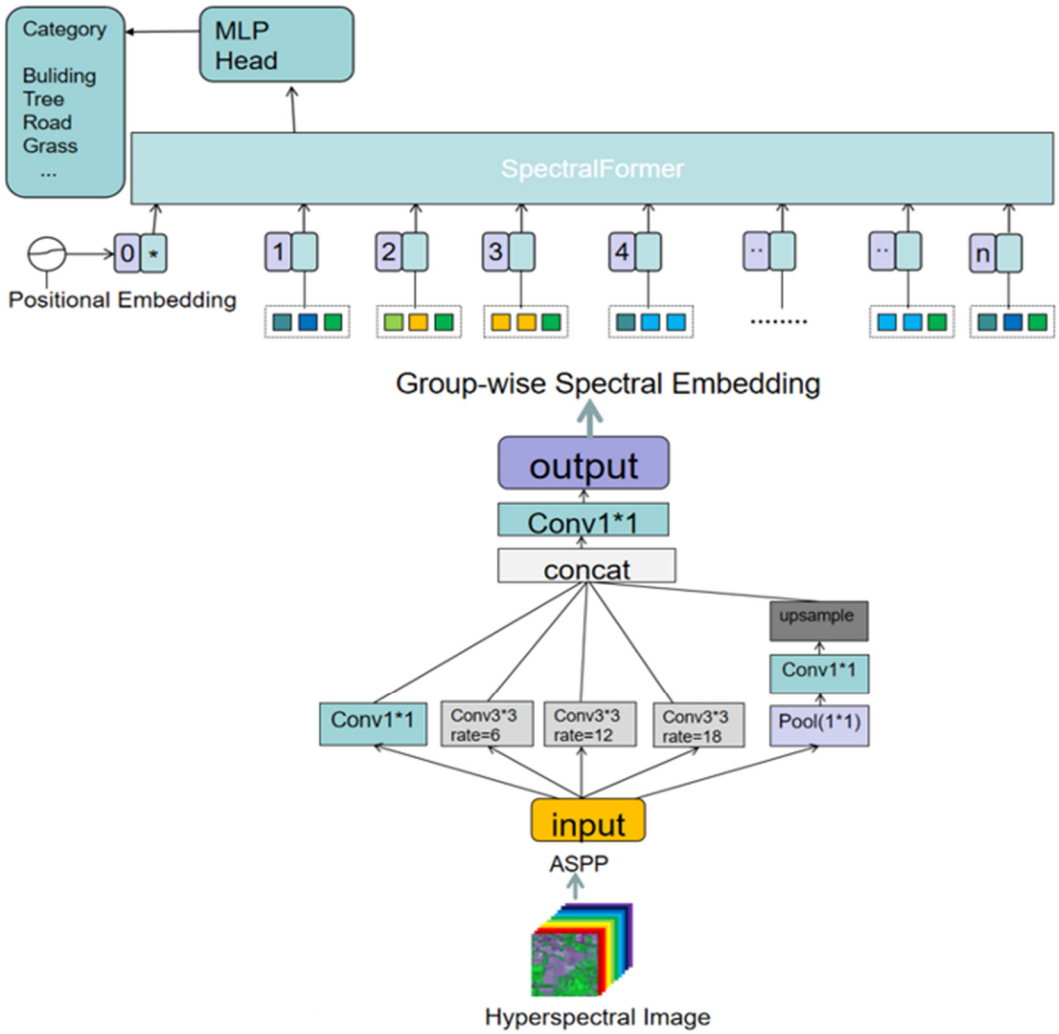

Figure 1. An overview illustration of the SpectralFormer+ network proposed for the hyperspectral image classification task

3. EXPERIMENTAL SECTION

3.1. Datasets Utilized

This section introduces three renowned hyperspectral (HS) datasets used in our experiments.

3.1.1. Indian Pines Dataset

The Indian Pines dataset was collected in 1992 by a collaboration between NASA's Jet Propulsion Laboratory (JPL) and Purdue University, using the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor over Indiana, USA. The dataset features a spatial resolution of 145×145 pixels, with a ground sampling distance (GSD) of 20 meters. The spectral range spans from 400 nm to 2500 nm, covering 220 spectral bands with a resolution of 10 nm. However, due to noise and water absorption,

20 bands are typically removed, leaving 200 valid spectral bands for analysis. The dataset includes 16 land cover categories, such as corn, soybeans, grassland, and trees. Table 1 lists the standard training and testing set sample sizes for each category.

Table 1. Training and testing sample sizes for the Indian Pines dataset.

Class No	Class Name	Training	Testing
1	Corn Notill	50	1348
2	Corn Mintill	50	784
3	Corn	50	184
4	Grass Pasture	50	447
5	Grass Trees	50	697
6	Hay Windrowed	50	439
7	Soybean Notill	50	918
8	Soybean Mintill	50	2418
9	Soybean Clean	50	564
10	Wheat	50	162
11	Woods	50	1244
12	Buildings Grass Trees Drives	50	330
13	Stone Steel Towers	50	45
14	Alfalfa	15	39
15	Grass Pasture Mowed	15	11
16	Oats	15	5
	Total	695	9671

3.1.2. Pavia University Dataset

The Pavia University dataset was captured by the ROSIS-3 sensor over Pavia, Italy. The dataset contains an image of 610×340 pixels, with 115 spectral bands. Due to noise, 12 bands are excluded, resulting in 103 usable spectral bands. The dataset is classified into nine land cover categories, including asphalt, grassland, gravel, trees, metal sheets, bare soil, bricks, and shadows. The dataset contains a total of 42,776 labeled samples. Detailed information on the training and testing sample sizes is provided in Table 2.

Table 2. Training and testing sample sizes for the Pavia University dataset.

Class No	Class Name	Training	Testing
1	Asphalt	548	6304
2	Meadows	540	18146
3	Gravel	392	1815
4	Trees	524	2912
5	Metal Sheets	265	1113
6	Bare Soil	532	4572
7	Bitumen	375	981
8	Bricks	514	3364
9	Shadows	231	795
	Total	3921	40002

3.1.3. Houston 2013 Dataset

The Houston 2013 dataset was captured by the Compact Airborne Spectrographic Imager (CASI) sensor over the University of Houston campus and its surrounding urban areas. The dataset features 144 spectral bands with a spectral range of 364-1046 nm and a spatial resolution of 2.5 meters. The dataset includes 15 challenging land cover and land use categories. Table 3 lists the training and testing sample sizes for each category.

Table 3. Training and testing sample sizes for the Houston 2013 dataset.

Class No	Class Name	Training	Testing
1	Healthy Grass	198	1053
2	Stressed Grass	190	1064
3	Synthetic Grass	192	505
4	Tree	188	1056
5	Soil	186	1056
6	Water	182	143
7	Residential	196	1072
8	Commercial	191	1053
9	Road	193	1059
10	Highway	191	1036
11	Railway	181	1054
12	Parking Lot1	192	1041
13	Parking Lot2	184	285
14	Tennis Court	181	247
15	Running Track	187	473
	Total	2832	12197

3.2. Experimental Setup

3.2.1. Evaluation Metrics

We adopted three commonly used metrics to quantitatively assess the classification performance of each model: Overall Accuracy (OA), Average Accuracy (AA), and Kappa Coefficient (K). Additionally, classification maps generated by different models were visualized for qualitative comparison.

3.2.2. Model Architecture

The proposed method integrates the Atrous Spatial Pyramid Pooling (ASPP) module into the SpectralFormer architecture. The SpectralFormer model comprises two key modules: Global Spectral Embedding (GSE) and Cross-Attention Filter (CAF). The ASPP module is incorporated to enhance the model's ability to capture multi-scale spatial features.

The hyperspectral image is divided into multiple patches, with each patch serving as an input sample. This patch-wise input method enables the model to capture spatial context information effectively. The experiments employed standard training and testing set partitions to ensure accurate evaluation of model performance. Ablation studies were conducted to assess the impact of the ASPP module on model performance. The proposed method was implemented on the PyTorch platform using a workstation equipped with an AMD Ryzen 7 6800H processor (3.20 GHz), 16.0 GB RAM, and an NVIDIA GeForce RTX 3050 Laptop GPU (4 GB GDDR6). The Adam optimizer was used with a batch size of 64. The initial learning rate was set to 5e-4 and decayed by a factor of 0.9 every tenth of the total training epochs. The number of training epochs was set to 300 for the Indian Pines dataset, 480 for the Pavia University dataset, and 600 for the Houston 2013 dataset.

3.3. Model Analysis

3.3.1. Experimental Process

Experiments were conducted on the three datasets to validate the improvement in classification performance of the SpectralFormer model with the ASPP module. The effectiveness of the ASPP module was evaluated using the patch-wise input method.

3.3.2. Ablation Study

Extensive ablation experiments were performed on the Indian Pines, Pavia University, and Houston 2013 datasets to verify the impact of the ASPP module on classification accuracy. The classification

performance of the SpectralFormer model with and without the ASPP module was compared in terms of OA, AA, and K.

3.4. Analysis of Experimental Results

For the three hyperspectral datasets, Tables 4-6 report the quantitative classification results, including OA, AA, and K, as well as the accuracy for each category. The results indicate that the ASPP module significantly enhances the model's performance across all metrics. Specifically, the Kappa Coefficient (K) reflects improved consistency in classification results. In summary, the SpectralFormer model with the ASPP module outperformed the baseline SpectralFormer model in terms of overall classification accuracy, average classification accuracy, and Kappa coefficient. The ASPP module effectively enhanced the model's ability to capture multi-scale features in hyperspectral images, thereby improving classification performance.

3.5. Performance Comparison

Table 4. The overall accuracy (OA), average accuracy (AA), Kappa coefficient (κ), and the accuracy for each class of different classification methods on the Indian Pines dataset will be presented in the following table.

Class No	Transformers (ViT)	SpectralFormer		SpectralFormer+
		pixel-wise	patch-wise	patch-wise
1	49.92	66.84	62.50	79.91
2	62.50	66.96	91.45	94.64
3	83.70	92.39	94.57	98.91
4	88.81	92.84	93.51	99.78
5	86.66	84.79	86.66	99.86
6	92.48	94.30	95.44	100.00
7	80.28	88.01	88.34	95.97
8	71.62	71.50	78.74	91.89
9	46.10	72.16	69.15	95.04
10	98.77	98.77	99.38	100.00
11	90.68	93.01	95.02	97.75
12	48.48	59.70	84.24	93.03
13	100.00	100.00	100.00	100.00
14	82.05	87.18	61.54	100.00
15	90.91	90.91	90.91	100.00
16	100.00	100.00	100.00	100.00
OA	72.50	78.50	82.79	93.76
AA	79.56	84.96	86.97	96.67
K	68.50	75.55	80.40	92.45

The experimental results on the three datasets are shown in Tables 4-6. In Table 4, ViT, the basic Transformer model, achieves the worst performance. In comparison, the pixel-wise Spectralformer significantly improves performance, primarily due to its effective extraction of local continuous spectral features using group-wise spectral embedding. To further utilize the neighborhood information of pixels, the patch-wise Spectralformer uses hyperspectral cubes as input, which further boosts performance. Most importantly, the proposed Spectralformer+ achieves the highest OA, AA, and K values, which are 93.76%, 96.67%, and 92.45%, respectively. This improvement is mainly

attributed to the ability of Spectralformer+ to effectively extract spatial features using the ASPP module.

Table 5. The overall accuracy (OA), average accuracy (AA), Kappa coefficient (κ), and the accuracy for each class of different classification methods on the Pavia University dataset will be presented in the following table

Class No	Transformers (ViT)	SpectralFormer		SpectralFormer+
		pixel-wise	patch-wise	patch-wise
1	74.30	85.37	78.24	94.15
2	68.75	89.66	91.97	89.20
3	72.07	81.10	71.90	75.15
4	93.89	94.23	96.39	96.22
5	99.37	99.37	98.83	96.68
6	88.47	76.86	92.74	90.38
7	87.26	87.05	85.12	95.72
8	85.02	90.55	95.39	98.75
9	99.25	99.87	91.45	96.60
OA	77.14	87.94	91.62	94.44
AA	85.37	89.34	89.11	92.54
K	71.03	83.89	86.20	88.46

Table 6. The overall accuracy (OA), average accuracy (AA), Kappa coefficient (κ), and the accuracy for each class of different classification methods on the Houston2013 dataset will be presented in the following table.

Class No	Transformers (ViT)	SpectralFormer		SpectralFormer+
		pixel-wise	patch-wise	patch-wise
1	82.91	83.67	83.38	79.68
2	96.15	98.59	99.72	100.00
3	100.00	99.80	93.86	99.80
4	99.34	96.88	96.78	90.71
5	97.35	98.58	99.81	100.00
6	95.10	95.10	93.71	98.60
7	79.76	86.57	86.66	85.26
8	59.45	70.85	79.68	80.91
9	64.59	72.33	80.64	75.26
10	86.10	77.51	63.71	93.92
11	75.14	91.18	85.20	90.98
12	49.76	75.12	80.98	66.67
13	57.89	73.33	71.93	83.50
14	99.60	99.19	98.38	100.00
15	98.31	98.52	99.58	100.00
OA	80.81	86.14	87.85	90.04
AA	82.76	87.81	87.60	89.69
K	79.20	85.27	85.47	86.76

On the Pavia University and Houston2013 datasets, the proposed Spectralformer+ also achieves the best performance, attaining the highest values in OA, AA, and K across all metrics.

4. CONCLUSION

In hyperspectral imaging, rich spectral data is formed by collecting hundreds of narrow wavelength bands of information at each pixel point, enabling detailed identification. SpectralFormer has demonstrated high performance in hyperspectral image classification. However, the SpectralFormer architecture slices the original image into small patches and inputs them into the model in sequence, disrupting the spatial features of the original image and resulting in the loss of original information. This makes it difficult for the classification model to learn the spatial features of objects and effective contextual information. To address this, we propose a new architecture based on SpectralFormer, called SpectralFormer+. It utilizes the atrous spatial pyramid pooling (ASPP) module, which employs dilated convolutional layers with different dilation rates to capture image features at different scales, to solve the problem of disrupting the original spatial feature structure faced by the previous architecture. This significantly enhances classification performance.

ACKNOWLEDGMENTS

This work was supported by the Jiangsu College Students' Innovation and Entrepreneurship Training Program under Grant 202411287088Y.

REFERENCES

[1] W. Cao, K. Wang, G. Han, J. Yao, and A. Cichocki, "A robust PCA approach with noise structure learning and spatial–spectral low-rank modeling for hyperspectral image restoration," IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 11, no. 10, pp. 3863–3879, Oct. 2018.

[2] J. Peng et al., "Low-rank and sparse representation for hyperspectral image processing: A review," IEEE Geosci. Remote Sens. Mag., early access, Jun. 10, 2021, doi: 10.1109/MGRS.2021.3075491.

[3] D. Hong, N. Yokoya, J. Chanussot, J. Xu, and X. X. Zhu, "Learning to propagate labels on graphs: An iterative multitask regression framework for semi-supervised hyperspectral dimensionality reduction," ISPRS J. Photogramm. Remote Sens., vol. 158, pp. 35–49, Dec. 2019.

[4] Q. Li, B. Zheng, B. Tu, J. Wang, and C. Zhou, "Ensemble EMD based spectral-spatial feature extraction for hyperspectral image classifi-cation," IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 13, pp. 5134–5148, 2020.

[5] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, pp. 436–444, May 2015.

[6] X. Zhao et al., "Joint classification of hyperspectral and LiDAR data using hierarchical random walk and deep CNN architecture," IEEE Trans. Geosci. Remote Sens., vol. 58, no. 10, pp. 7355–7370, Oct. 2020.

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

[8] H. Abdi and L. J. Williams, "Principal component analysis," Wiley Interdiscipl. Rev., Comput. Statist., vol. 2, no. 4, pp. 433-459, 2010.

[9] Ahmad M, Ghous U, Hong D, et al. A disjoint samples-based 3d-cnn with active transferlearning for hyperspectral image classification[J]. IEEE Transactions on Geoscience andRemote Sensing, 2022, 60: 1-16.

[10] Yang L, Yang Y, Yang J, et al. FusionNet: a convolution-transformer fusion network forhyperspectra1 image classification [J]. Remote Sensing, 2022, 14(16): 4066.

[11] A. Vaswani et al., "Attention is all you need," 2017, arXiv:1706.03762.

[12] G. Ke, D. He, and T.-Y. Liu, "Rethinking positional encoding in language pre-training," 2020, arXiv:2006.15595.

[13] D. Hong et al., "SpectralFormer: Rethinking Hyperspectral Image Classification With Transformers," in IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1-15, 2022, Art no. 5518615, doi: 10.1109/ TGRS. 2021. 3130716.

[14] L. -C. Chen, G. Papandreou, I. Kokkinos, K. Murphy and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 4, pp. 834-848, 1 April 2018, doi: 10.1109/TPAMI.2017.2699184.