remote sensing

MDPI

Article

An Adaptive Feature Enhanced Gaussian Weighted Network for Hyperspectral Image Classification

Fei Zhu [1], Cuiping Shi [2,*], Liguo Wang [3] and Haizhu Pan [4]

[1] College of Communication and Electronic Engineering, Qiqihar University, Qiqihar 161000, China; 2022935750@qqhru.edu.cn
[2] College of Information Engineering, Huzhou University, Huzhou 313000, China
[3] College of Information and Communication Engineering, Dalian Nationalities University, Dalian 116000, China; wangliguo@hrbeu.edu.cn
[4] College of Computer and Control Engineering, Qiqihar University, Qiqihar 161000, China; panhaizhu@qqhru.edu.cn
* Correspondence: shicuiping@zjhu.edu.cn

Abstract: Recently, research on hyperspectral image classification (HSIC) methods has made significant progress. However, current models commonly only focus on the primary features, overlooking the valuable information contained in secondary features that can enhance the model's learning capabilities. To address this issue, an adaptive feature enhanced gaussian weighted network (AFGNet) is proposed in this paper. Firstly, an adaptive feature enhancement module (AFEM) was designed to evaluate the effectiveness of different features and enhance those that are more conducive to model learning. Secondly, a gaussian weighted feature fusion module (GWF2) was constructed to integrate local and global feature information effectively. Finally, a multi-head collaborative attention (MHCA) mechanism was proposed. MHCA enhances the feature extraction capability of the model for sequence data through direct interaction and global modeling. Extensive experiments were conducted on five challenging datasets. The experimental results demonstrate that the proposed method outperforms several SOTA methods.

Keywords: hyperspectral image classification; convolutional neural networks; transformer; feature enhancement; gaussian weight; attention

Academic Editor: Salah Bourennane

Received: 25 December 2024
Revised: 15 February 2025
Accepted: 20 February 2025
Published: 22 February 2025

Citation: Zhu, F.; Shi, C.; Wang, L.; Pan, H. An Adaptive Feature Enhanced Gaussian Weighted Network for Hyperspectral Image Classification. *Remote Sens.* **2025**, *17*, 763. https://doi.org/10.3390/rs17050763

1. Introduction

Hyperspectral images (HSIs) are cubic data that simultaneously capture spatial and spectral information of target objects [1]. By combining imaging technology with spectroscopy, hyperspectral images generate continuous and narrow-band spectral image data, enabling a more comprehensive and detailed revelation of ground features [2]. Due to their unique spatial spectral integration characteristics, hyperspectral images are widely used in various fields such as medical imaging [3,4], agriculture [5,6], and mineral exploration [7].

HSIC is the process of analyzing the information within the image to assign category labels to each pixel in the image [8]. Early popular HSIC techniques primarily relied on the unique spectral information of different ground objects for classification. Some representative methods include K-nearest neighbor classifiers [9], support vector machines [10–12], and sparse representation classification [13]. Although these methods are simple and easy to implement, they only utilize spectral information and do not consider the continuity of spatial land distribution, resulting in incomplete feature extraction. In addition, traditional classification methods overly rely on expert experience, making it difficult to ensure the robustness and generalization of the methods.

In recent years, deep learning has shown strong application potential in the field of HSIC [14–18]. Convolutional neural network (CNN) [19,20], deep belief networks (DBNs) [21], and stacked autoencoders (SAEs) [22] have achieved good results in HSIC. However, both DBNs and SAEs require input data to be 1D vectors, which leads to significant limitations in the application of these models. CNN, with its characteristics of local perception and parameter sharing, can extract deeper data features while avoiding a sharp increase in model parameters, thus receiving widespread attention. Zhang et al. [23] proposed a spectral partitioning residual network (SPRN) based on 2D convolution kernels. This method divides the input spectrum into multiple non-overlapping continuous sub-bands by equivalently using grouped convolution, and employs cascaded parallel improved residual blocks to extract spectral–spatial features from these sub-bands separately [24]. Roy et al. [25] proposed a hybrid spectral CNN (HybridSN). Compared with the method that only uses 2D convolution kernels, this method combines more suitable 3D convolution kernels for hyperspectral data based on 2D convolution kernels, thus learning more abstract feature representations. Li et al. [26] proposed a central vector oriented self-similarity network (CVSSN), aiming to address the issue that existing CNN-based models ignore the potential relationship between the central pixel and its neighboring pixels when processing HSIs. Su et al. [27] proposed a method based on normalized spectral clustering with kernel-based learning (NSCKL). This method adopts a normalized spectral clustering algorithm, which can learn new features under the Manifold Hypothesis, and combines clustering features with extreme learning machines through kernel learning methods to achieve semi-supervised classification. Mei et al. [28] proposed a step activation network with binary weight (SAWB). This method replaces floating-point operations with integers, thereby accelerating the inference process of the network and reducing computational costs.

In traditional methods, treating each pixel equally makes it difficult to focus on valuable pixels [29–31]. To alleviate this problem, Zhang et al. [32] proposed a local–global cross fusion network with Gaussian initialization position prompts (LGGNet). Yang et al. [33] proposed a cross-attention spectral–spatial network (CASSN). This method utilizes cross-spectral and cross-spatial attention components to generate frequency band weights and spectral–spatial features, aiming to alleviate the problem of previous methods being less robust to HSI rotation. Wu et al. [34] proposed a network combining cross-channel dense connection and multi-scale dual aggregated attention (CDC_MDAA) to alleviate the difficulty of labeling data in HSIC. Zhang et al. [35] proposed a spectral–spatial self-attention network (SSSAN). This network can adaptively combine the local features of the pixels to be classified with long-distance dependencies.

Despite the numerous significant achievements of CNN in the field of HSIC, the limited receptive field has remained a persistent challenge. This problem makes it relatively difficult for networks to learn the dependency relationships of long-distance spatial distributions. Although this problem can be alleviated by using convolution kernels of different sizes, it also increases the complexity of the model and may lead to overfitting during network training.

Transformer, a deep learning model initially proposed by Vaswani et al. in 2017, was originally applied in the field of NLP and achieved remarkable results [36]. However, the early application of Transformer in the field of CV faced various challenges, including computational efficiency, model adaptability, and the complexity of training and inference. With the emergence of research achievements such as ViT, LeViT, DeepViT, and others, the application of Transformer in the CV field has been greatly expanded [37–41]. In view of this, some scholars have begun to apply Transformer in the field of HSIC. Hong et al. [42] proposed a backbone network specifically designed for HSIC, known as SpectralFormer (SF). This method can not only learn local spectral representations but also effectively

transfers similar components from shallow to deep layers. Mei et al. [43] proposed a group-aware hierarchical Transformer (GAHT), which extracts more detailed local spatial spectral relationships through a group embedding module. Zhong et al. [44] proposed a spectral spatial Transformer network (SSTN), which comprises spatial attention and spectral correlation modules to overcome the limitations of the limited receptive field of convolutional kernels.

The multi-head self-attention mechanism (MHSA) in Transformer can calculate the cross-correlation between all elements in the input sequence, but this also makes it difficult to fully focus on local features. To overcome this limitation, some researchers have attempted to combine Transformer with other models. Song et al. [45] proposed a bottleneck spatial spectral Transformer (BS2T), which combines CNN and Transformer to achieve feature extraction and capture long-distance dependencies. Specifically, CNN is utilized for feature compression and expansion, while Transformer enhances the feature representation by capturing long-distance dependencies. Sun et al. [46] proposed a spectral spatial feature tokenization transformer (SSFTT), which extracts spectral spatial features through CNN and employs a feature tokenizer to perform feature transformation to obtain advanced semantic features. Jiang et al. [47] proposed a graph generative structure aware transformer (GraphGST). This method proposes an absolute position encoding to obtain the absolute position sequence of pixels and integrate it into the Transformer architecture. Feng et al. [48] proposed a central attention transformer (CAT). This method improves classification performance by using superpixel sampling and multi-level random sampling mechanisms, combined with spatial spectral tokens and central attention structures.

In addition, Wang et al. [49] proposed a capsule attention network (CAN). CAN combines activity vectors with attention mechanisms to improve HSIC. Convolution and Transformer consume a lot of computing resources during training. Therefore, Jamali et al. [50] proposed a novel backbone network called HybridKAN. This method achieves faster convergence speed by fusing 1D, 2D, and 3D KAN modules.

Although the above methods have achieved good classification performance, there are still some challenges, including the following:

1. Existing models often only focus on the primary features, neglecting the potential value of secondary features in model learning.
2. The local receptive field of the convolution kernel makes it difficult to obtain global information.

To address these issues, an adaptive feature enhanced gaussian weighted network (AFGNet) is proposed. Firstly, an adaptive feature enhancement module (AFEM) was designed. AFEM can accurately evaluate the impact of different features on classification performance and dynamically enhance features that are more conducive to model learning. Secondly, a Gaussian weighted feature fusion module (GWF2) was proposed to effectively fuse local and global features by establishing mapping relationships between local features. Finally, a multi-head collaborative attention (MHCA) mechanism was developed. Through direct interaction and global modeling, it enhances the model's ability to extract key features from the sequence data. Extensive experiments are conducted on five challenging datasets. The experimental results demonstrate that the proposed method outperforms several SOTA methods.

The main contributions of this paper can be summarized as follows:

1. An adaptive feature enhancement module (AFEM) is proposed. AFEM can adaptively enhance features that are more conducive to model learning.
2. A Gaussian weighted feature fusion module (GWF2) is proposed. GWF2 effectively integrates local and global features by extracting and constructing mapping relationships between local features.

3. A multi-head collaborative attention (MHCA) mechanism is designed. By direct interaction and global modeling, MHCA can more adequately capture the key features in the input sequence.

The remaining parts of this paper are arranged as follows: Section 2 introduces the three modules of the proposed method in detail. Section 3 first introduces the datasets and experimental settings. Then, detailed experimental verification was conducted on the proposed method. Section 4 provides conclusions and prospects for future research directions.

2. Methodology

Original HSI data are denoted as $\boldsymbol{I} \in \mathbb{R}^{h \times w \times l}$, where $h \times w$ is the spatial size and l is the number of spectral bands.

2.1. Overall Structure

The overall structure of the proposed AFGNet is shown in Figure 1, which mainly consists of three modules: an AFEM module for adaptively enhancing features that are more beneficial for model learning, a GWF2 module for extracting and fusing "spatial-spectral" features, and multiple Transformer encoders (TE) incorporating MHCA to learn relationships among global features. Detailed introductions about all three modules are presented in Sections 2.2–2.4.

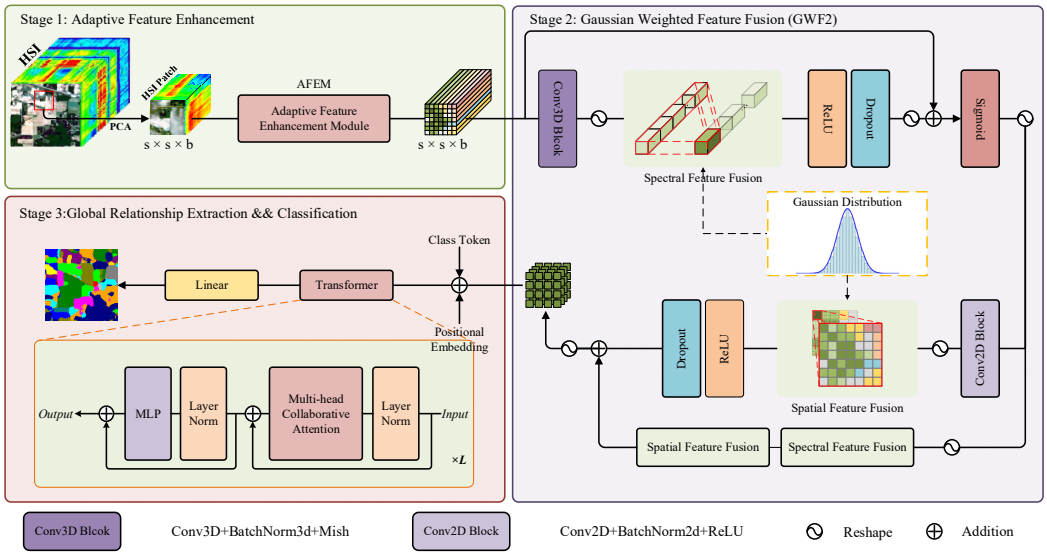

Figure 1. Overall structure of proposed AFGNet. First, the PCA algorithm is used to remove redundant bands. Then, the proposed AFEM module is constructed to distinguish the primary features from the secondary features. Next, the proposed GWF2 module is designed to extract local features with the assistance of global features. Finally, the proposed MHCA module is utilized to extract global features and complete classification prediction.

First, the HSI data $\boldsymbol{I}$ are preprocessed. The HSI data $\boldsymbol{I}$ contain a vast amount of spectral bands, offering valuable spectral information while introducing redundancy. To mitigate this, PCA [51] is applied to reduce the dimensionality of the original HSI data, thereby decreasing computational complexity and preserving essential spectral features. The HSI data after PCA dimension reduction are denoted as $\boldsymbol{I}^{\text{pca}} \in \mathbb{R}^{h \times w \times b}$, where b is the number of spectral bands after PCA. Next, 3D-patch extraction is performed on the HSI data $\boldsymbol{I}^{\text{pca}}$. Each patch $\boldsymbol{P} \in \mathbb{R}^{s \times s \times b}$ covers a window size of $s \times s$. The central pixel of each patch is set to $x_c \in \mathbb{R}^{1 \times 1 \times b}$. The true label of a patch is determined by the label of its central pixel.

2.2. AFEM Module

The features with strong correlation in the input feature map are considered as the primary features. The features other than these are considered as secondary features. In the field of HSIC, the phenomenon of different objects with similar spectra leads to the existing attention mechanisms inevitably introducing noise from inter-class spectral similarities while enhancing the representation of primary spectral features of target objects. Moreover, the coexisting phenomenon of the same object with different spectra further exacerbates the issue of feature confusion. Due to the suppression effect of attention mechanisms on secondary discriminative features, it is difficult for the model to effectively capture spectral variations within the same category of objects caused by factors such as lighting conditions and phenological changes. To address this issue, an AFEM is proposed, with its detailed structure shown in Figure 2.

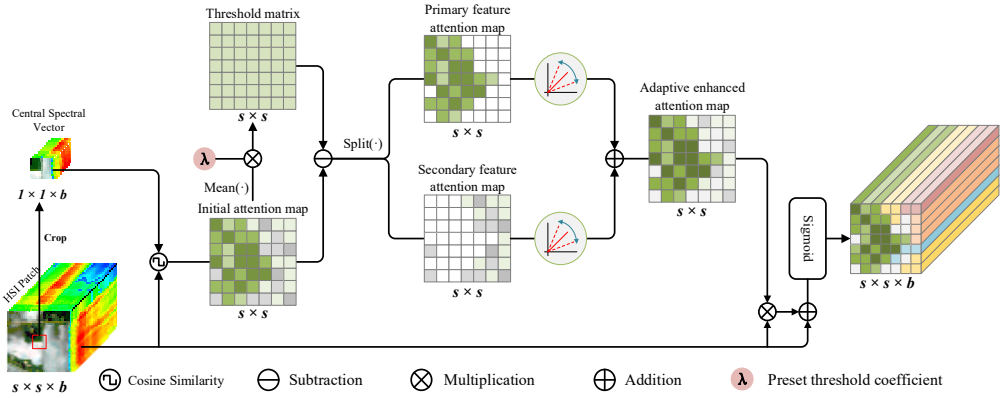

Figure 2. Structure of AFEM. First, the central pixel is extracted from the input spectral cube. After that, the similarity matrix between the surrounding pixels and the central pixel is obtained using cosine similarity. Next, the similarity matrix is split into primary and secondary features by a preset threshold and multiplied with the adaptive coefficient. Finally, the attention map is merged and the original feature map is weighted.

Firstly, the central spectral pixel x_c is extracted from the input patch $\boldsymbol{P}$. Next, cosine similarity is employed to assess the similarity between the central pixel and the surrounding pixels in patch $\boldsymbol{P}$, resulting in an initial attention map $\mathbf{M} \in \mathbb{R}^{s \times s}$. This process can be represented as

$$\mathbf{M} = F_{\cos}(x_c, \boldsymbol{P}) = \frac{x_c \times x_{i,j}}{\|x_c\| \times \|x_{i,j}\|} \tag{1}$$

Here, $x_{i,j}$ denotes the surrounding pixel, where $0 < i, j < s$. $F_{\cos}$ represents the cosine similarity function.

Then, a threshold matrix with the same shape as the initial attention map $\mathbf{M}$ is constructed. The value of this matrix is determined by the product of the mean of $\mathbf{M}$ and the preset threshold coefficient λ. Next, based on this threshold matrix, the initial attention map is divided into a primary feature attention map $\mathbf{M}'_{\mathrm{h}}$ and a secondary feature attention map $\mathbf{M}'_{\mathrm{l}}$. Following this, an adaptive adjustment strategy is adopted to process the two types of attention maps. Specifically, we introduce two parameters, α and β, and incorporate them into the backpropagation process of the network.

By calculating gradients, we can quantify the contribution of different features to model learning. The positive gradient indicates that this feature contributes to the improvement of model performance, thereby prompting the model to pay higher attention to it in subsequent training.

Finally, the two types of attention maps are fused again to obtain an adaptively enhanced attention map, denoted as $\mathbf{M}''$. This process can be represented as

$$\mathbf{M}'_{\mathrm{h}}, \mathbf{M}'_{\mathrm{l}} = \mathrm{Split}(\mathbf{M} - (\lambda \times \mathrm{AvgPool}(\mathbf{M}))) \tag{2}$$

$$\mathbf{M}'' = \alpha \cdot \mathbf{M}'_{\mathrm{h}} + \beta \cdot \mathbf{M}'_{\mathrm{l}} \tag{3}$$

Here, α and β are two parameters with gradients. Based on attention map $\mathbf{M}''$, the weighted feature map $\boldsymbol{X} \in \mathbb{R}^{s \times s \times b}$ can be expressed as

$$\boldsymbol{X} = \sigma\left(\mathbf{M}'' \otimes \boldsymbol{P} + \boldsymbol{P}\right) = \frac{1}{1 + \mathrm{e}^{-(\mathbf{M}'' \otimes \boldsymbol{P} + \boldsymbol{P})}} \tag{4}$$

Here, σ denotes the Sigmoid function and $\otimes$ denotes the element-wise multiplication.

2.3. GWF2 Module

The limited receptive field of convolution kernels means that each kernel can only capture local information within its direct area of effect, which restricts the network's ability to capture global information. To address this issue, a Gaussian weighted feature fusion module (GWF2) was designed, and its detailed structure is shown in Figure 1.

The data enhanced by AFEM will be used as input for this section. Firstly, $\boldsymbol{X}$ are processed by 3D convolutional block to jointly extract "spatial-spectral" features. The convolution outputs feature maps with c channels. These feature maps will be further abstracted and fused in subsequent operations to obtain higher-level representations. Afterwards, the channel dimension and spectral dimension are merged, denoted as $\boldsymbol{X}' \in \mathbb{R}^{s \times s \times q}$, where $q = b \times c$. This process can be expressed as

$$\boldsymbol{X}' = \mathrm{Reshape}\left(F_{\tau}\left(F_{\mathrm{BN}}\left(F'''_{3\times3}(\boldsymbol{X})\right)\right)\right) \tag{5}$$

Here, $F'''_{3\times3}$ represents a 3D convolutional layer with a kernel size of 3, F_{BN} represents a batch normalization layer, and F_{τ} represents the Mish function.

We then model the relationships between channels using a mapping operation, allowing features from each channel to be weighted by features from other channels. Subsequently, we fuse the original input with the weighted result and normalize the fused features using a Sigmoid function. The result is denoted as $\boldsymbol{A} \in \mathbb{R}^{s \times s \times q}$. This process can be expressed as

$$\boldsymbol{X}'' = \mathrm{Dropout}\left(\mathrm{ReLU}\left(F_{\mathrm{spe}}\left(\boldsymbol{X}'\right)\right)\right) \tag{6}$$

$$\boldsymbol{A} = \mathrm{Reshape}\left(\frac{1}{1 + \mathrm{e}^{-(\boldsymbol{X} + \boldsymbol{X}'')}}\right) \tag{7}$$

Here, $\boldsymbol{X}''$ represents the data after global spectral fusion. F_{spe} represents a linear layer initialized with a Gaussian distribution [52]. Next, 2D convolutional block is used to extract spatial features from the feature map. Then, the spatial dimension is flattened. The result is denoted as $\boldsymbol{B}' \in \mathbb{R}^{p \times q}$. This process can be represented as

$$\boldsymbol{B} = \mathrm{ReLU}\left(F_{\mathrm{BN}}\left(F''_{3\times3}(\boldsymbol{A})\right)\right) \tag{8}$$

$$\boldsymbol{B}' = \mathrm{Flatten}(\mathrm{Dropout}(\boldsymbol{B})) \tag{9}$$

Here, $\boldsymbol{B}$ represents the features after convolution processing, and $\boldsymbol{B}'$ represents the data after $\boldsymbol{B}$ is flattened. $F''_{3\times3}$ represents a 2D convolutional layer with a kernel size of 3.

Similarly, we model the relationships between spatial positions using a mapping operation, enabling features at each position to be weighted by features from other positions. Following this, we perform two feature fusions on the original input and merge the fused

result with the weighted result. The result is denoted as $\boldsymbol{T} \in \mathbb{R}^{p \times p}$. This process can be represented as

$$\boldsymbol{B}'' = \text{Dropout}\big(\text{ReLU}\big(\big(F_{\text{spa}}\big(\boldsymbol{B}'\big)\big)\big)\big) \tag{10}$$

$$\boldsymbol{A}' = \text{Reshape}\big(F_{\text{spa}}\big(F_{\text{spe}}(\boldsymbol{A})\big)\big) \tag{11}$$

$$\boldsymbol{T} = \text{Reshape}\big(\boldsymbol{A}' + \boldsymbol{B}''\big) \tag{12}$$

Here, $\boldsymbol{B}''$ represents the result after global spatial fusion, and $\boldsymbol{A}'$ represents the skip connection data after global spectral and spatial fusion. F_{spa} adopts the same weight initialization as F_{spe}.

GWF2 extracts the local structure of feature maps through convolution, while establishing dependencies between local features using mapping operations, thereby achieving more effective feature extraction.

2.4. MHCA Mechanism

By dividing the sequence into multiple heads and computing attention maps in parallel, multi-head self-attention (MHSA) can obtain multiple forms of representations, but the sequence information that each head focuses on is limited. To alleviate this problem, a multi-head collaborative attention (MHCA) mechanism is proposed, and its detailed structure is shown in Figure 3.

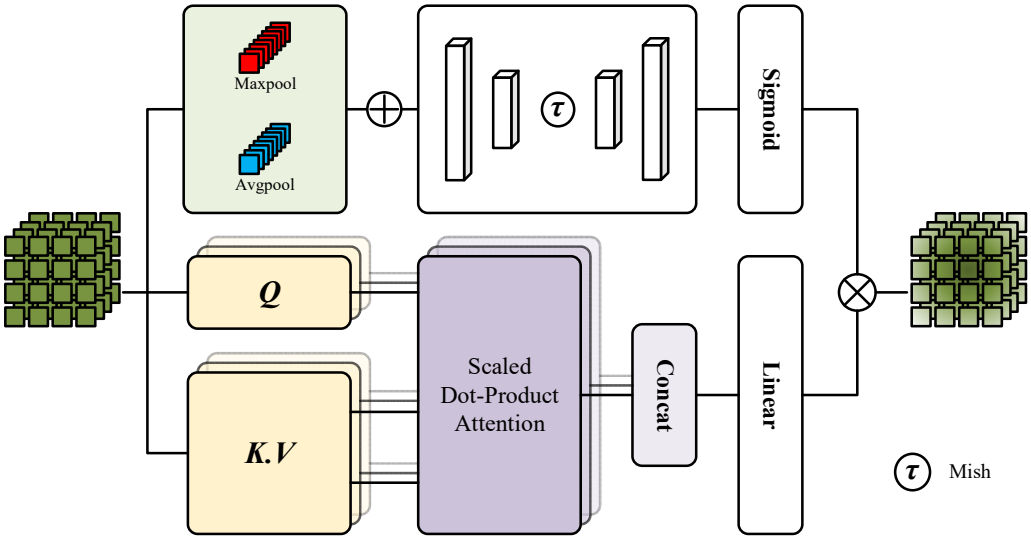

Figure 3. Structure of MHCA. The input sequence is divided into two branches. First, the first branch uses maximum pooling and mean pooling to obtain the attention map. Then, the second branch maps the ***Q*** branch and the ***K.V*** branch independently and uses scaled dot product attention to obtain the weighted feature map. Finally, the attention map of the first branch is used to re-weight the result of the second branch to obtain the final feature map.

The sequence $\boldsymbol{T}$ processed by GWF2 will be used as the input for this section. We concatenate this sequence with a learnable semantic token $\boldsymbol{T}^{\text{cls}}$, which will be used for the final classification prediction. Next, mark the position information of each element in the sequence through positional encoding. Afterwards, layer normalization (LN) is utilized to ensure the stability of the feature distribution. This result is recorded as $\boldsymbol{T}_{\text{in}} \in \mathbb{R}^{p+1 \times p}$. This process can be expressed as

$$\boldsymbol{T}_{\text{in}} = F_{\text{LN}}(\text{Concat}(\boldsymbol{T}^{\text{cls}}, \boldsymbol{T}_1, \boldsymbol{T}_2, \ldots, \boldsymbol{T}_{\text{p}}) + \text{PE}) \tag{13}$$

Here, F_{LN} denotes the LN layer and PE represents the positional embedding.

To preserve more original information in the Query, we adopt an independent mapping strategy, where Key and Value are segmented from the mapping result of the same linear layer. Next, we perform a dot-product operation between Query and the transpose of Key, and then scale by the square root of the dimension of Key. Following this, the Softmax function is employed to convert the results into attention scores. Finally, we perform a matrix multiplication between the attention scores and Value to obtain the output of a single head. By concatenating the outputs of all heads, the result is denoted as $\boldsymbol{T}' \in \mathbb{R}^{p+1\times p}$. This process can be represented as follows:

$$\boldsymbol{Q}, (\boldsymbol{K}, \boldsymbol{V}) \leftarrow \text{Linear}(\boldsymbol{T}_{\text{in}}), \text{Linear}(\boldsymbol{T}_{\text{in}}) \tag{14}$$

$$SA = \text{Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{Softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_K}}\right)\boldsymbol{V} \tag{15}$$

$$\boldsymbol{T}' = \text{Multi-head}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{Concat}(SA_1, SA_2, \cdots, SA_{\text{h}}) \cdot \mathbf{W} \tag{16}$$

Here, SA represents the weighted result with an attention head. $\mathbf{W}$ represents the parameter matrix. To better capture the global dependencies in the sequence, we use global max pooling and global average pooling to compress the channel dimensions in the original sequence, which can be expressed by the following formula:

$$\boldsymbol{U} = \text{Reshape}(\text{MaxPool}(\boldsymbol{T}_{\text{in}}) + \text{AvgPool}(\boldsymbol{T}_{\text{in}})) \tag{17}$$

The result is denoted as $\boldsymbol{U}$. Then, we project the compressed feature into a low-dimensional space using a fully connected layer and introduce non-linearity with the Mish function. Subsequently, a linear transformation is applied to restore the original dimension, and a sigmoid gating function is employed to obtain attention weights, thereby enabling fine-grained modeling of global information. Finally, we perform an element-wise multiplication of sequence and attention weights, resulting in the final output. This part is inspired by the work of SE, and the process can be represented as:

$$\boldsymbol{U}' = \text{Linear}(F_\tau(\text{Linear}(\boldsymbol{U}))) \tag{18}$$

$$\boldsymbol{T}'' = \boldsymbol{T}' \otimes \text{Reshape}\left(\frac{1}{1+\text{e}^{-\boldsymbol{U}'}}\right) \tag{19}$$

After these operations, each attention head not only captures more sequence information but also undergoes a quadratic re-weighting process involving all sequence elements, enabling the model to capture global dependencies more accurately. Subsequently, $\boldsymbol{T}''$ undergoes (LN), a multi-layer perceptron MLP, and residual connection, as illustrated in Figure 1. After repeating this encoder block multiple times, we extract $\boldsymbol{T}^{\text{cls}}$ for final classification.

3. Experiments

To verify the effectiveness of the proposed method, extensive experiments were conducted on five publicly available classic datasets. To avoid the randomness of the experiment, the average results of 10 repeated experiments were used for all experimental data.

3.1. Datasets

To validate the performance of the proposed method, this paper conducted extensive experiments on five publicly available datasets: Indian Pines (Indiana, USA), Pavia University (Pavia, Italy), Houston 2013 (Houston, TX, USA), Longkou (Jingzhou, China), and LaoYuHe (Kunming, China).

The Indian Pines (IP) dataset was imaged in June 1992 in Indiana, USA, and captured by an airborne visible infrared imaging spectrometer (AVIRIS) (NASA, Washington, DC, USA). The size of the dataset is 145 × 145, with a spatial resolution of approximately 20 m and a wavelength range of 0.4–2.5 nm. It contains 16 ground object categories and 224 continuous bands, of which 20 absorbing bands (104–108, 150–163, 220) are removed, leaving 200 bands for training.

The Pavia University (UP) dataset was acquired during a flight over Pavia, northern Italy, using the ROSIS sensor. The dataset has a size of 610 × 340, with a spatial resolution of 1.3 m, and contains 9 ground-object categories and 103 bands.

The Houston 2013 (HT) dataset [53] was imaged in Houston, Texas, and surrounding rural areas in the United States, using the compact airborne spectrographic imager (CASI) 1500 sensor (ITRES, Calgary, AB, Canada). The dataset has a size of 349 × 1905, with a spatial resolution of 2.5 m, and contains 15 ground object categories and 144 bands.

The LongKou (LK) dataset [54,55] was imaged on 17 July 2018 in Longkou Town, Hubei Province, China, using a DJI M600 Pro drone (DJI, Shenzhen, China) equipped with an 8 mm hyperspectral sensor. The size of this dataset is 550 × 400, with a band range of 400–1000 nm and a spatial resolution of 0.463 m. It contains six crop categories and 270 bands.

The LaoYuHe (LYH) dataset [56] was imaged on May 2024 in Laoyuhe Wetland Park, Kunming City, Yunnan Province, China, and collected by OHS satellite. The size of this dataset is 391 × 591, with a wavelength range of 0.4 to 1 μm and a spatial resolution of 10 m. It contains 8 land cover categories and 32 bands.

Tables 1 and 2 list the land cover class names, the number of training samples, and the number of testing samples for each dataset. For the Indian Pines and Houston 2013 datasets, the number of training samples is 5% of the total samples; for the Pavia University and LaoYuHe datasets, it is 1%; and for the Longkou dataset, it is 0.2%.

Table 1. Training and test samples numbers for Indian Pines, Pavia University, and Houston 2013.

No	Indian Pines			Pavia University			Houston 2013		
	Class Name	**Training**	**Test**	**Class Name**	**Training**	**Test**	**Class Name**	**Training**	**Test**
1	Alfafa	2	44	Asphalt	86	6545	Healthy Grass	63	1188
2	Corn N.	71	1357	Meadows	186	18,463	Stressed Grass	63	1191
3	Corn M.	41	789	Gravel	21	2078	Synthetic Grass	35	662
4	Corn	12	225	Trees	31	3033	Trees	62	1182
5	Grass P.	24	459	Painted M. S.	13	1332	Soil	62	1180
6	Grass T.	37	693	Bare soil	50	4979	Water	16	309
7	Grass P. M.	1	27	Bitumen	13	1317	Residential	63	1205
8	Hay W.	24	454	S. B. B.	37	3645	Commerical	62	1182
9	Oats	1	19	Shadows	10	937	Road	63	1189
10	Soybean N.	49	923		427	42,349	Highway	61	1166
11	Soybean M.	123	2332				Railway	62	1173
12	Soybean C.	30	563				Parking Lot 1	62	1171
13	Wheat	10	195				Parking Lot 2	23	446
14	Woods	63	1202				Tennis Court	21	407
15	B. G. T. D.	19	367				Running Track	33	627
16	S. S. T.	5	88						
-	Total	512	9737	Total	427	42,349	Total	751	14,278

Table 2. Training and test samples numbers for Longkou and LaoYuHe.

No	Longkou			LaoYuHe		
	Class Name	**Training**	**Test**	**Class Name**	**Training**	**Test**
1	Corn	69	34,442	Metasequoia	55	5452
2	Cotton	17	8357	Other Tree Species	29	2853
3	Sesame	6	3025	Greenhouse Farmland	67	6599
4	Broad-leaf soybean	126	63,086	Bare Land	21	2135
5	Narrow-leaf soybean	8	4143	Water Bodies	77	7625
6	Rice	24	11,830	Buildings	24	2432
7	Water	134	66,922	Asphalt	68	6735
8	Roads and houses	14	7110	Pitches	2	166
9	Mixed weed	11	5218			
-	Total	409	204,133		343	33,997

3.2. Experiment Details

1. The proposed method was implemented using PyTorch 1.10.1 and its performance was evaluated on an NVIDIA GeForce RTX 3090 GPU (NVIDIA, Santa Clara, CA, USA).
2. This paper uses three common evaluation indicators, namely overall accuracy (OA), average accuracy (AA), and Kappa coefficient ($\kappa \times 100$), to compare the performance of all methods.
3. To ensure the fairness of the experiment, all compared methods followed the optimal parameter configuration in their respective papers. Figure 4 shows the impact of different patches and learning rates on the classification performance of the proposed method. For the IP and UP datasets, the patch size was set to 13×13, and the learning rate was set to 1×10^{-3}. For the HT dataset, the patch size was set to 13×13, and the learning rate was set to 5×10^{-4}. For the LK dataset, the patch size was set to 15×15, and the learning rate was set to 5×10^{-3}. For the LYH dataset, the patch size was set to 15×15, and the learning rate was set to 5×10^{-4}. In addition, the StepLR strategy was adopted to adjust the learning rate during training, that is, the learning rate was multiplied by 0.9 for every tenth of the total training epoch. Finally, the network was trained using the Adam optimizer, and the training epoch and batch size were set to 100 and 64, respectively. The threshold coefficient λ in the AFEM module was set to 1.05.

3.3. Ablation for GWF2

In Section 2.3, GWF2 achieves effective extraction of spectral and spatial features by combining a linear layer with a Gaussian weight of the convolution kernel. To verify the impact of spectral and spatial feature fusion on model performance, this section designs a set of comparative experiments. The experimental results are shown in Figure 5. When no feature fusion mechanism is included, i.e., GWF2 consists only of basic convolutional layers, the model's classification performance on all datasets is at the lowest level. When any feature fusion mechanism is introduced, the classification performance of the model is significantly improved. Furthermore, when the GWF2 module adopts a complete structure, the model achieves the best classification effect. This experimental result demonstrates the effectiveness of spectral/spatial feature fusion.

3.4. Ablation for AFEM and MHCA

This section verifies the effectiveness of AFEM and MHCA. Taking the GWF2 module as the baseline model, the changes in model performance were compared by removing different components. The experimental results are shown in Table 3.

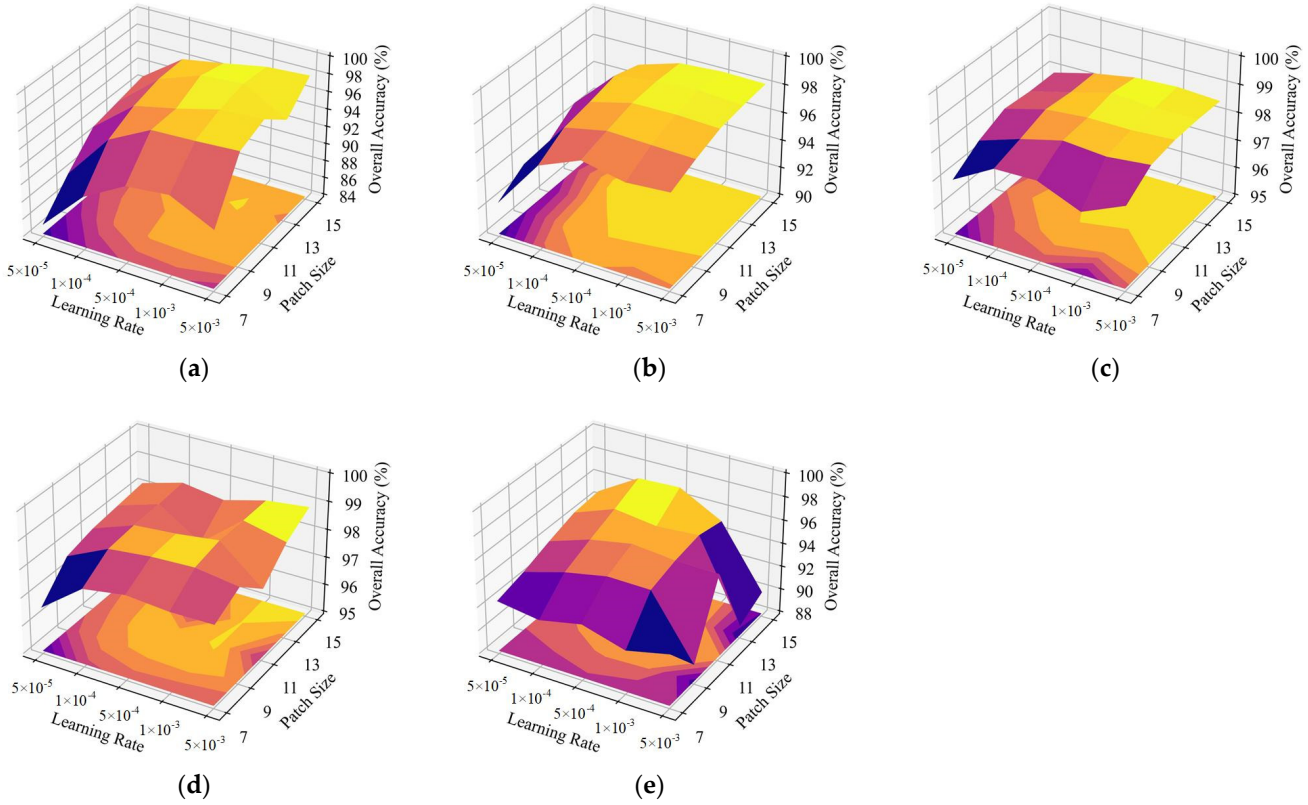

Figure 4. Impact of different patch sizes and learning rates on classification performance, (**a**–**e**): Indian Pines, Pavia University, Houston 2013, Longkou, and LaoYuHe. For the IP and UP datasets, the patch size was set to 13 × 13, and the learning rate was set to 1 $\times 10^{-3}$. For the HT dataset, the patch size was set to 13 × 13, and the learning rate was set to 5 $\times 10^{-4}$. For the LK dataset, the patch size was set to 15 × 15, and the learning rate was set to 5 $\times 10^{-3}$. For the LYH dataset, the patch size was set to 15 × 15, and the learning rate was set to 5 $\times 10^{-4}$.

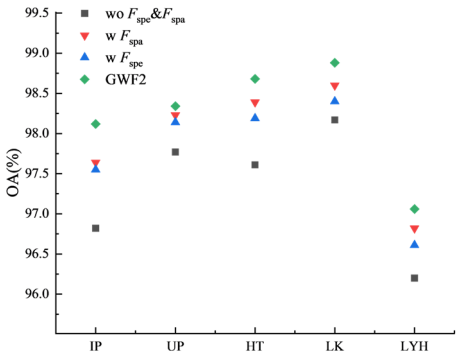

Figure 5. Ablation for GWF2. F_{spe} and F_{spa} represent spectral feature fusion and spatial feature fusion, respectively. Markings in the upper left corner represent, from top to bottom, not using F_{spe} and F_{spa}, only using F_{spa}, only using F_{spe}, and complete GWF2 module (including F_{spe} and F_{spa}).

Table 3. Ablation experiments. (OA is adopted as evaluation indicator, the optimal performance is bolded).

Case	Components			Dataset				
	AFEM	GWF2	MHCA	IP	UP	HT	LK	LYH
1	✘	✔	✘	97.27	97.94	98.17	97.99	96.31
2	✔	✔	✘	97.78	98.26	98.56	98.72	96.94
3	✘	✔	✔	97.60	98.28	98.58	98.58	96.68
4	✔	✔	✔	**98.12**	**98.34**	**98.68**	**98.88**	**97.06**

1. In the proposed AFEM module, we achieved adaptive enhancement of features by differentially processing different features. In Case 1 of Table 3, the model only contained the GWF2 module, and the classification performance of the model was the lowest at this time. In Case 2, AFEM was introduced into the model, and it can be observed that the classification performance of the model has been significantly improved. Compared with Case 1, the classification performance of the model at this time improved by 0.49%, 0.32%, 0.39, 0.71%, and 0.63%, respectively, on all datasets. This result strongly proves the effectiveness of AFEM.
2. In the proposed MHCA module, the global dependencies in the features were fully captured through the collaborative work of multiple attentions. In Case 3 of Table 3, the MHCA module was introduced into the model. At this time, the model obtained performance improvements comparable to Case 2. OA was improved by 0.33%, 0.40%, 0.51%, 0.89%, and 0.75% on all datasets, respectively. This result proves the effectiveness of MHCA. In addition, in Case 4, the model included all three modules. At this time, the classification performance of the model was also the best, which proves that the proposed modules can work together effectively.

3.5. Analysis of Learnable Parameters in AFEM

In the AFEM module, by introducing learnable parameters α and β, the input features are differentiated. To verify the effectiveness of the mechanism, 5 sets of experiments were conducted, and the results are shown in Figure 6, where α corresponds to the primary feature and β corresponds to the secondary feature.

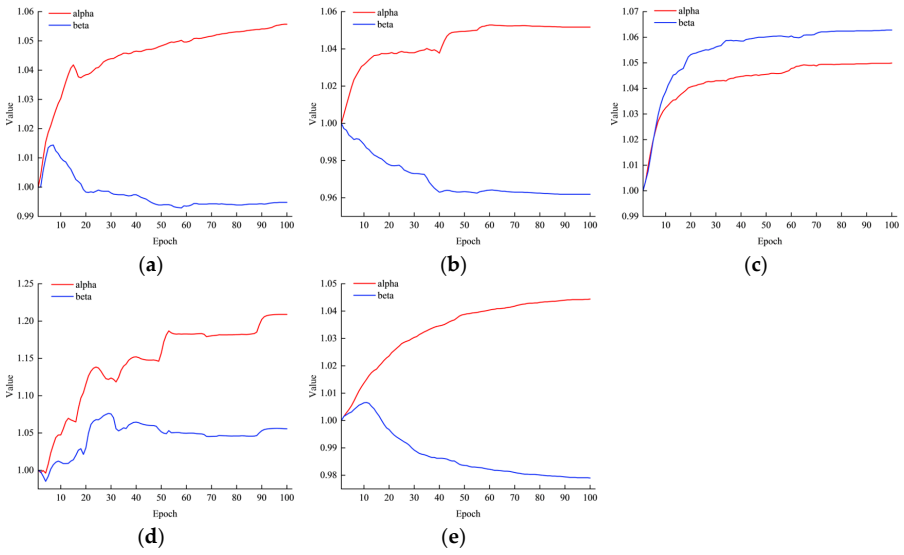

Figure 6. Impact of different patch sizes and learning rates on classification performance, (**a**–**e**): Indian Pines, Pavia University, Houston 2013, Longkou, and LaoYuHe. Alpha and beta represent adaptive scaling coefficients of primary and secondary features, respectively.

It can be observed that the change trends of these parameters show obvious differences in different datasets. Specifically, on the IP and UP datasets, parameter α showed an upward trend, while parameter β related to the secondary feature gradually decreased, which is consistent with the performance of most attention mechanisms. However, the situation is different on the HT dataset. Compared with parameter α related to the primary feature, parameter β obtained a higher weight, indicating that on this dataset, the secondary features that are beneficial to the classification performance of the model were enhanced.

In addition, on the LK dataset, although parameter α obtained a significant gain, parameter β was not overly suppressed, indicating that on this dataset, the model effectively

utilizes the important information in the secondary features while maintaining attention to the primary features. Finally, the change trends of these two parameters on the LYH dataset were similar to those on the IP dataset.

These experimental results not only demonstrate that the AFEM module can flexibly process and enhance the most beneficial features according to the characteristics of the current input features, but also emphasize the importance of secondary features in the model learning process because they also contain information that is crucial to improving model performance.

3.6. Quantitative Evaluation

This section compares the proposed AFGNet with some state-of-the-art HSIC methods, including HybridSN [24], SSSAN [35], SPRN [23], CVSSN [26], GAHT [43], ViT [37], SF [42], SSTN [44], SSFTT [46], CAN [49], and HybridKAN [50]. The experimental results are shown in Tables 4–8. Among them, HybridSN, SSSAN, SPRN, and CVSSN adopt CNN-based frameworks; GAHT, SF, and SSTN are based on Transformer; and SSFTT and the proposed method integrate CNN and Transformer to form a hybrid structure. Particularly, SPRN and CVSSN also utilize the idea of adaptive feature enhancement.

Table 4. Classification performance obtained by different methods for Indian Pines dataset. (The optimal performance of OA, AA, and $\kappa \times 100$ are bolded, No. 1–16 represents accuracy of each category).

No.	HybridSN	SSSAN	SPRN	CVSSN	GHAT	ViT	SF	SSTN	SSFTT	CAN	HybridKAN	Proposed
1	92.10	77.97	99.75	88.63	77.92	13.99	98.57	99.53	59.77	0.93	22.80	88.40
2	86.65	80.38	98.03	93.06	66.72	62.28	63.99	95.04	94.75	71.08	45.83	95.88
3	85.37	80.70	97.29	95.93	59.67	57.82	63.53	97.51	97.13	72.98	42.75	98.25
4	91.35	80.72	97.48	93.56	76.48	49.53	68.07	94.79	91.64	23.01	33.53	96.80
5	93.33	92.96	97.30	94.82	85.23	75.51	88.37	98.08	99.82	90.94	58.72	99.85
6	96.62	95.24	98.20	98.13	78.21	80.26	84.10	98.91	99.24	99.19	72.81	99.49
7	88.00	74.26	95.65	76.07	75.19	54.42	82.99	72.89	97.40	25.60	31.39	97.78
8	94.09	97.15	100	99.10	90.78	84.76	89.56	99.08	99.69	99.74	81.56	99.67
9	48.38	56.26	79.75	78.44	62.42	40.57	73.19	89.24	78.42	60.00	9.42	91.58
10	86.73	85.99	94.82	90.66	69.64	68.16	72.79	93.46	97.01	64.59	39.79	97.26
11	91.17	89.67	98.30	94.51	75.77	69.75	69.93	97.79	99.03	76.47	57.72	98.84
12	89.79	67.95	99.44	88.47	57.86	48.79	51.17	92.86	91.15	47.66	32.76	94.14
13	97.35	94.16	96.82	99.49	83.99	74.15	83.52	98.74	99.94	98.67	8089	99.79
14	97.78	96.16	97.23	96.87	87.66	86.08	88.99	98.42	99.17	95.01	78.35	99.95
15	97.45	85.28	98.52	90.22	68.57	47.16	74.73	94.96	92.99	73.19	44.25	98.26
16	84.09	94.69	91.06	90.47	95.23	91.87	93.94	95.46	86.47	75.51	77.20	99.89
OA	91.07	87.19	97.62	94.13	74.47	68.96	73.47	96.51	97.02	76.89	56.91	**98.12**
AA	88.76	84.35	96.23	91.78	75.71	62.82	78.07	94.80	92.73	67.16	50.61	**97.24**
$\kappa \times 100$	89.79	85.38	97.29	93.31	70.74	64.49	69.31	96.02	96.61	73.57	50.21	**97.86**

Table 5. Classification performance obtained by different methods for Pavia University dataset. (The optimal performance of OA, AA, and $\kappa \times 100$ are bolded, No. 1–9 represents accuracy of each category).

No.	HybridSN	SSSAN	SPRN	CVSSN	GHAT	ViT	SF	SSTN	SSFTT	CAN	HybridKAN	Proposed
1	94.77	93.35	91.17	95.66	85.35	85.36	84.34	97.88	97.20	95.12	62.93	98.21
2	99.16	97.34	99.67	98.61	90.52	88.51	93.81	97.97	99.92	97.86	84.47	99.98
3	89.73	81.99	98.08	87.96	61.52	63.20	69.77	98.13	88.33	0.00	48.13	87.08
4	96.11	98.52	98.68	97.52	96.53	86.71	98.80	95.99	94.08	2.47	76.91	94.88
5	95.68	95.49	99.95	96.64	98.40	93.74	99.05	97.94	99.45	98.80	96.61	99.90
6	99.58	92.68	98.61	96.74	82.45	73.68	89.02	97.96	99.73	13.47	74.20	99.60
7	94.91	86.31	79.10	91.41	71.73	69.24	68.49	99.12	99.93	0.00	44.08	99.97
8	90.99	86.74	87.26	89.73	77.56	78.45	77.43	87.30	92.93	89.08	50.15	96.90
9	91.55	98.96	98.98	97.59	98.01	96.26	99.69	100	93.72	66.87	43.75	97.69
OA	96.68	94.12	96.19	96.20	86.67	83.65	89.12	96.87	97.73	71.42	74.07	**98.34**
AA	94.72	92.38	94.61	94.65	84.68	81.68	86.71	96.92	96.14	51.52	64.58	**97.13**
$\kappa \times 100$	95.59	92.19	94.91	94.96	82.09	78.47	85.42	95.84	97.00	58.86	64.67	**97.80**

Table 6. Classification performance obtained by different methods for Houston 2013 dataset. (The optimal performance of OA, AA, and $\kappa \times 100$ are bolded, No. 1–15 represents accuracy of each category).

No.	HybridSN	SSSAN	SPRN	CVSSN	GHAT	ViT	SF	SSTN	SSFTT	CAN	HybridKAN	Proposed
1	96.85	94.83	97.81	96.93	94.51	91.32	96.18	90.93	97.24	85.62	92.74	99.35
2	98.39	95.16	97.14	97.79	96.70	94.03	97.76	96.73	99.30	83.52	96.30	99.08
3	98.70	96.78	99.93	98.76	97.68	99.49	96.30	99.31	99.68	84.16	99.03	100
4	93.27	95.02	98.29	96.21	94.93	99.02	95.29	99.27	98.79	92.03	94.38	99.78
5	98.85	99.24	99.02	98.97	97.69	96.64	97.82	99.48	100.00	89.25	96.56	100
6	98.29	99.25	99.27	97.03	87.32	96.13	91.06	99.57	96.79	11.20	96.42	98.05
7	87.58	87.78	96.60	91.67	82.94	82.44	85.51	94.18	96.85	78.34	86.42	95.56
8	97.19	92.10	98.82	98.61	88.48	78.11	83.31	96.66	92.65	51.57	88.58	95.12
9	91.80	93.67	95.81	94.60	85.70	74.53	84.54	94.80	94.36	68.86	86.57	99.33
10	95.16	78.82	93.17	94.49	79.30	82.91	83.19	86.04	99.31	51.20	89.18	99.66
11	97.62	96.37	98.39	95.82	82.56	78.81	82.78	97.25	99.57	65.09	90.74	100
12	97.68	85.75	92.35	95.32	83.03	76.94	89.58	94.29	98.19	47.24	87.71	99.12
13	93.14	90.84	96.87	95.83	85.95	62.52	87.58	90.36	94.82	0.22	91.57	93.87
14	99.27	87.13	100	99.03	88.24	91.83	95.42	98.40	100	64.13	94.39	99.95
15	97.36	99.03	100	97.59	95.39	96.12	95.94	97.81	99.44	70.62	96.44	100
OA	95.60	92.03	97.01	96.22	88.94	86.07	90.04	95.00	97.76	68.19	91.72	**98.68**
AA	96.08	92.82	97.56	96.58	89.36	86.72	90.82	95.67	97.80	62.87	92.47	**98.59**
$\kappa \times 100$	95.25	91.39	96.77	95.92	88.03	84.93	89.23	94.60	97.57	65.51	91.04	**98.57**

Table 7. Classification performance obtained by different methods for Longkou dataset. (The optimal performance of OA, AA, and $\kappa \times 100$ are bolded, No. 1–9 represents accuracy of each category).

No.	HybridSN	SSSAN	SPRN	CVSSN	GHAT	ViT	SF	SSTN	SSFTT	CAN	HybridKAN	Proposed
1	98.46	98.53	98.23	99.19	98.96	94.92	94.78	98.75	99.59	99.38	79.92	99.95
2	94.24	65.14	91.14	94.70	77.27	49.89	66.88	89.27	98.50	97.57	57.52	98.53
3	91.92	84.48	99.17	91.70	75.04	45.05	78.75	88.93	97.41	23.80	69.27	98.21
4	98.25	94.86	98.14	97.28	94.01	85.34	93.98	97.69	99.69	94.00	80.57	99.63
5	87.85	45.45	79.19	91.33	52.82	25.59	47.64	78.86	73.81	0.05	60.79	85.68
6	94.70	96.55	84.67	98.64	96.48	87.97	94.04	99.61	99.09	99.77	79.10	99.25
7	99.49	99.52	99.97	99.66	99.79	98.96	99.86	99.30	99.52	99.99	96.45	99.94
8	89.03	91.23	95.05	90.41	69.65	69.26	64.69	83.32	85.76	82.17	62.45	89.61
9	86.71	82.66	88.48	89.18	91.54	62.78	75.58	84.65	92.95	84.23	43.54	92.42
OA	97.35	94.27	94.15	97.68	93.77	88.18	92.19	96.74	98.38	93.74	84.02	**98.88**
AA	93.41	84.27	92.67	94.68	83.95	68.86	79.58	91.15	94.08	75.66	69.96	**95.91**
$\kappa \times 100$	96.52	92.45	92.61	96.95	91.81	84.28	89.75	95.75	97.87	91.79	78.50	**98.53**

Table 8. Classification performance obtained by different methods for LaoYuHe dataset. (The optimal performance of OA, AA, and $\kappa \times 100$ are bolded, No. 1–8 represents accuracy of each category).

No.	HybridSN	SSSAN	SPRN	CVSSN	GHAT	ViT	SF	SSTN	SSFTT	CAN	HybridKAN	Proposed
1	94.88	94.58	95.99	95.12	90.25	90.23	91.86	97.82	95.19	96.45	76.17	96.70
2	94.19	95.56	98.21	96.49	90.46	88.28	95.25	96.06	94.60	97.12	80.69	94.55
3	97.93	93.38	94.12	94.84	84.19	73.95	77.17	97.44	94.68	91.12	73.64	99.39
4	89.81	72.24	95.71	79.78	62.44	53.13	60.48	92.29	90.48	41.11	43.64	94.07
5	97.68	97.76	99.26	97.08	97.30	96.15	94.90	97.74	98.34	97.62	88.47	99.10
6	85.09	77.69	73.45	82.39	56.83	51.12	59.86	86.24	88.14	50.78	41.61	91.67
7	89.58	84.26	90.83	86.50	69.02	62.67	64.73	85.83	96.82	67.09	53.23	97.47
8	79.74	62.83	80.59	80.89	48.67	38.63	62.29	93.93	60.24	0.12	15.41	66.08
OA	93.88	90.11	93.52	91.86	82.45	77.79	80.74	93.86	95.10	82.71	70.76	**97.06**
AA	91.12	84.79	91.02	89.14	74.89	69.27	75.82	92.17	89.81	67.68	59.11	**92.38**
$\kappa \times 100$	92.64	88.10	92.21	90.20	78.76	73.20	76.64	92.59	94.07	79.06	64.50	**96.46**

Firstly, HybridSN combines 3D convolution and 2D convolution to explore spectral spatial features. However, due to the limited receptive field of traditional convolutional kernels, they did not achieve good classification accuracy on all five datasets. Secondly, SSSAN introduces self-attention mechanisms on top of convolutions, enabling the network to focus on longer-range information. However, its fusion strategy does not fully take effect, resulting in lower classification performance. Afterwards, SPRN divides the input

spectrum into multiple sub-bands by equivalently using grouped convolutions. In addition, this method also adopts cosine similarity as a metric to enhance the features of samples that are more similar to the center vector. It can be observed that its classification performance is better among the comparison methods. Furthermore, CVSSN adopts multiple metrics such as Euclidean distance and cosine similarity to enhance the main features, and it also achieves good classification performance. Furthermore, GAHT and SF fully explore the spectral information in HSI, providing new ideas for HSIC. However, due to the lack of extraction of spatial information, its classification performance is not ideal. ViT is a baseline model for GHAT and SF, and its performance and problems are like those of GHAT and SF. SSTN is a recent Transformer-based HSIC method that effectively explores spectral–spatial information through a structured search framework, achieving good classification performance. Finally, methods with hybrid structures generally outperform those that use only CNNs or only Transformers. SSFTT extracts shallow information through CNN, then fully leverages the ability of Transformer to extract high-level semantic features through feature tokens. These two methods fully utilize the advantages of CNN and Transformer, so they both achieve good classification performance. CAN and HybridKAN are new backbone networks in the HSIC field, which provide new perspectives for research in this field. However, their problems are also obvious. When the sample size of categories is unbalanced, the understanding ability of these two methods for some categories is not ideal.

Although the above methods have achieved good classification performance, the proposed AFGNet still achieves better classification performance than other methods on all five datasets. Specifically, on the IP dataset, compared with SPRN, the method with the highest classification accuracy, the proposed method is 0.5%, 1.01%, and 0.57% higher in OA, AA, and $\kappa \times 100$, respectively; on the UP dataset, compared with SSFTT, the method with the highest classification accuracy, the proposed method is 0.61%, 0.99%, and 0.8% higher in OA, AA, and $\kappa \times 100$, respectively; on the HT dataset, compared with SSFTT, the method with the highest classification accuracy, the proposed method is 0.92%, 0.79%, and 1% higher in OA, AA, and $\kappa \times 100$, respectively; on the LK dataset and LYH dataset, the comparison method with the highest classification accuracy is still SSFTT, but the proposed method still maintains a lead in OA, AA, and $\kappa \times 100$. These experimental results fully demonstrate the superiority of the proposed AFGNet in classification performance.

3.7. Comparison of Classification Maps

To more intuitively display the results of quantitative experiments, this section visualizes the classification maps of all methods on different datasets, and the specific results are shown in Figures 7–11, along with the ground truth maps of each dataset. In CNN-based methods, the influence caused by the limitation of the receptive field of convolutional kernels can be clearly observed, resulting in many mixed and discontinuous patches in the classification maps. This is particularly evident in the IP dataset. Additionally, due to the limitations of its fusion strategy, the visualization results of SSSAN are not satisfactory. SPRN exhibits good visualization results on the IP dataset, but performs averagely on other datasets. Particularly, on the LK dataset, there are relatively obvious misclassifications between the "cotton" (green) category and the "broad-leaf soybean" (yellow) category.

In Transformer-based methods, due to the failure of GAHT and SF to fully utilize spatial information, the classification maps of these two methods contain more noise, especially in the IP and UP datasets. The classification maps of ViT also have many misclassifications similar to salt and pepper noise. In contrast, SSTN achieves more ideal classification maps by utilizing its unique structured search strategy. The classification maps of CAN have a large number of misclassifications. HybridKAN has similar problems to CAN, which is consistent with the results of quantitative experiments.

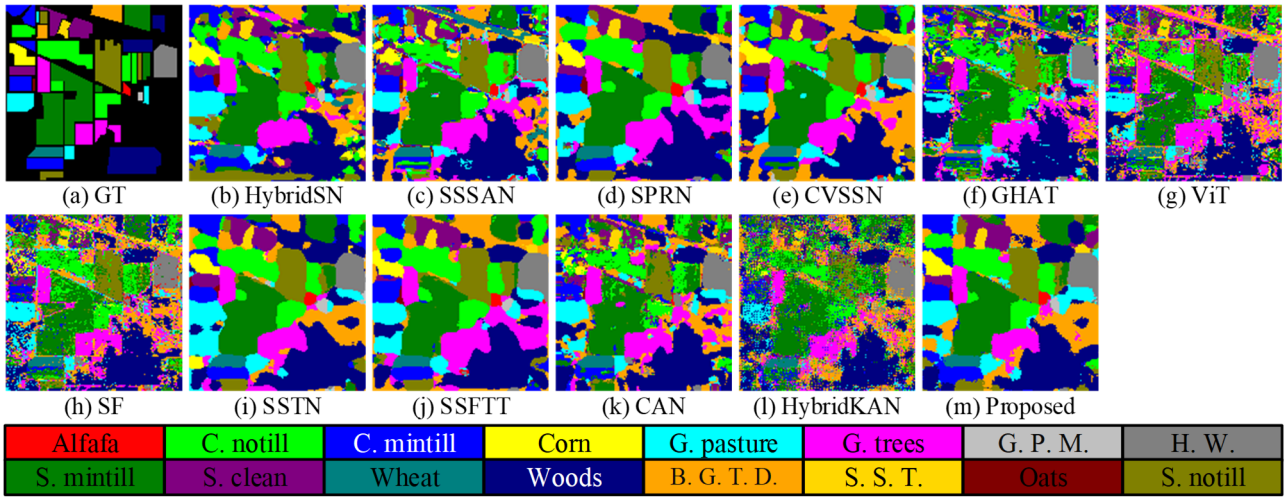

Figure 7. Classification maps obtained by different methods on the IP dataset, (**a**–**m**): Ground Truth, HybridSN, SSSAN, SPRN, CVSSN, GHAT, SF, SSTN, SSFTT, AFGNet, respectively.

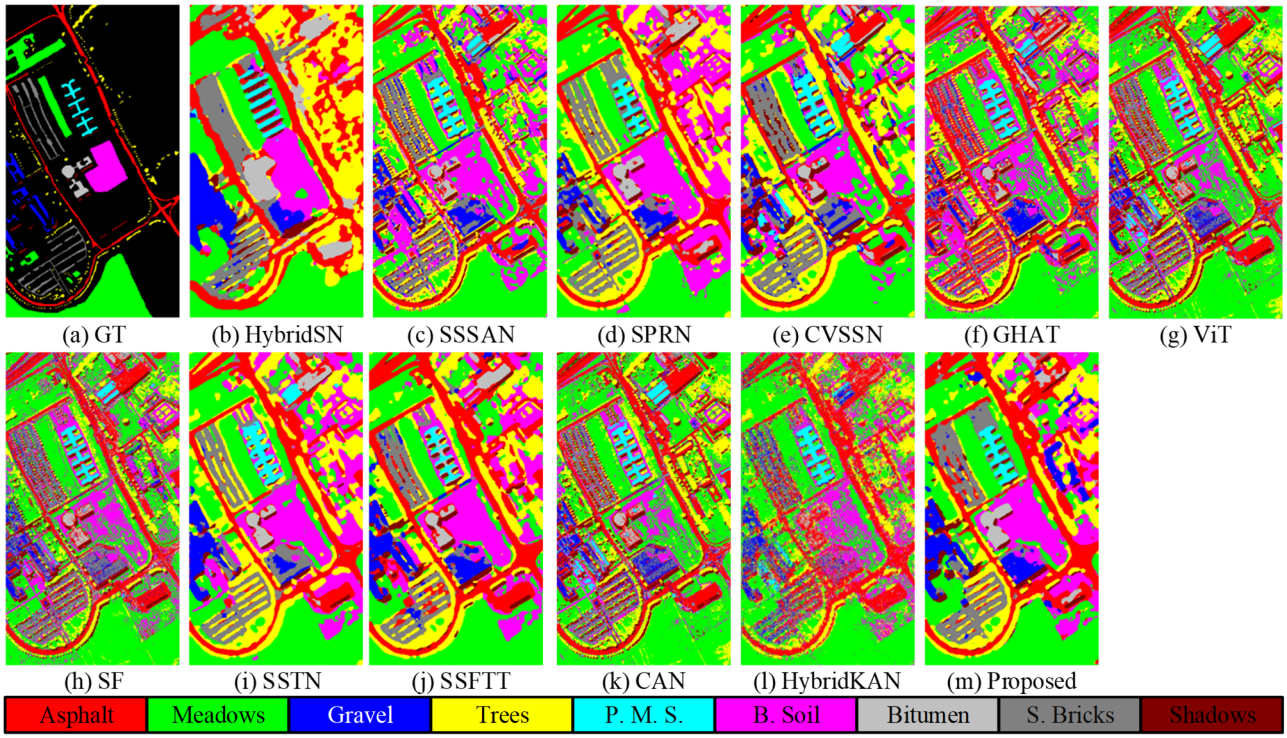

Figure 8. Classification maps obtained by different methods on the UP dataset, (**a**–**m**): Ground Truth, HybridSN, SSSAN, SPRN, CVSSN, GHAT, SF, SSTN, SSFTT, AFGNet, respectively.

As for the methods with hybrid structures, from the perspective of visualization effects, they are generally superior to the above methods, which is consistent with the quantitative results in Tables 4–8. In particular, the proposed AFGNet method has a classification effect that is closer to the ground truth maps, and the frequency and area of discontinuous patches in unlabeled regions are minimized. On the LK dataset, the classification map generated by the AFGNet method has the clearest edges, further verifying the effectiveness and superiority of this method.

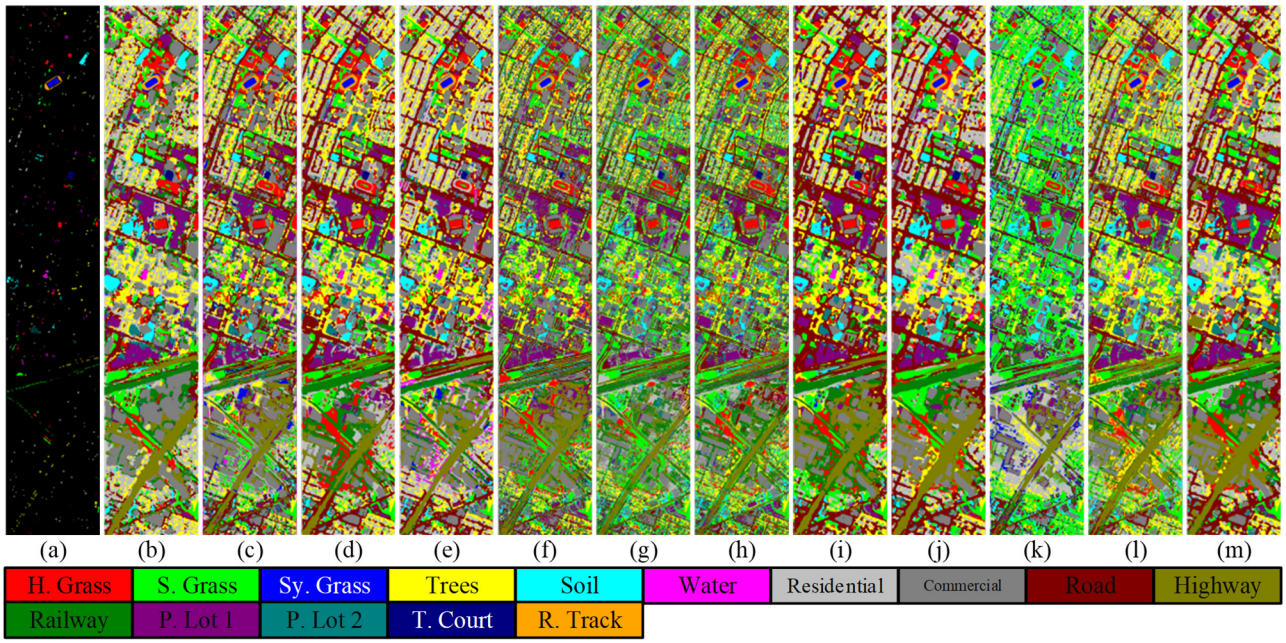

Figure 9. Classification maps obtained by different methods on the HT dataset, (**a**–**m**): Ground Truth, HybridSN, SSSAN, SPRN, CVSSN, GHAT, SF, SSTN, SSFTT, AFGNet, respectively.

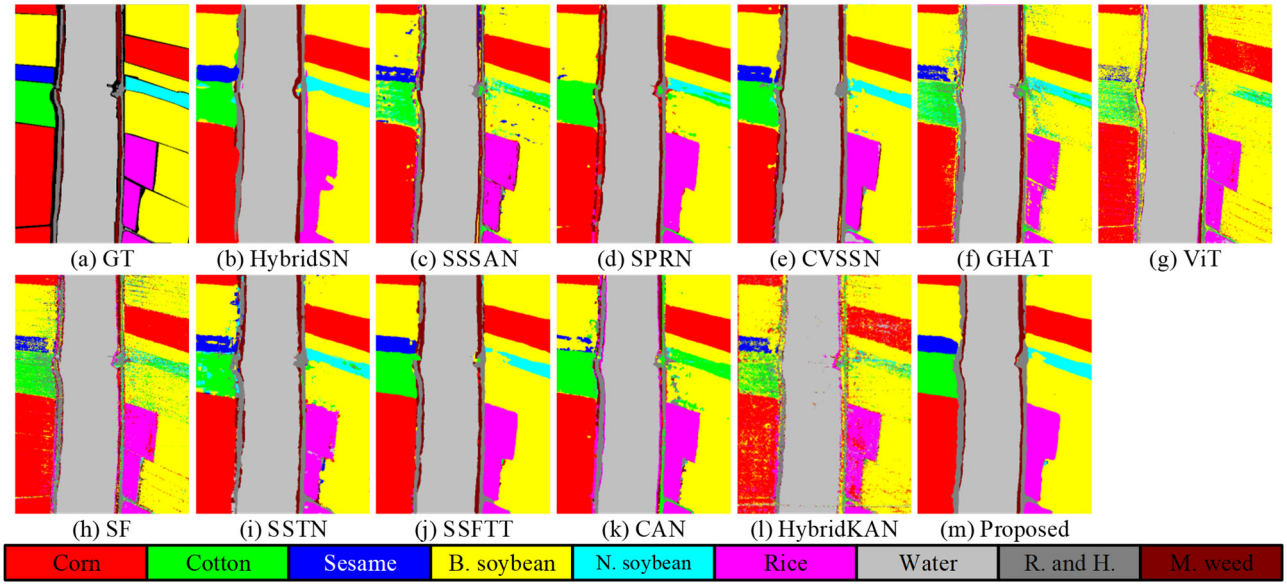

Figure 10. Classification maps obtained by different methods on LK dataset, (**a**–**m**): Ground Truth, HybridSN, SSSAN, SPRN, CVSSN, GHAT, SF, SSTN, SSFTT, and AFGNet, respectively.

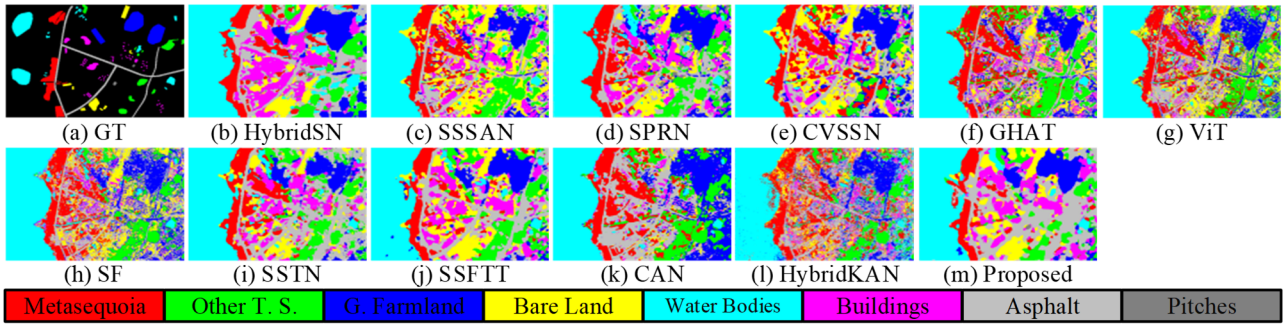

Figure 11. Classification maps obtained by different methods on LYH dataset, (**a**–**m**): Ground Truth, HybridSN, SSSAN, SPRN, CVSSN, GHAT, SF, SSTN, SSFTT, and AFGNet, respectively.

3.8. Comparison of Different Sample Proportions

In this section, OA was used as the evaluation index, and five groups of experiments with different training ratios were conducted on the five datasets of IP, UP, HT, LK, and LYH for the proposed method and some comparison methods. The experimental results are shown in Figure 12. Among them, for the IP and HT datasets, 2.5%, 5%, 7.5%, and 10% of the samples from each category were selected as the training set, respectively; for the UP and LYH datasets, the selected ratios were 1%, 2%, 3%, 4%, and 5%; and for the LK dataset, 0.2%, 0.4%, 0.6%, 0.8%, and 1% of the samples were selected as the training set.

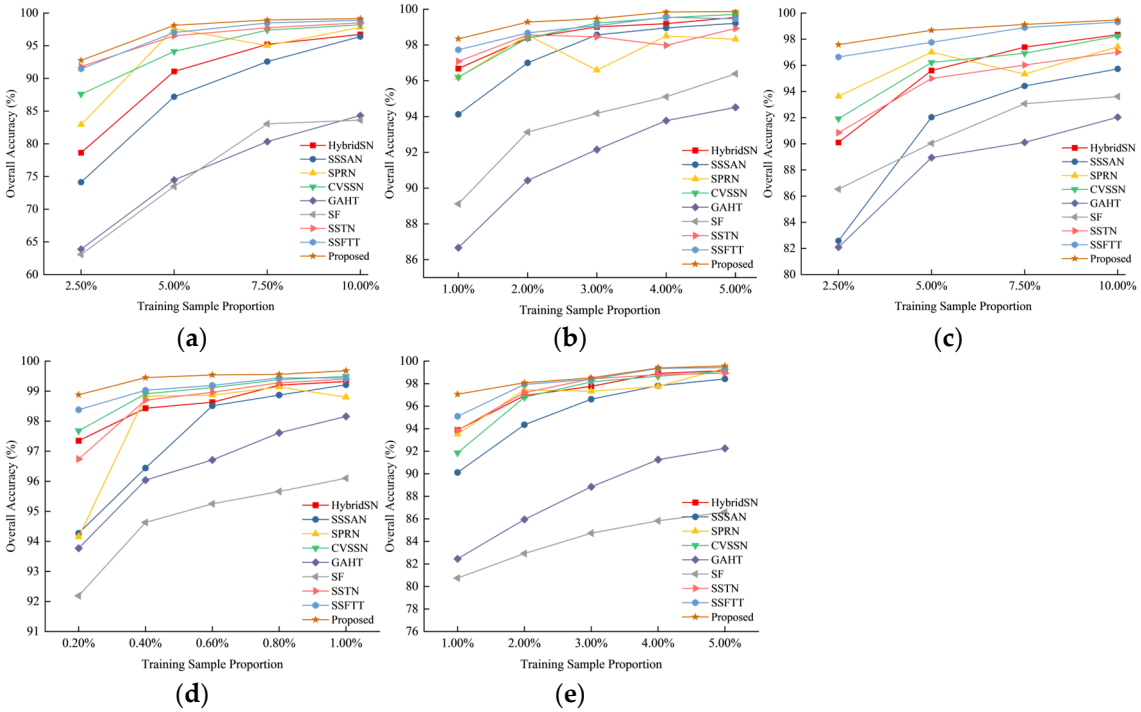

Figure 12. Comparison of different sample proportions; (**a**–**e**) IP, UP, HT, LK, and LYH datasets, respectively.

As the sample size increased, the classification performance of most methods improved, but the performance of individual methods showed unexpected fluctuations. Specifically, due to the failure to effectively utilize spatial information, the performance of GAHT and SF methods was limited, and their performance ranked at the bottom among all five datasets. In contrast, the change curve of SSSAN maintained an upward trend, indicating that its fusion strategy is more suitable for training scenarios with larger sample sizes. The change curves of SPRN showed unexpected fluctuations, especially SPRN, whose classification performance on multiple datasets did not improve despite the increase in sample size. The proposed method and SSFTT both performed stably on all datasets and consistently ranked in the top two. It is particularly noteworthy that on the LK dataset, the proposed method showed significant and sustained performance gains. On all five datasets, the proposed method was able to maintain the optimal classification level, whether in scenarios with extremely limited sample sizes or in situations with relatively ample sample sizes, which fully demonstrates the superiority of the proposed method.

3.9. t-SNE Feature Visualization

To comprehensively evaluate the performance of the proposed method, this section adopts the t-SNE [57] algorithm to visually analyze the feature distributions learned by

AFGNet and three comparison methods (SPRN, SSTN, SSFTT). Test samples from each dataset were selected for display, and the results are shown in Figures 13 and 14.

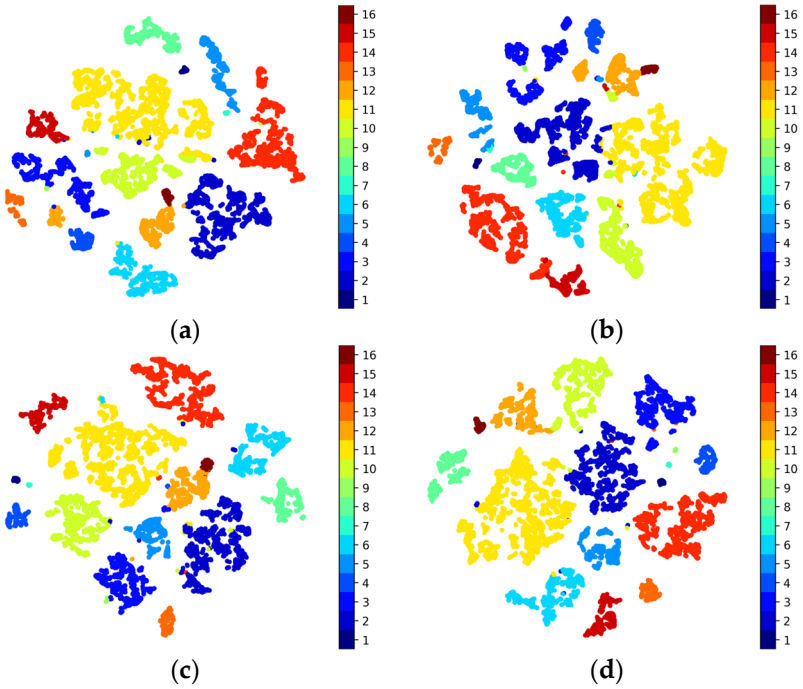

Figure 13. t-SNE visualization results obtained by different methods on IP dataset (**a**–**d**): SPRN, SSTN, SSFTT, Proposed.

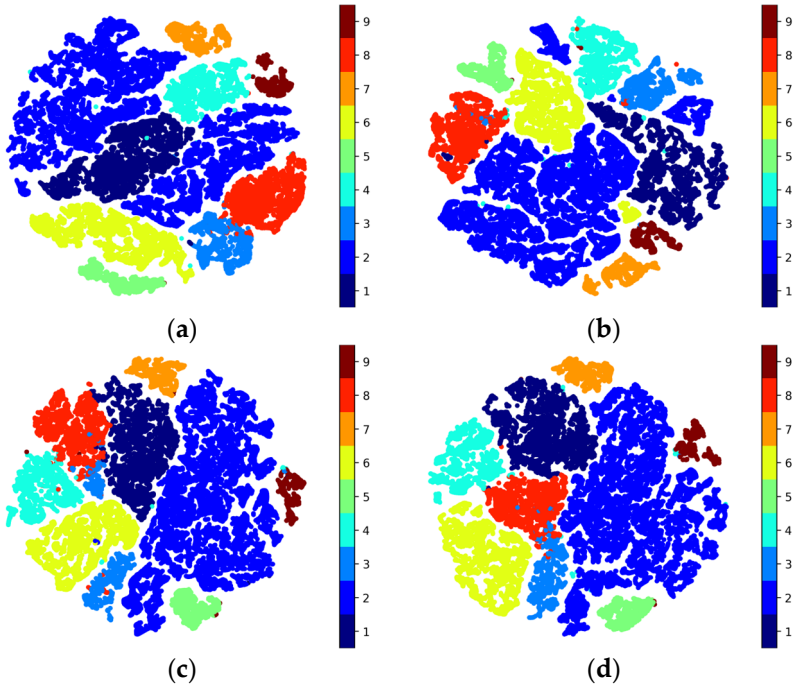

Figure 14. t-SNE visualization results obtained by different methods on UP dataset (**a**–**d**): SPRN, SSTN, SSFTT, Proposed.

Among the comparison methods, SSFTT exhibited the best feature visualization results, while SPRN and SSTN showed limitations to varying degrees. Specifically, SPRN displayed significant intra-class feature dispersion on multiple datasets, while SSTN suffered from inter-class feature confusion. This phenomenon indicates that these two methods have deficiencies in feature discrimination ability.

In contrast, the proposed method demonstrated significant advantages in visualization effects. On the IP dataset, AFGNet achieved the largest distance between different classes and the mildest intra-class dispersion. On the UP dataset, the proposed method also exhibited the lowest level of inter-class confusion, and features of the same class are basically distributed in the same region, unlike the dispersion observed in SPRN's performance on the UP dataset. These visualization results not only intuitively demonstrate the superiority of AFGNet in feature learning but also further validate the effectiveness of this method in improving classification accuracy and discrimination.

3.10. Heat Maps Feature Visualization

In this section, heat maps were utilized to visualize the features learned by the model at different stages. The experimental results are shown in Figure 15. In the heat map, compared with the blue area, the red area means that the model pays more attention to this area.

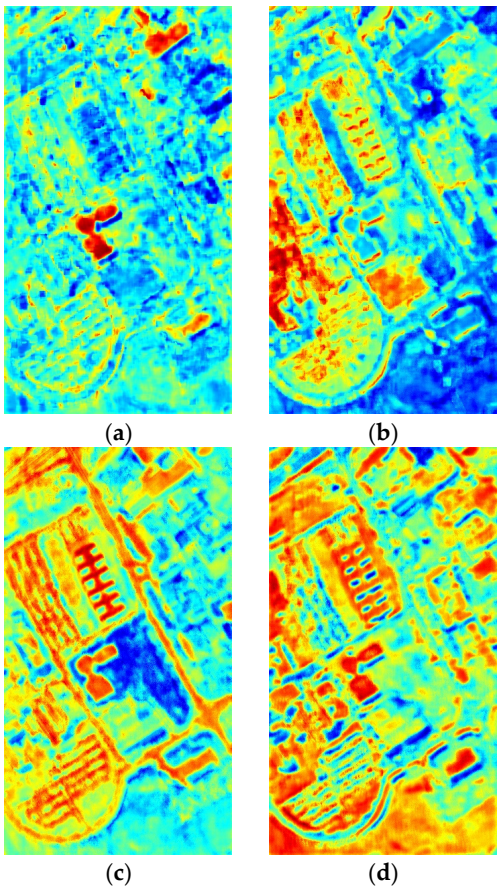

Figure 15. Heat map visualization of features obtained by model at different stages on UP dataset. Compared with the blue area, the red area means that the model pays more attention to this area. (**a**) Baseline. (**b**) AFEM. (**c**) AFEM + GWF2. (**d**) AFGNet.

First, the baseline model was used, which does not adopt any modules proposed in this paper. In Figure 15a, it can be observed that the model does not show obvious areas of interest. Then, the AFEM module was introduced. As seen in Figure 15b, more concentrated thermal responses appear in some areas, indicating that the model paid more attention to these areas. Next, the GWF2 module was further merged, which is shown Figure 15c. At this stage, the model not only paid attention to local structures, but also significantly increased its attention to long-distance features. Finally, the model containing

all modules was adopted. As shown in Figure 15d, the heat map clearly demonstrates that the model has the strongest feature learning ability.

3.11. Robustness

Spectral data are often affected by various degradations, noises, and changes during the imaging process, which can seriously reduce the data quality and thus affect the performance of the classifier. This section verified the robustness of the proposed method in a noisy environment by adding different proportions of Gaussian noise to the UP dataset. The experimental results are shown in Figure 16, and OA is used as the evaluation indicator. It can be observed that with the increase of the noise ratio, the performance of all methods is affected by varying degrees. Among them, the performance of SSRN and SSFTT is most significantly affected. In contrast, SSTN has the least performance degradation after 10%. For the proposed method, its classification accuracy is obviously also affected to a certain extent, but its performance is still better than other methods.

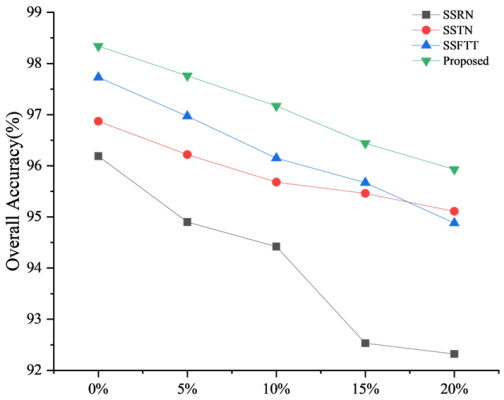

Figure 16. Impact of different proportions of Gaussian noise on each method on UP dataset: SPRN, SSTN, SSFTT, Proposed.

4. Conclusions

This paper proposes an AFGNet method for HSIC task. Firstly, an AFEM module is proposed. By accurately evaluating the impact of different features, AFEM can dynamically enhance the most beneficial features, leading to improved classification performance. Secondly, a GWF2 is proposed. GWF2 not only effectively extracts local detail information but also efficiently fuses information from different channels, thereby assisting the model in more accurately comprehending features. Finally, we propose a MHCA mechanism. By directly interacting with and globally modeling the input sequence, MHCA can more effectively extract key features. Extensive experimental results demonstrate that, compared to some state-of-the-art methods, the proposed method can provide superior classification performance.

Nevertheless, this method still has some limitations. For example, in the GWF2 module, although global context information is obtained through spatial mapping and spectral mapping to assist feature extraction, this process also introduces additional computational costs. In future work, we will build on this work and explore more lightweight network structures.

Author Contributions: Conceptualization, C.S.; Data curation, C.S. and F.Z.; Formal analysis, L.W.; Methodology, C.S.; Software, F.Z.; Validation, H.P. and F.Z.; Writing—original draft, F.Z.; Writing—review and editing, C.S. and L.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grant 42271409, in part by the Science and Technology Plan Project of Huzhou under Grant 2024GZ36, and in part by the Fundamental Research Funds in Heilongjiang Provincial Universities under Grant 145409207.

Data Availability Statement: Data and code are available at: https://github.com/Isee-max/AFGNet (accessed on 8 February 2025).

Acknowledgments: We would like to thank the handling editor and the anonymous reviewers for their careful reading and helpful remarks.

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

References

1. Ghamisi, P.; Yokoya, N.; Li, J.; Liao, W.; Liu, S.; Plaza, J.; Rasti, B.; Plaza, A. Advances in Hyperspectral Image and Signal Processing: A Comprehensive Overview of the State of the Art. *IEEE Geosci. Remote Sens. Mag.* **2018**, *5*, 37–78. [CrossRef]
2. Mei, S.; Geng, Y.; Hou, J.; Du, Q. Learning hyperspectral images from RGB images via a coarse-to-fine CNN. *Sci. China Inf. Sci.* **2021**, *65*, 152102. [CrossRef]
3. Siddique, N.; Paheding, S.; Elkin, C.P.; Devabhaktuni, V. U-Net and Its Variants for Medical Image Segmentation: A Review of Theory and Applications. *IEEE Access* **2021**, *9*, 82031–82057. [CrossRef]
4. Khan, U.; Paheding, S.; Elkin, C.P.; Devabhaktuni, V.K. Trends in Deep Learning for Medical Hyperspectral Image Analysis. *IEEE Access* **2021**, *9*, 79534–79548. [CrossRef]
5. Kamilaris, A.; Prenafeta-Boldú, F.X. Deep learning in agriculture: A survey. *Comput. Electron. Agric.* **2018**, *147*, 70–90. [CrossRef]
6. Yang, G.; Huang, K.; Sun, W.; Meng, X.; Mao, D.; Ge, Y. Enhanced mangrove vegetation index based on hyperspectral images for mapping mangrove. *ISPRS J. Photogramm. Remote. Sens.* **2022**, *189*, 236–254. [CrossRef]
7. Yokoya, N.; Chan, J.C.-W.; Segl, K. Potential of Resolution-Enhanced Hyperspectral Data for Mineral Mapping Using Simulated EnMAP and Sentinel-2 Images. *Remote. Sens.* **2016**, *8*, 172. [CrossRef]
8. Fauvel, M.; Tarabalka, Y.; Benediktsson, J.A.; Chanussot, J.; Tilton, J.C. Advances in Spectral-Spatial Classification of Hyperspectral Images. *Proc. IEEE* **2013**, *101*, 652–675. [CrossRef]
9. Song, W.; Li, S.; Kang, X.; Huang, K. Hyperspectral image classification based on KNN sparse representation. In Proceedings of the 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Beijing, China, 10–15 July 2016; pp. 2411–2414.
10. Melgani, F.; Bruzzone, L. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. Geosci. Remote. Sens.* **2004**, *42*, 1778–1790. [CrossRef]
11. Pal, M.; Foody, G.M. Feature Selection for Classification of Hyperspectral Data by SVM. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 2297–2307. [CrossRef]
12. Yin, H.; Hu, W.; Li, F.; Lou, J. One-step multi-view spectral clustering by learning common and specific nonnegative embeddings. *Int. J. Mach. Learn. Cybern.* **2021**, *12*, 2121–2134. [CrossRef]
13. Yuan, Y.; Lin, J.; Wang, Q. Hyperspectral Image Classification via Multitask Joint Sparse Representation and Stepwise MRF Optimization. *IEEE Trans. Cybern.* **2015**, *46*, 2966–2977. [CrossRef] [PubMed]
14. Zhang, Y.; Li, W.; Zhang, M.; Wang, S.; Tao, R.; Du, Q. Graph Information Aggregation Cross-Domain Few-Shot Learning for Hyperspectral Image Classification. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *35*, 1912–1925. [CrossRef]
15. Li, C.; Sun, W.; Peng, J.; Ren, K. Deep Dynamic Adaptation Network Based on Joint Correlation Alignment for Cross-Scene Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote. Sens.* **2023**, *61*, 5532312. [CrossRef]
16. Zhang, M.; Li, W.; Zhang, Y.; Tao, R.; Du, Q. Hyperspectral and LiDAR Data Classification Based on Structural Optimization Transmission. *IEEE Trans. Cybern.* **2022**, *53*, 3153–3164. [CrossRef] [PubMed]
17. Su, Y.; Chen, J.; Gao, L.; Plaza, A.; Jiang, M.; Xu, X.; Sun, X.; Li, P. ACGT-Net: Adaptive Cuckoo Refinement-Based Graph Transfer Network for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote. Sens.* **2023**, *61*, 5521314. [CrossRef]
18. Shi, C.; Zhang, X.; Wang, L.; Jin, Z. A lightweight convolution neural network based on joint features for Remote Sensing scene image classification. *Int. J. Remote. Sens.* **2023**, *44*, 6615–6641. [CrossRef]
19. Huang, Y.; Peng, J.; Zhang, G.; Sun, W.; Chen, N.; Du, Q. Adversarial Domain Adaptation Network with Calibrated Prototype and Dynamic Instance Convolution for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5514613. [CrossRef]
20. Shi, C.; Chen, J.; Wang, L. Hyperspectral image classification based on a novel Lush multi-layer feature fusion bias network. *Expert Syst. Appl.* **2024**, *247*. [CrossRef]
21. Chen, Y.; Zhao, X.; Jia, X. Spectral–Spatial Classification of Hyperspectral Data Based on Deep Belief Network. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2015**, *8*, 2381–2392. [CrossRef]

22. Chen, Y.; Lin, Z.; Zhao, X.; Wang, G.; Gu, Y. Deep Learning-Based Classification of Hyperspectral Data. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 2094–2107. [CrossRef]
23. Zhang, X.; Shang, S.; Tang, X.; Feng, J.; Jiao, L. Spectral Partitioning Residual Network with Spatial Attention Mechanism for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote. Sens.* **2021**, *60*, 5507714. [CrossRef]
24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
25. 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]
26. Li, M.; Liu, Y.; Xue, G.; Huang, Y.; Yang, G. Exploring the Relationship Between Center and Neighborhoods: Central Vector Oriented Self-Similarity Network for Hyperspectral Image Classification. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *33*, 1979–1993. [CrossRef]
27. Su, Y.; Gao, L.; Jiang, M.; Plaza, A.; Sun, X.; Zhang, B. NSCKL: Normalized Spectral Clustering with Kernel-Based Learning for Semisupervised Hyperspectral Image Classification. *IEEE Trans. Cybern.* **2022**, *53*, 6649–6662. [CrossRef]
28. Mei, S.; Chen, X.; Zhang, Y.; Li, J.; Plaza, A. Accelerating Convolutional Neural Network-Based Hyperspectral Image Classification by Step Activation Quantization. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5502012. [CrossRef]
29. Yin, H.; Hu, W.; Zhang, Z.; Lou, J.; Miao, M. Incremental multi-view spectral clustering with sparse and connected graph learning. *Neural Netw.* **2021**, *144*, 260–270. [CrossRef]
30. Shi, C.; Yue, S.; Wang, L. Attention Head Interactive Dual Attention Transformer for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote. Sens.* **2024**, *62*, 5523720. [CrossRef]
31. Shi, C.; Wu, H.; Wang, L. A Feature Complementary Attention Network Based on Adaptive Knowledge Filtering for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5527219. [CrossRef]
32. Zhang, X.; Zhang, R.; Li, L.; Li, W. Local–Global Cross Fusion Network with Gaussian-Initialized Learnable Positional Prompting for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote. Sens.* **2023**, *61*, 5532216. [CrossRef]
33. Yang, K.; Sun, H.; Zou, C.; Lu, X. Cross-Attention Spectral–Spatial Network for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote. Sens.* **2021**, *60*, 5518714. [CrossRef]
34. Wu, H.; Shi, C.; Wang, L.; Jin, Z. A Cross-Channel Dense Connection and Multi-Scale Dual Aggregated Attention Network for Hyperspectral Image Classification. *Remote. Sens.* **2023**, *15*, 2367. [CrossRef]
35. Zhang, X.; Sun, G.; Jia, X.; Wu, L.; Zhang, A.; Ren, J.; Fu, H.; Yao, Y. Spectral–Spatial Self-Attention Networks for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote. Sens.* **2021**, *60*, 5512115. [CrossRef]
36. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *arXiv* **2017**, arXiv:1706.03762.
37. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
38. Heo, B.; Yun, S.; Han, D.; Chun, S.; Choe, J.; Oh, S.J. Rethinking Spatial Dimensions of Vision Transformers. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 11916–11925.
39. Graham, B.; El-Nouby, A.; Touvron, H.; Stock, P.; Joulin, A.; Jegou, H.; Douze, M. LeViT: A Vision Transformer in ConvNet's Clothing for Faster Inference. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 12239–12249.
40. 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.
41. Yuan, L.; Chen, Y.; Wang, T.; Yu, W.; Shi, Y.; Jiang, Z.; Tay, F.E.H.; Feng, J.; Yan, S. Tokens-to-Token ViT: Training Vision Transformers from Scratch on ImageNet. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 558–567.
42. Hong, D.; Han, Z.; Yao, J.; Gao, L.; Zhang, B.; Plaza, A.; Chanussot, J. SpectralFormer: Rethinking Hyperspectral Image Classification With Transformers. *IEEE Trans. Geosci. Remote. Sens.* **2021**, *60*, 5518615. [CrossRef]
43. Mei, S.; Song, C.; Ma, M.; Xu, F. Hyperspectral Image Classification Using Group-Aware Hierarchical Transformer. *IEEE Trans. Geosci. Remote. Sens.* **2022**, *60*, 5539014. [CrossRef]
44. Zhong, Z.; Li, Y.; Ma, L.; Li, J.; Zheng, W.-S. Spectral–Spatial Transformer Network for Hyperspectral Image Classification: A Factorized Architecture Search Framework. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5514715. [CrossRef]
45. Song, R.; Feng, Y.; Cheng, W.; Mu, Z.; Wang, X. BS2T: Bottleneck Spatial–Spectral Transformer for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote. Sens.* **2022**, *60*, 5532117. [CrossRef]
46. Sun, L.; Zhao, G.; Zheng, Y.; Wu, Z. Spectral–Spatial Feature Tokenization Transformer for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5522214. [CrossRef]

47. Jiang, M.; Su, Y.; Gao, L.; Plaza, A.; Zhao, X.-L.; Sun, X.; Liu, G. GraphGST: Graph Generative Structure-Aware Transformer for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote. Sens.* **2024**, *62*, 5504016. [CrossRef]
48. Feng, J.; Wang, Q.; Zhang, G.; Jia, X.; Yin, J. CAT: Center Attention Transformer with Stratified Spatial–Spectral Token for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote. Sens.* **2024**, *62*, 5615415. [CrossRef]
49. Wang, N.; Yang, A.; Cui, Z.; Ding, Y.; Xue, Y.; Su, Y. Capsule Attention Network for Hyperspectral Image Classification. *Remote. Sens.* **2024**, *16*, 4001. [CrossRef]
50. Jamali, A.; Roy, S.K.; Hong, D.; Lu, B.; Ghamisi, P. How to Learn More? Exploring Kolmogorov–Arnold Networks for Hyperspectral Image Classification. *Remote Sens.* **2024**, *16*, 4015. [CrossRef]
51. Licciardi, G.; Marpu, P.R.; Chanussot, J.; Benediktsson, J.A. Linear Versus Nonlinear PCA for the Classification of Hyperspectral Data Based on the Extended Morphological Profiles. *IEEE Geosci. Remote Sens. Lett.* **2012**, *9*, 447–451. [CrossRef]
52. He, K.; Zhang, X.; Ren, S.; Sun, J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1026–1034.
53. Debes, C.; Merentitis, A.; Heremans, R.; Hahn, J.; Franfiadakis, N.; van Kasteren, T.; Liao, W.; Bellens, R.; Pižurica, A.; Gautama, S.; et al. Hyperspectral and LiDAR Data Fusion: Outcome of the 2013 GRSS Data Fusion Contest. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 2405–2418. [CrossRef]
54. 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]
55. 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]
56. Wang, Q.; Huang, J.; Meng, Y.; Shen, T. DF2Net: Differential Feature Fusion Network for Hyperspectral Image Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2024**, *17*, 10660–10673. [CrossRef]
57. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.