Dynamic Dependency Convolution and Hybrid Attention Transformer Network for Hyperspectral Image Classification

Hongmeng Lu[1,2], Liejun Wang[1,2,*], Shaochen Jiang[1,2]

***Abstract*—Recently, Convolutional Neural Networks (CNNs) and Transformer have gradually become mainstream methods for hyperspectral image (HSI) classification tasks. However, due to the complex land cover scenes and high-dimensional spectral characteristics of HSI, existing methods are often disturbed by redundant information during the feature learning process, which increases the difficulty of effective feature extraction. To address this challenge, this paper proposes a network based on dynamic dependency convolution and hybrid attention Transformer (DDCHAT). Specifically, the model designs a multi-scale dynamic dependency convolution (MDDC)module to enhance the ability to model local spatial details, while incorporating a hybrid attention Transformer (HAT) module to capture global spectral-spatial dependencies. This enables the collaborative effect of dynamic convolution operators and key discriminative information enhancement strategies. Experimental results on three publicly available datasets demonstrate that the proposed DDCHAT achieves high classification accuracy with relatively low parameters and computational cost.**

***Index Terms*—Hyperspectral image (HSI) classification, convolution neural networks (CNNs), transformer.**

I. Introduction

Hyperspectral images (HSIs) are high-resolution digital images with hundreds of continuous spectral bands, capable of providing detailed spectral and spatial information. Therefore, HSIs have been widely applied in fields such as environmental monitoring [1], agricultural production [2], and urban planning [3]. The goal of HSI classification is to assign an accurate class label to each pixel, enabling effective recognition and analysis of different land cover types.

With the development of deep learning (DL) technologies, HSI classification tasks have increasingly relied on DL methods. Compared to traditional machine learning, DL can automatically extract deeper features. Among these, Convolutional Neural Networks (CNN) and Transformers are the two most widely used models. CNN is inspired by the biological visual system, featuring local connections, weight sharing, and hierarchical feature extraction [4] [5] [6]. For example, Yang et al. [7]designed a dual-branch architecture combining 1D CNN and 2D CNN to extract local spectral features and spatial features. However, CNN's feature extraction method based on fixed convolutional operators lacks adaptability when processing diverse regions [8] [9]and also struggles to effectively capture long-range dependencies [10] [11].

The emergence of Vision Transformer has provided a new research direction for HSI classification. Transformer-based models, with their self-attention mechanism, can effectively model long-range dependencies [12] [13] [14]. For example, Mohamed et al. [15] introduced a factorized self-supervised pretraining method in Transformer, which effectively captures the spectral-spatial information in HSI. However, due to the high-dimensional nature of HSI, Transformer often introduce a large amount of redundant information, which may obscure key discriminative features [16] [17]. Additionally, Transformer tend to overlook local important information within the spectral bands [18] [19].Given the limitations of these two methods, researchers have further proposed classification frameworks that integrate CNN and Transformer [20] [21]. However, these frameworks still fail to achieve optimal classification performance.

To solve the above problems, this paper proposes a network based on dynamic dependency convolution and hybrid attention Transformer (DDCHAT) for HSI classification. Specifically, we design a multi-scale dynamic dependency convolution (MDDC) module to adaptively extract local spectral-spatial features from HSI. Additionally, we design a sparse-dense hybrid attention (SDHA) and channel gated feedforward network (CGFN) to construct a hybrid attention Transformer (HAT), which effectively suppresses redundant information from both the spatial and channel dimensions, ensuring the effective extraction and propagation of features. The main innovative contributions are summarized as follows.

1) We propose a hybrid network named DDCHAT that combines the strengths of CNN and Transformers for HSI classification.
2) We design an MDDC module that adaptively extracts local features from HSI in complex land cover environments through the collaborative effect of dynamic convolution operators and multi-scale branches.
3) We propose a CGFN module that applies a gated weighting mechanism along the channel dimension to effectively highlight the spectral channels that are discriminative for the classification task.
4) We have designed an SDHA module that effectively reduces redundant information in the spatial dimen-

This work was supported by the project "Data Elemental-ization Information System for Unmanned Platforms Orientedto Low-Altitude Inspection" (Grant No. 2025YFF0515604).

[1]College of Computer Science and Technology, Xinjiang University, Urumqi, China.
[2]Xinjiang Multimodal Intelligent Processing and Information Security Engineering Technology Research Center, Urumqi, China.
Correspondence: wljxju@xju.edu.cn

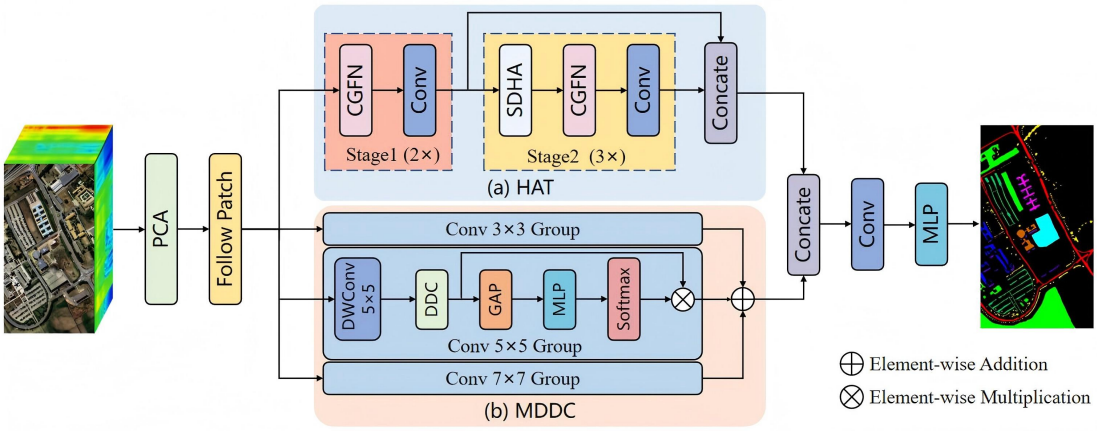


Fig. 1: Overall architecture of DDCHAT. MDDC denotes multi-dimensional dynamic convolutional. HAT denotes Hybrid Attention Transformer,composed of channel gated feedforward network(CGFN) and sparse-dense hybrid attention(SDHA).

sion while ensuring information integrity by employing sparse attention (SSA) and dense attention (DSA).

II. METHODOLOGY

A. Overview

The overall architecture of the proposed DDCHAT is illustrated in Fig. 1. Given an input HSI $I \in \mathbb{R}^{H\times W\times C}$, principal component analysis (PCA) is first applied to reduce the spectral dimensionality. The dimension-reduced image is then partitioned into patches of identical size using the Follow Patch method proposed by Yang et al. [22], resulting in $X \in \mathbb{R}^{P\times P\times C}$. Subsequently, X is fed in parallel into the MDDC module and the HAT module to extract local and global features, respectively. The HAT module consists of two stages. In the first stage, only the CGFN is introduced to perform shallow feature extraction while suppressing channel redundancy. In the second stage, the joint operation of SDHA and CGFN enables further deep feature modeling and reduces spatial redundancy. Finally, the extracted features are concatenated and fused, and then fed into a multilayer perceptron (MLP) to complete the classification prediction.

B. Multi-scale Dynamic Dependency Convolution

In the MDDC, multi-scale depthwise convolution (DWConv) are employed to perform local feature modeling on the input feature X, and dynamic dependency convolution (DDC) is further introduced for each branch convolution output, as illustrated in Fig. 2. Specifically, for the input X, adaptive average pooling (AAP) and global average pooling (GAP) are applied to obtain $X_{AAP} \in \mathbb{R}^{K\times K\times C}$ and $X_{GAP} \in \mathbb{R}^{1\times 1\times C}$, respectively. These two features are then fed into two separate 1×1 convolution layers for channel-wise compression and reconstruction, yielding $T_1 \in \mathbb{R}^{K\times K\times C}$ and $T_2 \in \mathbb{R}^{1\times 1\times 2C}$. Subsequently, T_1 is concatenated with X_{AAP} and reshaped into $T_3 \in \mathbb{R}^{\beta\times K\times K\times C}$, which is multiplied with a learnable parameter M to generate the dynamic dependency convolution kernel W. Meanwhile, T_2 is reshaped and multiplied with another learnable parameter N to produce the corresponding dynamic bias B.The equation is as follows:

$$W = \text{Reshape}(\text{Concat}(T_1, X_{AAP})) \otimes M \tag{1}$$

$$B = \text{Reshape}(T_2) \otimes N \tag{2}$$

Where $\otimes$ denotes the multiplication.Finally, W and B are jointly applied to X to obtain the output feature Y_i corresponding to each branch. GAP and an MLP layer are then applied to Y_i, followed by Softmax normalization to obtain adaptive weights for each branch. Perform element-wise multiplication between the weights and the corresponding branch features, then sum over the channel dimension to obtain output feature X_{MDDC} of the MDDC module.The equation is as follows:

$$X_{\text{MDDC}} = \sum_{i=1}^{S} Y_i \otimes \text{Softmax}(\text{MLP}(\text{GAP}(Y_i))) \tag{3}$$

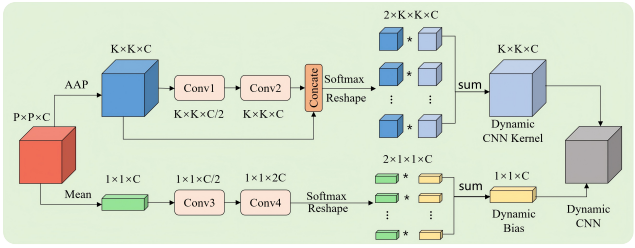


Fig. 2: Architecture of the proposed DDC.

C. Channel Gated Feedforward Network

Since conventional feedforward networks struggle to adaptively adjust the contributions of different spectral channels according to the input, this paper proposes a CGFN module, as shown in Fig. 3. Specifically, partial convolution (PConv) is first applied to the effective regions of the input feature X to produce the feature Z. Then, Z is split into two parts along the channel dimension: $z_1 \in \mathbb{R}^{P\times P\times \frac{C}{2}}$ and $z_2 \in \mathbb{R}^{P\times P\times \frac{C}{2}}$. The feature z_1 is reshaped into $B \times P \times P \times \frac{C}{2}$ and enhanced

using DWConv. The enhanced result is then flattened back to its original shape and multiplied element-wise with z_2, yielding the gated feature $X_1 \in \mathbb{R}^{P\times P\times \frac{C}{2}}$. Finally, X_1 is linearly projected back to C channels and added to the input X via a residual connection, producing the output of the CGFN module, denoted as X_{CGFN}.The equation is as follows:

$$Z = \text{GELU}(W_1 \cdot \text{PConv}(X)),\ [z_1, z_2] = Z \tag{4}$$

$$X_1 = z_2 \otimes \text{Flatten}\left(\text{DWConv}\left(\text{Reshape}(z_1)\right)\right) \tag{5}$$

$$X_{\text{CGFN}} = \text{GELU}(W_2 \cdot X_1) + X \tag{6}$$

Where W_1 and W_2 denote the learnable parameters of the fully connected layers, and $[\,,\,]$ represents the operation of splitting features along the channel dimension. $\text{GELU}(\cdot)$ denotes the GELU activation function.

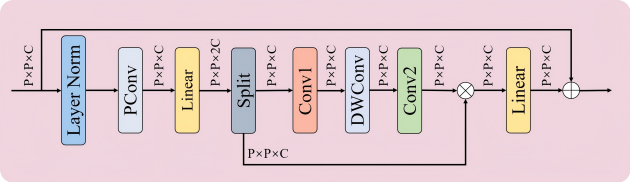


Fig. 3: Architecture of the proposed CGFN.

D. Sparse-dense Hybrid Attention

To address the issue that self-attention mechanisms perform interactive computations across all positions of input features during modelling, resulting in high computational complexity and susceptibility to redundant information, this paper proposes the SDHA module, as illustrated in Figure 4. The SDHA module introduces sparse self-attention (SSA) and dense self-attention (DSA) to form a dual-branch attention architecture. Specifically, SSA employs a squared ReLU activation function to sparsify attention weights, thereby focusing solely on the most relevant Q and corresponding key matrix K. DSA, conversely, retains the standard Softmax activation function to perform global modelling across all Q and K elements, ensuring the complete transmission of critical information.The equation is as follows:

$$A_{\text{SSA}} = \text{ReLU}^2\left(\frac{QK^\top}{\sqrt{C}} + B\right)V \tag{7}$$

$$A_{\text{DSA}} = \text{Softmax}\left(\frac{QK^\top}{\sqrt{C}} + B\right)V \tag{8}$$

To balance information retention and redundancy suppression, the SDHA module combines the outputs of the SSA and DSA branches through a weighted summation process.The equation is as follows:

$$A_{\text{SDHA}} = w_1 \cdot A_{\text{SSA}} + w_2 \cdot A_{\text{DSA}} \tag{9}$$

$$w_n = \frac{e^{a_n}}{\sum_{i=1}^{N} e^{a_i}}, \quad n = \{1, 2\} \tag{10}$$

Where A_{SDHA} denotes the output of the SDHA module, and w_1 and w_2 are the learnable adaptive weights corresponding to the two branches. a_n is a learnable scalar parameter associated with the n-th attention branch, and N denotes the total number of attention branches.

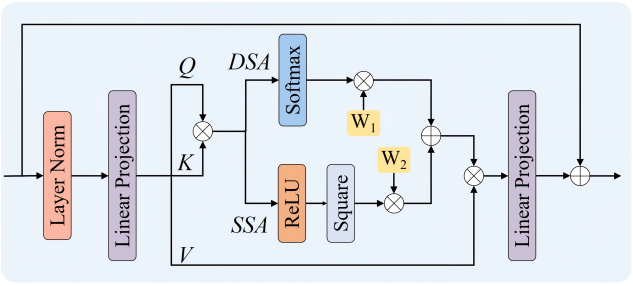


Fig. 4: Architecture of the proposed SDHA.

III. EXPERIMENTS

A. Experimental Setup

To evaluate the performance of DDCHAT, extensive experiments were conducted on three public HSI datasets: Indian Pines (IP), Pavia University (PU), and Houston2013 (HU). For the IP, PU, and HU datasets, 5%, 1%, and 2% of the samples were used for training, respectively, with the remaining samples used for testing. The input feature dimension was uniformly normalized to 90. All experiments were performed on a computing device equipped with an NVIDIA GeForce RTX 3090 GPU. Model parameters were optimized using the Adam optimizer, and the cross-entropy loss function was employed to compute the training error. The initial learning rate was set to 0.0005, the weight decay was 0.05, the batch size was 64, and the model was trained for 200 epochs.We employed overall accuracy (OA), average accuracy (AA), and the Kappa coefficient as quantitative evaluation metrics for the entire network.

B. Analysis of Comparative Experiments

On the three public datasets, we conducted comparative experiments between DDCHAT and several typical methods, including 2DCNN [23], 3DCNN [24], SSFTT [25], FactoFormer [15], CTMixer [26], MorphFormer [27], GTCFN [28],

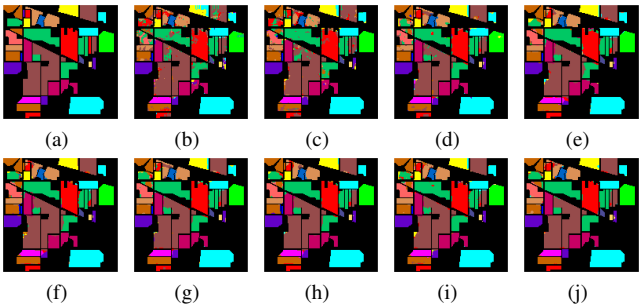


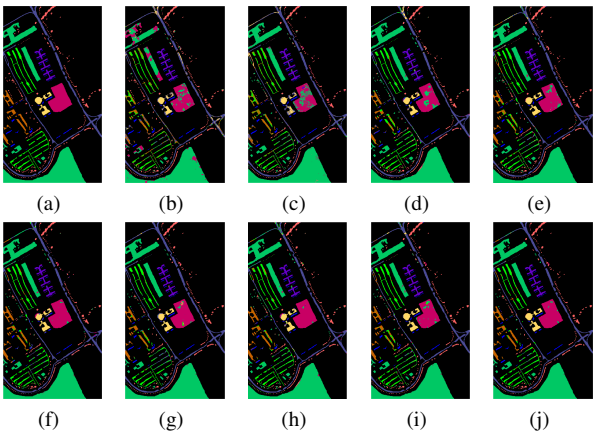
Fig. 5: Classification graph of IP dataset. (a)-(j) denote Ground Truth, 2DCNN, 3DCNN, SSFTT, FactoFormer, CTMixer, MorphFormer, GTCFN, SAGT and DDCHAT.

TABLE I: COMPARATIVE EXPERIMENTAL RESULTS ON THE INDIAN PINES DATASET

Class	Color	2DCNN	3DCNN	SSFTT	FactoFormer	CTMixer	MorphFormer	GTCFN	SAGT	DDCHAF
1		95.12±2.65	96.95±3.17	96.94±5.15	93.78±1.22	93.72±1.21	**99.72±0.83**	98.49±0.90	97.48±2.00	97.22±1.84
2		68.23±2.14	65.98±1.43	87.59±1.43	63.36±2.15	66.32±1.67	93.94±1.72	70.69±6.88	69.93±1.24	**95.9±0.99**
3		84.95±3.03	80.50±1.92	86.54±2.90	90.21±4.75	85.93±4.31	91.16±2.69	93.91±0.55	92.42±3.29	**94.67±2.21**
4		89.17±1.56	81.99±5.63	85.77±5.42	93.12±3.34	91.40±2.88	93.83±3.27	95.66±3.33	**96.70±5.93**	94.69±1.85
5		88.20±5.64	75.39±6.32	90.76±1.25	87.47±2.07	90.04±1.73	92.97±1.98	93.10±5.92	94.02±1.27	**94.34±3.05**
6		92.00±3.03	86.42±9.82	97.82±1.58	88.26±8.96	91.4±3.18	99.11±0.51	92.72±2.25	92.25±4.45	**100.00±0.00**
7		88.36±9.70	84.57±4.13	100.00±0.00	78.56±4.37	85.94±7.14	100.00±0.00	94.56±2.01	85.76±8.79	**100.00±0.00**
8		95.28±0.98	83.33±9.30	99.45±0.73	99.69±0.63	96.86±2.26	99.08±0.82	95.48±4.26	98.12±1.56	**99.86±0.16**
9		96.04±3.13	88.29±6.40	100.00±0.00	91.40±3.35	93.46±3.44	**100.00±0.00**	90.79±3.93	96.77±3.51	93.68±0.18
10		88.24±3.55	77.27±18.06	85.73±1.56	74.28±4.16	90.93±4.38	90.79±1.93	87.50±2.67	85.44±10.67	**95.98±0.64**
11		94.06±2.72	98.59±1.03	94.32±1.07	90.08±3.68	97.70±1.82	97.10±0.68	99.28±0.34	99.49±1.71	**99.62±0.14**
12		49.66±2.90	51.90±11.78	81.93±2.99	65.46±12.77	73.75±12.71	82.63±2.38	90.24±7.01	87.11±8.27	**92.88±1.16**
13		82.03±5.19	76.03±10.70	98.71±0.77	87.35±2.81	87.72±4.79	**98.56±1.03**	93.23±2.59	90.40±4.47	88.47±0.35
14		90.75±4.95	90.80±4.19	97.79±0.73	86.68±1.18	95.73±2.67	98.64±0.73	97.82±1.44	95.48±3.26	**98.98±0.48**
15		85.08±6.62	73.31±5.25	89.66±1.55	80.31±3.11	86.42±3.48	96.76±3.45	90.98±8.20	90.10±7.31	**98.91±0.83**
16		92.49±2.07	93.75±5.14	98.97±1.23	98.19±1.42	97.58±4.84	98.86±1.45	98.45±1.78	90.71±13.16	**99.30±0.28**
OA(%)		84.70±2.04	88.21±1.81	91.70±0.44	92.84±0.62	93.65±2.53	94.97±0.37	95.76±0.82	94.17±1.83	**96.80±0.47**
AA(%)		85.62±6.07	88.61±3.63	93.21±0.68	86.59±2.57	89.40±2.76	95.82±0.42	90.75±2.72	90.53±4.73	**97.01±0.39**
$\kappa \times 100$		85.10±2.72	87.16±3.22	90.51±0.51	90.29±0.55	91.67±0.41	94.26±0.42	95.49±0.72	93.65±1.92	**96.34±0.54**

TABLE II: COMPARATIVE EXPERIMENTAL RESULTS ON THE PAVIA UNIVERSITY DATASET

Class	Color	2DCNN	3DCNN	SSFTT	FactoFormer	CTMixer	MorphFormer	GTCFN	SAGT	DDCHAF
1		78.84±3.12	84.77±6.12	90.28±2.62	97.86±0.91	97.43±1.31	92.90±1.59	94.35±5.08	**98.52±4.35**	90.60±0.29
2		91.10±3.98	93.85±5.62	98.19±0.70	97.11±2.56	93.28±1.96	98.80±0.75	96.86±5.70	94.84±2.47	**99.98±0.02**
3		72.31±8.82	61.76±8.25	94.29±3.90	74.61±0.32	84.48±4.63	86.90±2.14	71.62±1.33	92.70±7.18	**95.67±1.46**
4		74.34±5.69	90.00±3.39	85.40±2.84	97.46±0.13	89.21±1.07	84.17±4.89	**95.74±8.81**	84.24±4.21	91.38±1.96
5		93.35±2.26	90.92±3.43	82.94±2.34	98.97±1.83	97.93±3.06	97.47±0.70	94.84±4.32	98.33±2.70	**99.17±0.26**
6		74.68±15.41	57.72±5.07	85.78±4.45	88.93±0.14	88.42±7.67	88.06±0.91	88.62±8.96	92.97±6.77	**99.77±0.76**
7		60.23±11.29	96.94±5.62	85.29±2.88	82.27±3.90	82.81±1.59	90.89±3.77	82.05±5.19	98.18±0.89	**99.49±0.69**
8		64.82±4.36	92.43±1.82	79.68±4.75	97.26±1.28	94.93±1.56	88.50±3.43	94.93±2.11	97.47±7.56	**98.72±6.21**
9		90.56±2.40	93.85±2.66	76.88±4.47	94.93±1.92	96.95±2.59	97.33±2.64	93.85±2.88	93.86±2.92	**98.11±3.20**
OA(%)		81.98±1.92	87.86±3.42	91.67±0.55	92.07±1.17	93.32±1.99	94.96±0.51	93.98±2.69	96.40±3.27	**97.89±0.35**
AA(%)		77.81±2.17	87.64±3.17	87.34±0.88	89.34±1.07	94.45±1.89	92.78±0.69	91.14±3.18	95.05±4.45	**96.53±0.46**
$\kappa \times 100$		76.09±2.57	83.63±4.37	88.92±0.74	89.12±1.52	91.01±2.21	93.30±0.69	91.88±3.28	95.13±2.57	**96.92±0.69**



(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Fig. 6: Classification graph of PU dataset. (a)-(j) denote Ground Truth, 2DCNN, 3DCNN, SSFTT, FactoFormer, CTMixer, MorphFormer, GTCFN, SAGT and DDCHAT.

and SAGT [29]. All experiments were repeated ten times, and the averaged results are reported as the final performance. The quantitative comparison results are presented in Tables I–III, and the classification maps are shown in Figs. 5–7.

In the IP dataset, as shown in Table 1, under the condition of using only 5% training samples, DDCHAT achieves an OA value of 96.80 ± 0.47%, surpassing the second-best method MHCFormer by 1.04%. In contrast, 2DCNN and 3DCNN methods, constrained by their local receptive fields, exhibit relatively lower classification performance. Although models such as SAGT and MorphFormer, which fuse CNNs with Transformers, can jointly extract airborne spectral features, they still exhibit certain limitations in feature extraction within complex terrain regions. As evidenced by the visualisation results in Figure 5, DDCHAT demonstrates greater consistency between predicted and ground truth labels for feature recognition in the brown and purple regions at the bottom of the image. Similar trends emerge across the other two datasets, confirming DDCHAT's superiority through both quantitative metrics and visual analysis.

C. Ablation Experiment Setup and Model Analysis

Ablation experiments were conducted on the IP dataset in this paper, with results shown in Table IV. Specifically, the introduction of any one of the MDDC, CGFN, or SDHA modules significantly improves the model's performance; however,

TABLE III: COMPARATIVE EXPERIMENTAL RESULTS ON THE HOUSTON2013 DATASET

Class	Color	2DCNN	3DCNN	SSFTT	FactoFormer	CTMixer	MorphFormer	GTCFN	SAGT	DDCHAF
1		91.84±2.94	89.74±1.95	93.04±1.80	88.73±3.95	96.30±0.43	94.70±1.29	90.84±2.45	96.22±0.13	**98.65±0.46**
2		69.13±8.16	96.81±0.65	96.30±2.15	99.24±0.62	93.62±1.58	**99.83±0.53**	97.32±1.80	97.82±0.76	97.65±0.75
3		95.17±1.26	91.40±1.52	96.38±1.26	96.83±0.95	95.02±0.60	**100.00±0.00**	99.55±0.28	96.68±1.14	99.84±0.08
4		87.31±4.64	90.86±2.33	96.79±1.95	**99.92±0.08**	97.80±0.77	98.56±0.49	97.29±1.60	98.98±0.09	95.14±3.91
5		94.75±3.52	96.78±1.90	99.32±0.17	100.00±0.00	99.75±0.25	99.83±0.06	100.00±0.00	99.58±0.20	**100.00±0.00**
6		52.43±20.15	74.43±5.42	86.41±5.49	87.70±3.82	93.53±1.27	94.82±2.96	90.94±3.13	95.15±1.34	**98.38±0.67**
7		56.85±16.13	74.52±4.92	84.07±5.74	96.68±1.74	95.86±0.85	93.94±1.37	89.54±9.20	88.05±4.52	**95.93±1.89**
8		61.08±11.43	72.25±3.19	87.39±8.22	90.44±2.92	96.11±2.93	90.36±2.10	87.90±4.62	90.78±2.30	**96.44±1.31**
9		58.66±7.48	74.12±4.36	86.05±6.45	57.90±20.47	92.61±0.36	92.94±2.85	**98.32±0.79**	87.98±5.99	94.36±2.19
10		69.30±14.23	79.50±9.09	82.25±10.64	70.58±15.37	98.46±0.14	95.11±0.66	89.28±9.18	92.45±1.93	**98.96±1.21**
11		68.99±6.92	69.85±11.55	92.25±1.38	97.36±0.61	94.97±2.20	95.83±1.62	95.66±0.97	92.84±2.46	**98.58±0.26**
12		79.27±21.37	70.73±9.19	83.02±5.12	80.55±5.27	96.93±0.13	91.55±0.96	84.30±4.88	93.60±1.50	**97.69±0.97**
13		48.65±10.29	49.10±27.11	78.25±14.57	97.98±0.71	93.27±3.60	83.18±8.11	95.96±1.61	88.79±6.00	**98.52±0.30**
14		81.57±5.47	95.82±0.82	99.51±0.16	99.75±0.10	99.26±0.13	98.53±0.73	99.51±0.04	**100.00±0.00**	95.73±1.34
15		82.93±6.98	87.72±8.16	97.45±3.28	99.84±0.09	97.93±1.76	100.00±0.00	99.71±0.22	96.65±1.81	**100.00±0.00**
OA(%)		74.07±5.28	81.50±2.19	90.49±4.25	89.71±3.36	96.19±0.65	96.40±0.52	93.89±2.61	94.13±2.34	**97.47±2.00**
AA(%)		73.20±3.67	80.91±1.98	90.56±2.53	90.90±1.95	96.10±0.91	97.78±1.31	94.43±2.47	94.37±2.05	**97.64±1.80**
$k \times 100$		71.99±2.21	80.00±2.44	89.72±3.16	88.87±1.33	95.88±0.43	96.00±0.92	93.39±1.60	93.65±2.61	**97.27±2.64**

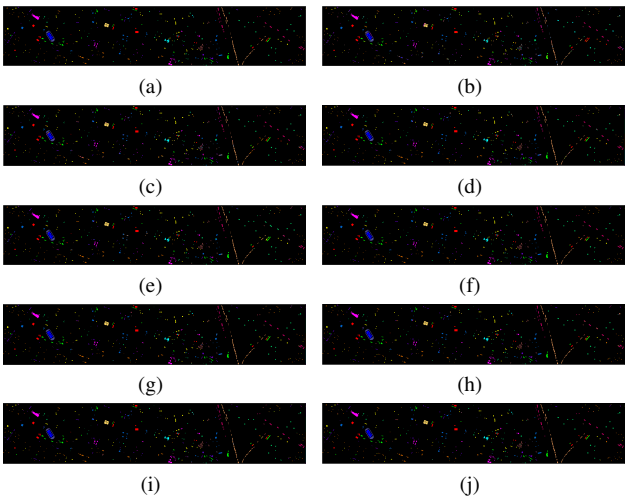


Fig. 7: Classification graph of HU dataset. (a)-(j) denote Ground Truth, 2DCNN, 3DCNN, SSFTT, FactoFormer, CTMixer, MorphFormer, GTCFN, SAGT and DDCHAT.

the performance is still lower than that of any configuration combining two modules. The complete model further improves the OA metric compared to any two-module combination, indicating that the three innovative modules exhibit good complementarity and synergistic benefits.

D. Discussion of the Computational Costs

Table V compares the complexity of the proposed method with eight mainstream models on the IP dataset, from the perspectives of model parameter count and floating point operations (FLOPs). The results show that, although the FLOPs of DDCHAT are slightly higher than those of SAGT, it achieves a 2.63% improvement in the OA metric, while the model parameter count is significantly lower than that of SAGT. This demonstrates that the proposed method achieves more competitive classification performance through a more efficient structural design, while keeping the computational cost manageable.

TABLE IV: ABLATION STUDY RESULTS ON IP DATASET

Module	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
MDDC	×	✓	×	×	✓	✓	×	✓
CGFN	×	×	✓	×	✓	×	✓	✓
SDHA	×	×	×	✓	×	✓	✓	✓
OA(%)	80.72	92.74	84.24	89.37	93.98	95.30	90.96	**96.80**
AA(%)	86.13	91.05	88.13	92.53	94.27	95.83	89.64	**97.01**
$k \times 100$	79.39	91.52	82.27	87.91	93.17	93.41	91.25	**96.34**

TABLE V: COMPUTATIONAL COST ON THE IP DATASET

Methods	Parameters (K)	FLOPs (M)
2DCNN	1696.47	67.56
3DCNN	204.68	185.63
SSFTT	937.31	52.16
FactoFormer	243.82	40.59
CTMixer	841.59	47.62
MorphFormer	227.64	37.21
GTCFN	3520.48	324.89
SAGT	172.65	**13.46**
DDCHAT	**137.32**	14.36

E. Parameters Analyzed

With other parameters held constant, this paper conducted comparative experiments using six distinct patch sizes. The overall accuracy (OA) values across three datasets are presented in Table VI. The experimental results indicate that for

TABLE VI: IMPACT OF DIFFERENT PATCH SIZES ON OA(%)

Patch Size	Indian Pines	Pavia University	Houston2013
5×5	89.05	88.76	90.92
7×7	92.97	91.76	92.45
9×9	94.31	92.97	94.61
11×11	96.08	95.27	95.96
13×13	**96.80**	96.25	**97.47**
15×15	96.24	**97.89**	96.74

the IP and HU datasets, classification performance is optimal when the patch size is set to 13×13. Conversely, on the PU dataset, the OA value gradually increases, reaching its optimum when the patch size is 15×15. This occurs because the PU dataset possesses a larger sample size; consequently, excessively small patch sizes lead to fragmented feature representation and information loss.

IV. CONCLUSION

To address the challenges of feature extraction and information redundancy arising from complex terrain features and high dimensionality in HSI, this paper innovatively proposes the DDCHAT model architecture. This framework employs a synergistic strategy combining dynamic convolution operators with enhanced key discriminative information to efficiently extract spectral-spatial features while effectively reducing computational complexity. Notably, the MDDC module achieves thorough mining and precise extraction of local HSI features through the organic integration of multi-scale deep convolution, dynamic dependency convolution, and adaptive weighting mechanisms. The HAT module efficiently captures global HSI features by leveraging the synergistic effects of CGFN and SDHA. In future work, we shall continue exploring the strengths of CNNs and Transformer to develop more efficient and lightweight networks.

REFERENCES

[1] Audebert N, Le Saux B, Lefèvre S. "Deep learning for classification of hyperspectral data: A comparative review," IEEE geoscience and remote sensing magazine, 2019, 7(2): 159-173.

[2] Paoletti et al. "Deep learning classifiers for hyperspectral imaging: A review," ISPRS Journal of Photogrammetry and Remote Sensing 158 (2019): 279-317.

[3] Benediktsson J A, Palmason J A, Sveinsson J R. "Classification of hyperspectral data from urban areas based on extended morphological profiles," IEEE Transactions on Geoscience and Remote Sensing, 2005, 43(3): 480-491.

[4] Hu W, Huang Y, Wei L, et al. "Deep convolutional neural networks for hyperspectral image classification," Journal of Sensors, 2015, 2015(1): 258619.

[5] Liang L, Zhang Y, Zhang S, et al. "Fast hyperspectral image classification combining transformers and SimAM-based CNNs," IEEE Transactions on Geoscience and Remote Sensing, 2023, 61: 1-19.

[6] Chen Y, Jiang H, Li C, et al. "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," IEEE transactions on geoscience and remote sensing, 2016, 54(10): 6232-6251.

[7] Yang J, Zhao Y Q, Chan J C W. "Learning and transferring deep joint spectral–spatial features for hyperspectral classification," IEEE Transactions on Geoscience and Remote Sensing, 2017, 55(8): 4729-4742.

[8] D. Hong et al., "SpectralGPT: Spectral remote sensing foundation model," IEEE Trans. Pattern Anal. Mach. Intell, vol. 46, no. 8, pp. 5227–5244, Aug. 2024.

[9] Y. Gu, Q. Wang, H. Wang, D. You, and Y. Zhang, "Multiple kernel learning via low-rank nonnegative matrix factorization for classification of hyperspectral imagery," IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens, vol. 8, no. 6, pp. 2739–2751, Jun. 2015.

[10] M. Ahmad and M. Mazzara, "SCSNet: Sharpened cosine similarity-based neural network for hyperspectral image classification," IEEE Geosci. Remote Sens. Lett, vol. 21, pp. 1–4, 2024.

[11] N. Audebert, B. Le Saux, and S. Lefevre, "Deep learning for classification of hyperspectral data: A comparative review,"IEEE Geosci. Remote Sens. Mag, vol. 7, no. 2, pp. 159–173, Jun. 2019.

[12] Zhang M, Zhang C, Zhang Q, et al. "ESSAformer: Efficient transformer for hyperspectral image super-resolution," Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023: 23073-23084.

[13] Mei S, Song C, Ma M, et al. "Hyperspectral image classification using group-awarehierarchical transformer," IEEE Transactions on Geoscience and Remote Sensing, 2022,60: 1-14.

[14] Zhao Z, Xu X, Li S, et al. "Hyperspectral image classification using groupwise separable convolutional vision transformer network," IEEE Transactions on Geoscience and Remote Sensing, 2024, 62: 1-17.

[15] Mohamed S, Haghighat M, Fernando T, et al."FactoFormer: Factorized hyperspectral transformers with self-supervised pretraining," IEEE Transactions on Geoscience and Remote Sensing, 2023, 62: 1-14.

[16] S. Mei, C. Song, M. Ma, and F. Xu, "Hyperspectral image classification using group-aware hierarchical transformer," IEEE Trans. Geosci. Remote Sens, vol. 60, pp. 1–14, Sep. 2022.

[17] T. Yao, Y. Pan, Y. Li, C.-W. Ngo, and T. Mei, "Wave-ViT: Unifying wavelet and transformers for visual representation learning," in Proc. Eur. Conf. Comput. Vis, Cham, Switzerland: Springer, 2022, pp. 328–345.

[18] Z. Xue, Q. Xu, and M. Zhang, "Local transformer with spatial partition restore for hyperspectral image classification," IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens, vol. 15, pp. 4307–4325, 2022.

[19] C. Ma et al., "Light self-Gaussian-attention vision transformer for hyperspectral image classification," IEEE Trans. Instrum. Meas, vol. 72, pp. 1–12, 2023.

[20] Y. Zhou, X. Huang, X. Yang, et al., "DCTN: Dual-branch convolutional transformer network with efficient interactive self-attention for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–16, 2024.

[21] Z. Qiu, J. Xu, J. Peng, and W. Sun, "Cross-channel dynamic spatial-spectral fusion transformer for hyperspectral image classification," IEEE Trans. Geosci. Remote Sens, vol. 61, pp. 1–12, 2023.

[22] Yang A et al., "GTFN: GCN and transformer fusion network with spatial-spectral features for hyperspectral image classification," IEEE Trans. Geosci. Remote Sens, vol. 61, pp. 1–15, 2023.

[23] Lee H, Kwon H. "Contextual deep CNN based hyperspectral classification," 2016 IEEE international geoscience and remote sensing symposium (IGARSS). IEEE, 2016: 3322-3325.

[24] Hamida A B, Benoit A, Lambert P, et al. "3-D deep learning approach for remote sensing image classification," IEEE Transactions on geoscience and remote sensing, 2018, 56(8): 4420-4434.

[25] L. Sun, G. Zhao, Y. Zheng, and Z. Wu, "Spectral–spatial feature tokenization transformer for hyperspectral image classification," IEEE Trans. Geosci. Remote Sens, vol. 60, pp. 1–14, 2022.

[26] Zhang J, Meng Z, Zhao F, et al. "Convolution transformer mixer for hyperspectral image classification," IEEE Geoscience and Remote Sensing Letters, 2022, 19: 1-5.

[27] S. K. Roy et al., "Spectral–spatial morphological attention transformer for hyperspectral image classification," IEEE Trans. Geosci. Remote Sens, vol. 61, pp. 1–15, 2023.

[28] Zhao X, Ma J, Wang L, et al. "GTCFN: A Graph-Based Transformer and Convolution Fusion Network for Hyperspectral Image Classification," EEE Transactions on Geoscience and Remote Sensing, 2025, 63: 1-19.

[29] Zhang S, Jiang S, Yin W, et al. "SAGT: Structure-Adaptive Graph Transformer for Hyperspectral Image Classification," IEEE Transactions on Geoscience and Remote Sensing, 2025, 63: 1-16.