CSTFNet: A CNN and Dual Swin-Transformer Fusion Network for Remote Sensing Hyperspectral Data Fusion and Classification of Coastal Areas

Dekai Li, Harold Neira-Molina Mengxing Huang, Syam M.S. , Yu Zhang, Zhang Junfeng, Uzair Aslam Bhatti, *Senior Member, IEEE,* Muhammad Asif, Nadia M. Sarhan, E. M. Awwad

[1]***Abstract*— Hyperspectral imaging (HSI) can capture a large amount of spectral information at various wavelengths, enabling detailed material classification and identification, making it a key tool in remote sensing, particularly for coastal area monitoring. In recent years, the CNN framework and transformer models have demonstrated strong performance in HSI classification, especially in applications requiring precise change detection and analysis. However, due to the high dimensionality of HSI data and the complexity of spectral-spatial feature extraction, achieving accurate results in coastal areas remains challenging. This paper introduces a new hybrid model, CSTFNet, which combines an improved CNN module and dual-layer Swin Transformer (DLST) to tackle these challenges. CSTFNet integrates spectral and spatial processing capabilities, significantly reducing computational complexity while maintaining high classification accuracy. The improved CNN module employs one-dimensional convolutions to handle high-dimensional data, while the DLST module uses window-based multi-head attention to capture both local and global dependencies. Experiments conducted on four standard HSI datasets (Houston-2013, Samson, KSC, and Botswana) demonstrate that CSTFNet outperforms traditional and state-of-the-art algorithms, achieving overall classification accuracy exceeding 99%. In particular, on the Houston-2013 dataset, the results for OA and AA are 1.00 and the kappa coefficient is 0. 976.The results highlight the robustness and efficiency of the proposed model in coastal area applications, where accurate and reliable spectral-spatial classification is crucial for monitoring and environmental management.**

***Index Terms* –Hyperspectral image, CNN, Swin Transformer, Remote sensing**

This work was supported by National Science Foundation of China (NSFC) grant 62463006. This work was supported in part by the King Saud University, Riyadh, Saudi Arabia under the Researchers Support Program No. (RSP2025R1052) grant. (Corresponding author: Uzair Aslam Bhatti. uzair@hainanu.edu.cn)

Dekai Li, Uzair Aslam Bhatti, Zhang Junfeng and Mengxing Huang is with School of Information and Communication Engineering, Hainan University, China. (lidekai@hainanu.edu.cn, uzair@hainanu.edu.cn, jfzhang@hainanu.edu.cn and huangmx09@163.com) .

Harold Neira-Molina is with Universidad de la Costa, Department of computer science and electronics hneira@cuc.edu.co Colombia-Barranquilla

Syam M.S. is with 1. School of Artificial Intelligence, Jingchu University of Technology, Jingmen 448000, China., Jingmen Cryptometry Application Technology Research Center, Jingmen 448000, China and Internet of Intelligences Application Innovation Research Center, Jingchu University of Technology, Jingmen 448000, China. syamms@jcut.edu.cn

Yu Zhang is with School of Computer Science and Technology, Hainan University, Haikou, 570228, Hainan, China. (yuzhang2015@hainanu.edu.cn)

Muhammad Asif is with Hunan University of Science and Engineering masif@huse.edu.cn

Emad Mahrous Awwad is with Department of Electrical Engineering, College of Engineering, King Saud University, P.O. Box 800, Riyadh 11421, Saudi Arabia 442106835@student.ksu.edu.sa

Nadia Sarhan Department of Quantitative Analysis, College of Business Administration, King Saud University, Riyadh, Saudi Arabia nsarhan@ksu.edu.sa

I. Introduction

HYPERSPECTRAL imaging (HSI) is a versatile and effective remote sensing modality that records spectral information across multiple, very fine spectral bands beyond what could be distinguished under normal photographic methods. HSI plays a pivotal role in various climate and agriculture fields, providing precise spectral data that can detect subtle differences in earth observation, plant health, nutrient deficiencies, and disease presence, which are often invisible without monitoring. The technology has real-time monitoring and decision-making capabilities, thereby facilitating improvements in resource management and sustainability in modern farming practices.

HSI comprises hundreds of contiguous electromagnetic spectral bands, which are capable of capturing a wealth of information about the Earth's surface [1]. Each pixel in HSI contains detailed spectral features that provide rich information about the material composition of the Earth's surface, enabling the accurate identification and classification of materials, objects and land cover types. Thus, HSI technology has been extensively employed in a multitude of disciplines, including agriculture [2], environmental monitoring [3], mineral exploration [4], and earth sciences [5]. In order to fully exploit the potential of hyperspectral data (HSD), researchers have investigated a plethora of data processing methodologies, including data compression [6], spectral unmixing [7], target detection [8], data reconstruction and recovery [9], and classification [10]. Among the numerous techniques available, classification plays a pivotal role in data interpretation and has garnered significant attention from researchers.

The past decades have seen tremendous breakthroughs in hyperspectral image classification (HSIC) [11,12]. Among them, traditional HSIC methods usually contain feature selection or feature extraction, and then the processed features are fed into a classifier. Commonly available classifiers are support vector machine (SVM) [13], k-nearest neighbor (K-NN) [14], random forest [15] and logistic regression [16]. However, traditional algorithms rely on artificially crafted features and cannot fully utilize the inherent relationships in HSD, which may destroy the original spatial-spectral structure of the image and make it difficult to access the complex information in HSI, leading to unsatisfactory classification results [17]. In addition, traditional algorithms are often difficult to apply to different data, so choosing an appropriate feature extraction model has been a challenge [18,19].

Recently, the rapid development of deep learning has provided effective solutions for HSIC, including Capsule Networks (CapsNet) [20], Generative Adversarial Networks (GANs) [21], Graph Convolutional Networks (GCNs) [22], and Attention-Based Models [23] which mainly focused in improving the classification accuracy. Among them, convolutional neural networks (CNNs) have shown considerable promise in extracting spectral and spatial features [24-26]. CNN-based methods can usually be categorized into 1D, 2D and 3D CNNs, each capable of capturing different aspects of HSI. A 1D-CNN traverses the HSI using a 1D convolutional kernel to extract deep spectral features associated with each pixel [27]. 2D-CNN uses a 2D convolutional kernel to capture spatial information in different spectral bands [28]. 3D-CNN utilizes a 3D convolutional kernel to process both spatial and spectral features [29].

Feature based processing methods have been added in exisiting research to further improve the classification accuracy and reduce the classification time such as Hu et al. [30] applied 1D-CNNs to HSI classification, successfully extracting deep spectral features but failing to capture spatial information. Roy et al. [31] proposed a hybrid spectral CNN (HybridSN) for HSIC, which combines a spectral-spatial 3D-CNN with a spatial 2D-CNN. This architecture enables joint representation of spatial-spectral features from multiple spectral bands, effectively reducing model complexity while delivering strong performance across four public datasets. Gong et al. [32] developed a lightweight multi-scale squeeze-excitation pyramid pooling network that employs a multi-scale 3D CNN module alongside a pyramid pooling and squeeze-excitation module to enhance hierarchical spectral-spatial features, achieving high accuracy on benchmark datasets like Indian Pines, Salinas, and Pavia University. Despite many advancements in HSI classification it still faces two primary challenges:

1. The high dimensionality of HSD introduces significant computational complexity, increasing processing time.

2. HSI contains both spectral information and spatial context, making it challenging to achieve an effective and balanced fusion of these two aspects for accurate classification.

To address these limitations, Hinton et al. [33] introduced Capsule Networks (CapsNet), which use capsules (groups of neurons) to describe the pose and existence probability of entities. Unlike scalar neurons in CNNs, capsules contain richer information, improving the network's ability to capture spatial and spectral features while reducing computational complexity. Similarly, Zhang et al. [34] proposed 1D-ConvCapsNet to extract spectral-spatial features, thus reducing overfitting and computational cost. Not only that, in order to solve the sequential problem of spectral data more effectively, Transformer was also applied to HSI classification, and Hong et al. [35] proposed a cross-layer hopping model for adaptively fusing the information of each layer. Graph Convolutional Networks (GCNs) have also been introduced to model non-local dependencies by representing HSDs as graphs. Mou et al. [36] proposed a non-local GCN for semi-supervised learning of HSI classification, achieving competitive results and high-quality classification graphs. In addition, Swin Transformers (ST), originally developed for visual tasks, were also used for HSI. Huang et al. [37] introduced a 3D version of ST to take advantage of the spatial and spectral properties of HSI. This model addresses the limitations of traditional CNNs by employing multi-scale semantic representations and achieves excellent performance in HSI classification by reducing complexity. Ayas et al. [54] proposed a new spectral-swin transformer model for HSI classification. The modified model can process spatial and spectral features simultaneously and achieve good results. Long et al. [55] used ST to extract global and local spatial features and learned spectral sequence information from adjacent bands of HSI, achieving good classification results.

However, CNN network also faces challenges of not being able to capture long range dependencies as well as the global environment owing to the local features it employs. While ST has the ability of capturing the hierarchical features, but they face challenge in fully learning the spatial dependencies of the pixels. On the other hand, GCNs are good at modelling the spatial relations however they are restricted by insufficient capability of processing spectrally high dimensional feature maps and therefore the interaction between the spectral and spatial domains can hardly be modelled sufficiently by merely using GCNs. In summary, the limitations of HSI classification mainly include the following aspects:

1. Traditional machine learning methods, such as support vector machines (SVM) and k-nearest neighbors (KNN), are often limited by their inability to capture the complex spatial-spectral relationships inherent in HSI data.

2. Deep learning models, especially convolutional neural networks (CNNs), often have difficulties in capturing long-range dependencies and are computationally expensive when processing high-dimensional HSI data.

3. Transformer model approaches in HSI classification face challenges such as high computational requirements and limited integration with spectral information.

To solve the above problems, we develop a hybrid model which consists of improved CNN and dual Swin-Transformer based Fusion network (CSTFNet) which offers solutions to the problem of dealing with high dimensional data with enhanced feature fusion and strong classification capability. CSTFNet also focuses on the problem of effectively integrating spectral and spatial information in HSI by combining the CNN network's spectral processing with SwinTransformer's ability to capture global context. Our customized CNN module involved multiple convolutions and while transformers modules include Dual Layer-Swin Transformer (DLST) blocks, which can effectively extract spectral and spatial information, and apply the model to HSI. The main contributions of this paper are as follows:

1. In order to address the dimensionality and spectral noise issue associated with HSD, we use multilayer CNN module with one dimensional GlobalAveragePooling after convolution layers. It is an efficient step of percipient dimensionality reduction by summing up spectral feature vectors of the image across all the bands. This method helps the model to handle high-dimensional HSD with low computational cost and reduce the risk of being trapped in local optima or overfitting on spectral noise.

2. To model both local and global spatial dependencies within the HSI, a window-based multi-head attention (WMHA) module within the DLST blocks is applied. This approach allows for accumulation of detail in specific localized regions and progressive merging of these regions into deeper layers allowing for both detailed capturing and overall context preservation.

The mechanism makes the computation to be faster by first reducing attention on that region before widening the span of focus.

3. The method we use here GELU (Gaussian Error Linear Unit) in the MLP block of the DLST in place of ReLU because it provides smoother non-linearity. This choice allows for better gradients flowing during multi classing and specially with HSD. Moreover, the integration dropout regularization contributes to enhancing the general and stable model by reducing model over-fit problems, particularly when training sample sizes are relatively small.

The rest of this paper is organized as follows. Section 2 introduces the basic principles of the proposed model, Section 3 presents the experimental results, and Section 4 concludes.

II. Proposed Method

In this section, we introduce the proposed CSTFNet model for HSI classification, which integrates the CNN and Swin-Transformer modules. The detailed architecture is illustrated in Figure 1.

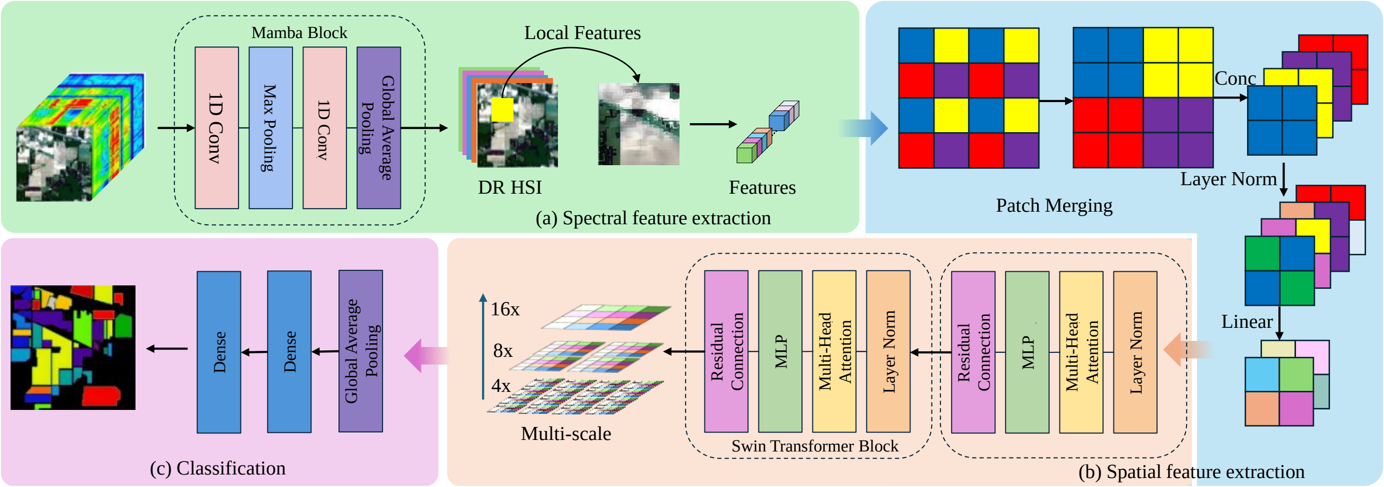


Figure 1. CSTFNet model based on CNN and DLST

A. CNN

A convolutional neural network captures spatial features through a local receptive field (LRF), reduces the number of parameters by utilising weight sharing, and builds a powerful feature extraction capability by combining a nonlinear activation function and a downsampling operation. In practice, by stacking multiple convolutional layers, pooling layers and fully connected layers, complex tasks such as image recognition, semantic segmentation and natural language understanding can be modelled. The basic implementation flow of a convolutional neural network is illustrated in Figure 2.

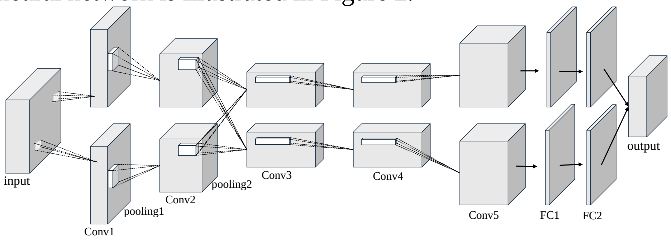


Fig. 2 Basic convolution flow

The convolutional layer represents the fundamental component of a convolutional neural network (CNN), responsible for the extraction of features from the input data through the application of a convolutional kernel. Let us consider an input feature map, X, and a convolution kernel, K. The output of the convolution is calculated according to the following equation:

$$Y(i,j)=\sum_{m=0}^{k_h-1}\sum_{n=0}^{k_w-1}\sum_{c=0}^{C-1}K(m,n,c)\cdot X(i+m,j+n,c)+b \quad (1)$$

where b is the bias and Y is the output feature.

Activation functions facilitate the introduction of non-linear mappings, thereby enabling the network to discern and process complex features. The most commonly utilized activation functions are ReLU, Sigmoid and Tanh. The pooling layer is employed for down sampling, which entails a reduction in the resolution of the feature map and a concomitant reduction in the amount of computation, while ensuring the preservation of the principal features. To illustrate, the maximum pooling is calculated in accordance with the formula presented in Equation 2.

$$Y(i,j)=\max_{m=0}^{p_h-1}\max_{n=0}^{p_w-1}X(i+m,j+n) \quad (2)$$

In this context, " p_h " and " p_w " represent the height and width of the pooling window, respectively. The fully connected layer performs the function of spreading the multidimensional features into one-dimensional vectors and of carrying out feature fusion by means of a weight matrix. The output can be expressed as follows.

$$y=W\cdot x+b \quad (3)$$

where W is the weight matrix, x is the input vector and b is the bias.

B. Dual Layer-Swin Transformer

To process the feature vector output by the CNN module, this paper introduces the DLST module for global feature extraction.

Unlike the traditional ST, the primary focus here is to capture long-range dependencies and global context in one-dimensional data, as illustrated in Figure 3.

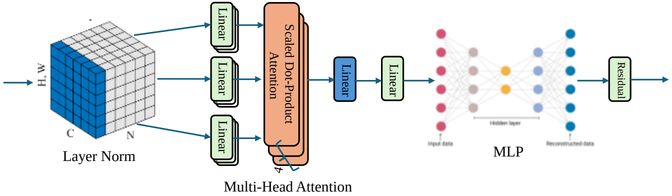


Figure 3 DLST flow chart.

The input is first divided into non-overlapping windows of size W, where W=7 in your model Each window represents a local portion of the input sequence that will be processed separately by the WMHA mechanism.

Secondly, the WMHSA mechanism, the core component of the DLST, is applied to windows of size W=7. Self-attention enables the model to dynamically assign weights to different parts of the input sequence, effectively capturing long-range dependencies. For each window, attention is computed independently.

For each token x_i , within a window, attention scores are computed by projecting the input sequence into query (Q), key (K), and value (V) matrices，As shown in Equation 4.

$$\begin{aligned} Q &= XW_Q(Query) \\ K &= XW_K(Key) \\ V &= XW_V(Value) \end{aligned} \tag{4}$$

Where $W_Q, W_K, W_V \in R^{d \times d_k}$ are learnable weight matrices and d_k is the dimension of the keys and queries.

The self-attention calculation for each pair of tokens i and j within the window is provided in Equation 5.

$$Attention(Q,K,V) = Soft\max(\frac{QK^T}{\sqrt{\mathrm{d}_k}})V \tag{5}$$

Where $\frac{1}{\sqrt{d_k}}$ is a scaling factor to prevent the dot products from becoming too large.

To capture multiple types of relationships simultaneously, WMHSA is used. The input is divided into multiple heads, each with its own set of projections for Q, and V.

For each head h, the attention output is computed as shown in Equation 6.

$$head_h = Attention(XW_{Qh}, XW_{Kh}, XW_{Vh}) \tag{6}$$

Where $W_{Qh}, W_{Kh}, W_{Vh} \in R^{d \times d_k}$ are the learned projection matrices for head h.

The outputs from each head are then concatenated and linearly transformed, as shown in Equation 7.

$$MultiHead(X) = Concat(head_1, head_2, head_3, head_4)W_O \tag{7}$$

Where $W_O \in R^{Hd_k \times d}$ is the output projection matrix.

In a standard window-based attention mechanism, the attention only captures dependencies within each window, which might limit the global understanding of the input sequence. The DLST addresses this by introducing shifted windows. This allows the model to capture cross-window dependencies without directly increasing computational complexity.

After the attention mechanism, the output is passed through a Feed-Forward Network (FFN)consisting of two fully connected layers with a GELU activation function in between, as shown in Equation 8.

$$FFN(X) = GELU(XW_1 + b_1)W_2 + b_2 \tag{8}$$

Where GELU is a smooth non-linear activation function that helps improve the learning of complex patterns.

Layer normalization, as shown in Equation 9, is applied between the attention layer and the feed-forward layer. This step not only stabilizes the learning process but also ensures smooth gradient flow, contributing to more effective training.

$$\hat{X} = \frac{X - \mu}{\sqrt{\sigma^2 + \varepsilon}} \tag{9}$$

Where μ and σ^2 are the mean and variance of input, and ε is a small constant to prevent division by zero.

Finally, residual connections are added between the attention and feed-forward layers to simplify optimization and enhance model convergence. These connections help preserve the original information while allowing the model to refine its predictions through the learned transformations, improving both training efficiency and performance.

C. Fusion Layer for CNN with Swin

The spectral features and spatial features are efficiently integrated, and the specific expression is shown in Equations10.

$$\begin{aligned} F_{spectral} &= \frac{1}{n}\sum_{i=1}^{n} S_i \\ F_{spatial} &= Soft\max(\frac{QW_i \cdot (KW_i)^T}{\sqrt{d_k}})VW_i \\ F_{fused} &= F_{Spatial} \oplus F_{Spectral} \end{aligned} \tag{10}$$

Where S_i is the spectral feature map after the i-th convolutional layer, n is the number of spectral bands, and $F_{spectral}$ is the reduced spectral feature representation. Q, K, and V are the query, key, and value matrices derived from the reshaped spectral features, W_i represents the window of attention, d_k is the dimensionality of the key, and $F_{spatial}$ is the reduced spatial feature representation. F_{fused} represents the fusion feature. And $\oplus$ represents a fusion operation, which is implicitly performed by passing the reshaped spectral features through ST, allowing the model to combine spectral and spatial features in a hierarchical manner.

This paper enhances the CNN block by incorporating 1D convolution in the CNN module to capture local features and identify short-term dependencies within the sequence, while the DLST is employed to capture long-range dependencies and understand the global context. Furthermore, the global average aggregation applied in the CNN block reduces the dimensionality of the output feature map, thereby facilitating more effective data management for the DLST module. This guarantees that the model can successfully process global dependencies without being overloaded by the input size. The window-based attention mechanism in the DLST permits the model to process larger input sequences without the quadratic complexity that is typically associated with full attention. When combined with the dimensionality reduction from the CNN block, this enables the model to efficiently scale to handle large datasets.

III. Experimental results

This section focus on the experimental settings and description of characteristic of four data-sets. It concludes with an analysis of the experimental results.

A. Data Description

The experiment employed four standard hyperspectral datasets. The datasets used were Houston-2013, Samson, KSC, and Botswana. The Houston-2013 dataset comprises 144 bands spanning the wavelength range of 380–1050 nm, with an image size of 349 x 1905 pixels. The dataset encompasses both urban and rural areas of Houston-2013 and offers 15 distinct categories of ground object labels. It is an appropriate means of assessing the resilience of HSI classification models. Figure 4 (a) shows the true color map and ground truth. The Samson dataset contains 156 bands (401–889 nm) and a resolution of 952 x 952 pixels. It captures soil, vegetation, and water scenes and is suitable for HSIC and spectral decomposition. Figure 4 (b) shows the true color map and ground truth. The Botswana dataset is also collected by AVIRIS, covering the Okavango Delta, containing 242 bands (400–2500 nm), a resolution of 30 m/pixel, an image size of 1476 x 256 pixels, and 14 types of land objects, suitable for environmental monitoring and land classification. Figure 4 (c) shows the true color map and ground truth. The KSC dataset was collected by NASA's AVIRIS sensor and contains 176 bands with a resolution of 18 meters/pixel and an image size of 512 x 614 pixels. It covers 13 types of ground objects, including vegetation, wetlands, and water bodies. Figure 4 (d) shows the true color map and ground truth.

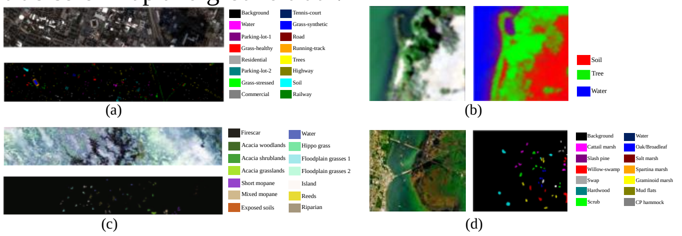


Figure 4. True color maps and ground truth for different datasets. (a)Houston-2013 dataset. (b) Samson dataset. (c) Botswana dataset. (d) KSC dataset.

B. Experimental Setting

The proposed algorithm is implemented using Python 3.8.5 and PyTorch 1.7.0. The hardware setup for training consists of an i7-10700K CPU and an NVIDIA GeForce RTX 3090 GPU.

To clearly demonstrate the performance comparison of different algorithms, this paper employs three commonly used evaluation metrics in HSI classification: overall accuracy (OA), average accuracy (AA), and the kappa coefficient. OA is used to evaluate the classification accuracy of the model across the entire dataset, and its calculation is presented in Formula 11.

$$OA = \frac{\sum_{i=1}^{n} TP_i}{N} \tag{11}$$

Where TP_i is the number of correctly classified samples of the i-th class, N is the total number of samples, and n is the total number of classes.

AA calculates the classification accuracy for each category and then takes the average value to assess the consistency of the model's performance across all categories. The calculation formula is presented in Equation 12.

$$AA = \frac{1}{n}\sum_{i=1}^{n}\frac{TP_i}{T_i} \tag{12}$$

Where T_i is the total number of true samples of the i-th class.

The kappa coefficient is used to comprehensively evaluate the stability and consistency of the classification results. Its calculation formula is provided in Equation 13.

$$kappa = \frac{OA - PE}{1 - PE} \tag{13}$$

Where PE represents the expected accuracy, which indicates the probability of the model correctly classifying samples under random classification. The calculation formula is provided in Equation 19.

$$PE = \sum_{i=1}^{n}\frac{(T_i \times P_i)}{N^2} \tag{14}$$

Where P_i is the number of samples predicted by the model to be of class i.

In addition, we use training time as another metric to evaluate the model's performance and further validate the effectiveness of the proposed algorithm by varying the ratio of different training samples. This study also compares the proposed algorithm with several state-of-the-art HSI classification methods, including CNN, KNN, LSTM, RNN, and GTFN. Ablation experiments are conducted on the proposed algorithm to further assess its effectiveness. For a fair comparison, all experiments are performed in the same environment, using the hyperparameters and recommended sample sizes as specified in the original papers.

C. Comparative Experimental Analysis

This section compares the performance of eight classic classification algorithms (CNN [31], CNN Encoder [44], SVM [13], KNN [14], LSTM [45], RNN [46], GTFN [47], and CSTFNet) on four hyperspectral datasets: Botswana, Samson, KSC, and Houston-2013. Tables 2 to 5 summarize the classification accuracy, overall accuracy (OA), average accuracy (AA), and Kappa coefficient of each algorithm in each category. The comparison of these indicators clearly shows that deep learning-based models generally outperform traditional machine learning algorithms, especially on more complex datasets.

As shown in Table 3, the performance of traditional machine learning algorithms such as SVM and KNN is relatively poor. In categories 1, 2, 6, 7, and 13, the accuracy of SVM is 0, while KNN performs equally poorly, with an accuracy of 0 in categories 7 and 13. In particular, on category 13, the performance of KNN is extremely limited, with an accuracy of only 6.45% and 4.35%, respectively. In contrast, deep learning models performed well in most categories, with CNN Encoder and GTFN achieving high classification accuracy in multiple categories. The CSTFNet model proposed in this study performed particularly well in 7 and 13 categories, achieving accuracies of 51.61% and 47.83% respectively, significantly outperforming other algorithms.

Table 2 shows that the overall accuracy (OA) of the CSTFNet model is 97.01%, the average accuracy (AA) is 97.3%, and the Kappa coefficient is 95.5%. In the water category, CSTFNet achieved an accuracy of 99.36%, which is close to perfect; in the tree category, the accuracy reached 99.55%, which is significantly better than other comparison methods. On the more challenging KSC dataset, as shown in Table 4, traditional algorithms (SVM, KNN) performed poorly in multiple categories, while CSTFNet performed well in category 12, with an accuracy of 64.25%, far exceeding other algorithms. In addition, the CSTFNet model also showed better performance in categories 5 and 9, highlighting its stability and adaptability.

As shown in Table 5, the classification performance of the CSTFNet model is close to the best, with an accuracy of 1.000 in categories 2, 3, 4, and 6, which is significantly better than other models, proving its efficiency in processing spectral and spatial information of HSI. The classification accuracy of the CSTFNet model remains high in all categories, with an OA of 99.6%, an AA of 99.61%, and a Kappa coefficient of 99.54%. These findings demonstrate that the CSTFNet model is capable of maintaining a high level of classification accuracy when processing more complex spectral data.

In conclusion, the CSTFNet model presented in this study demonstrates superior performance compared to traditional machine learning algorithms and other deep learning models on four hyperspectral datasets. In particular, it demonstrates enhanced robustness and generalization ability in the context of more complex datasets, such as Botswana and KSC. The CSTFNet model is able to effectively extract both spectral and spatial features in complex scenes by leveraging the combined advantages of CNN and DLST, thereby significantly improving classification accuracy. Compared with traditional methods, CSTFNet shows obvious advantages and wide applicability in HSIC tasks, further verifying its effectiveness.

Table 2. Quantitative comparison of the Samson dataset (Best in bold)

Class No.	CNN	CNN Encoder	SVM	KNN	LSTM	RNN	GTFN	Proposed
Soil	0.965	0.990	0.966	**0.984**	0.345	0.924	0.976	0.928
Tree	0.990	0.990	0.966	0.984	0.345	0.924	0.982	**0.996**
Water	0.985	0.990	0.966	0.984	0.345	0.924	**0.994**	**0.994**
OA(%)	98.01	99.28	96.57	98.39	34.46	92.35	**98.28**	97.01
AA(%)	98.15	**99.24**	96.79	98.50	33.33	92.93	98.39	97.30
Kappa×100	96.97	**98.91**	94.79	97.56	00.00	88.37	97.39	95.46

Table 3. Quantitative comparison of the Botswana dataset (Best in bold)

Class No.	CNN	CNN Encoder	SVM	KNN	LSTM	RNN	GTFN	Proposed
0	0.278	0.500	**0.685**	0.389	0.500	0.519	0.241	0.482
1	0.316	**0.526**	0.000	0.211	0.053	0.000	0.474	0.474
2	0.517	**0.667**	0.000	0.483	0.567	0.400	0.267	0.483
3	0.405	0.487	0.027	0.189	0.378	0.351	**0.432**	0.297
4	0.357	**0.571**	0.107	0.339	0.071	0.036	0.304	0.429
5	0.528	0.500	**0.778**	0.472	0.278	0.361	0.611	0.639
6	0.353	**0.471**	0.000	0.196	0.039	0.177	0.314	0.431
7	0.226	0.419	0.000	0.065	0.032	0.000	0.226	**0.516**
8	0.571	**0.651**	0.635	0.286	0.635	0.397	0.587	0.492
9	0.130	**0.544**	0.152	0.304	0.109	0.152	0.152	0.391
10	**0.435**	0.261	0.101	0.087	0.015	0.015	0.290	0.290
11	**0.717**	0.587	0.587	0.326	0.500	0.435	0.609	0.652
12	**0.780**	0.407	0.644	0.339	0.695	0.509	0.339	0.610
13	0.391	0.217	0.000	0.044	0.000	0.000	0.391	**0.478**
OA(%)	44.77	**49.54**	29.38	28.15	31.23	26.46	36.46	47.08
AA(%)	42.89	**48.62**	26.55	26.64	27.65	20.19	37.40	47.60
Kappa×100	40.02	**45.26**	23.34	21.97	25.00	23.93	31.17	42.62

Table 4. Quantitative comparison of the KSC dataset (Best in bold)

Class No.	CNN	CNN Encoder	SVM	KNN	LSTM	RNN	GTFN	Proposed
0	0.344	0.240	0.038	**0.414**	0.150	0.038	0.248	0.280
1	0.051	**0.240**	0.000	0.051	0.000	0.000	0.051	0.026
2	0.000	**0.240**	0.000	0.195	0.000	0.000	0.049	0.024
3	0.000	**0.240**	0.000	0.038	0.000	0.000	0.094	0.000
4	0.000	**0.240**	0.000	0.091	0.000	0.000	0.152	0.000
5	0.229	0.240	0.000	0.167	0.000	0.000	0.167	**0.375**

6	0.000	**0.240**	0.000	0.080	0.000	0.000	0.080	0.000
7	**0.287**	0.240	0.000	0.230	0.000	0.000	0.276	0.161
8	0.198	**0.240**	0.000	0.229	0.000	0.000	0.177	0.146
9	0.105	0.240	0.012	0.186	0.000	0.000	0.198	**0.256**
10	0.189	**0.240**	0.000	0.203	0.000	0.135	0.176	0.108
11	0.135	**0.240**	0.153	0.135	0.000	0.180	0.207	0.198
12	0.560	0.240	**0.855**	0.347	0.000	0.845	0.611	0.643
OA(%)	24.64	23.97	18.12	23.49	15.05	19.08	**26.37**	25.70
AA(%)	16.14	18.50	08.14	18.20	07.69	02.63	**19.12**	17.05
Kappa×100	13.12	14.51	00.86	13.76	00.00	09.22	**15.91**	14.28

Table 5. Quantitative comparison of the Houston-2013 dataset (Best in bold)

Class No.	CNN	CNN Encoder	SVM	KNN	LSTM	RNN	GTFN	Proposed
0	**1.000**	**1.000**	0.955	0.980	0.970	0.928	**1.000**	**0.984**
1	0.909	0.805	0.948	**0.980**	0.961	0.973	0.948	0.974
2	**1.000**	0.845	1.000	0.980	0.973	**1.000**	**1.000**	**1.000**
3	0.986	0.928	0.986	0.980	0.971	0.970	0.971	**1.000**
4	**1.000**	0.983	1.000	0.980	0.967	0.894	**1.000**	**1.000**
5	0.890	0.817	0.805	**0.980**	0.915	0.973	0.890	0.976
6	0.952	0.952	**1.000**	0.980	0.988	0.954	**1.000**	**1.000**
7	95.85	89.92	95.85	98.02	97.23	95.65	97.04	**99.60**
8	96.24	90.43	96.26	98.21	97.40	95.88	97.28	**99.61**
9	95.15	88.24	95.15	97.69	96.77	94.92	96.54	**99.54**
10	**1.000**	**1.000**	0.955	0.980	0.970	0.928	**1.000**	0.984
11	0.909	0.805	0.948	**0.980**	0.961	0.973	0.948	0.974
12	**1.000**	0.845	**1.000**	0.980	0.973	**1.000**	**1.000**	**1.000**
OA(%)	0.986	0.928	0.986	0.980	0.971	0.970	0.971	**1.000**
AA(%)	1.000	0.983	1.000	0.980	0.967	0.894	1.000	**1.000**
Kappa×100	0.890	0.817	0.805	0.980	0.915	0.973	0.890	**0.976**

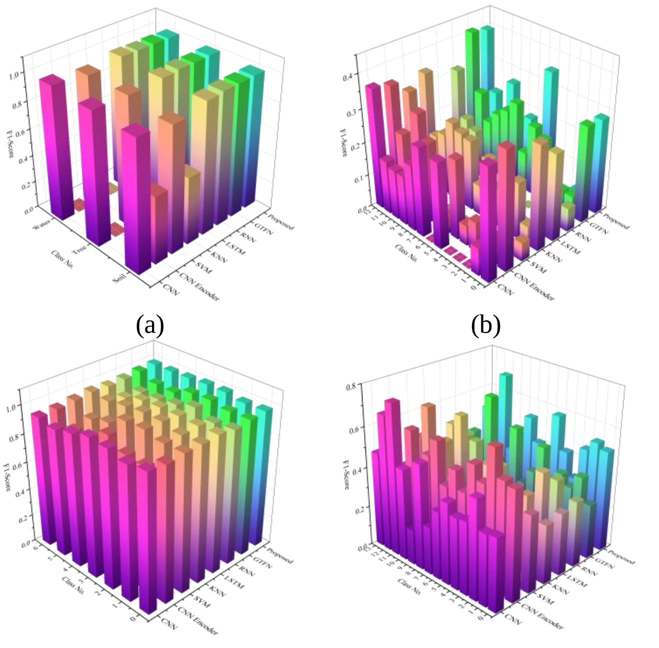
Table 6. Comparison of OA and Kappa values of different algorithms on Houston-2013 dataset (Best in bold)

	SSFTT	GAHT	DCTN	morphFormer	HiT	SS-TMNet	Proposed
OA(%)	98.91	98.05	98.17	97.82	93.45	95.92	**99.60**
Kappa×100	98.39	98.81	98.31	97.98	93.94	96.22	**99.54**

This paper not only compares the above classic classification algorithms, but also compares 6 latest classification algorithms, including innovative methods based on Transformer such as SSFTT [48], DCTN [49] and SS-TMNet [50], and methods based on multi-scale transformation such as GAHT [51] and HiT [52]. In addition, we also compare with the customized latest model morphFormer [53], as shown in Table 5, showing the OA and Kappa values of the Houston-2013 dataset. The results show that MCST2CNNoutperforms the latest algorithms [48-53].

Additionally, the F1-Score of different algorithms across the four datasets was calculated, as shown in Fig. 5. The results indicate that the CSTFNet model consistently outperforms other comparative methods in most categories, demonstrating high classification accuracy and robustness. Overall, CSTFNet achieves near-optimal classification results on the Houston-2013 and Samson datasets, particularly in several categories where the F1-Scores approach 1, underscoring its strong feature extraction capabilities. On the more complex datasets, such as Botswana and KSC, CSTFNet also exhibits superior stability. Although its F1-Score in certain categories is slightly lower than that of the GTFN model, it still maintains leading classification accuracy overall.

In contrast, traditional machine learning methods, such as SVM and KNN, perform poorly across multiple datasets, particularly on Botswana and KSC, where lower F1-Scores suggest their difficulty in effectively handling the complex features of HSI. While deep learning methods (e.g., CNN, GTFN) perform better in most scenarios, they are still unable to surpass the overall performance of the CSTFNet model, particularly in tasks that require the integration of spectral and spatial information.



(a)

(b)

(c)

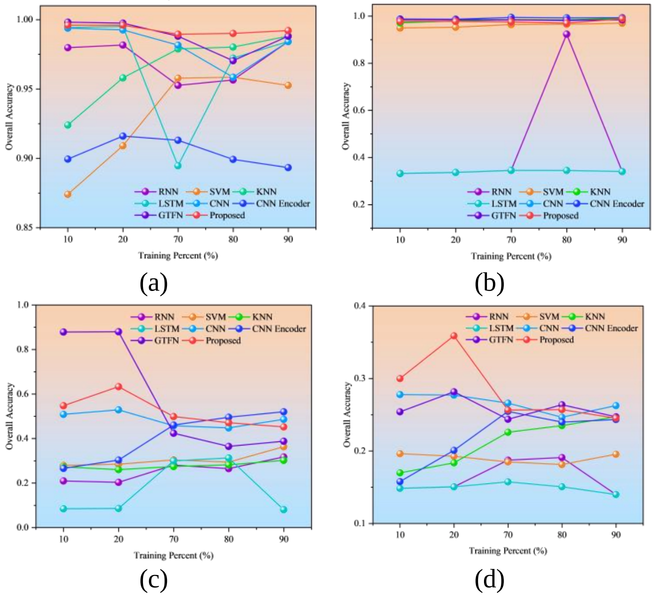
(d)

Fig. 5 Corresponding F1-Scores of different algorithms on different datasets. (a) Samson dataset; (b) KSC dataset; (c) Houston-2013 dataset; (d) Botswana dataset.

D. Training Samples Analysis

To further verify the effectiveness and robustness of the CSTFNet algorithm, this section calculates the overall accuracy (OA) and compares the performance of the eight algorithms mentioned earlier using different training sample ratios, as illustrated in Fig. 6. The ratios of training samples to the total dataset size for the four datasets are set at 0.1, 0.2, 0.7, 0.8, and 0.9. The experimental results demonstrate that the overall accuracy of all models improves as the proportion of training samples increases. However, traditional machine learning models, such as SVM and KNN, exhibit less stability with smaller training sample proportions, showing significant performance fluctuations, particularly on smaller-scale datasets. In contrast, deep learning models, including RNN, CNN, and the proposed CSTFNet model, display greater robustness, with classification accuracy markedly improving as the size of the training set increases.

Among these models, the proposed CSTFNet model achieves excellent classification performance across all datasets and training ratios, particularly excelling at higher training sample proportions, where its overall accuracy significantly surpasses that of the other models. These results validate the proposed model’s ability to maintain high classification accuracy across various datasets, further demonstrating its effectiveness and broad applicability in HSIC tasks.



(a)

(b)

(c)

(d)

Fig. 7 OA of different proportions of training samples on four datasets.(a) Houston-2013; (b) Samson; (c) Botswana; (d) KSC.

E. Time Comparison

This section discusses the comparison of training time across the Houston-2013, Samson, KSC, and Botswana datasets, using an 80:20 training-to-testing ratio. As shown in Table 7, the training times of traditional machine learning models (SVM and KNN) are significantly shorter compared to deep learning-based models, particularly on smaller datasets such as Samson and Botswana. For instance, KNN requires only 0.0041 seconds on the Houston-2013 dataset and 0.01 seconds on the Samson dataset.

In contrast, deep learning models require substantially more time, especially on larger datasets. For example, LSTM takes 808.95 seconds to train on the KSC dataset. The proposed method demonstrates a balanced performance, with training times significantly lower than those of RNN and LSTM, but slightly higher than CNN on most datasets. For instance, on the Houston-2013 dataset, the proposed model takes 112.3 seconds, which is higher than CNN's 56.56 seconds, but considerably lower than RNN's 130.45 seconds. This demonstrates that the proposed method strikes an effective balance between computational efficiency and classification accuracy, making it suitable for practical applications where both performance and efficiency are critical.

Table 7. Training time for different algorithms on 4 datasets, training test ratio 80-20

Model	Houston-2013	Samson	KSC	Botswana
RNN	130.45s	449.92s	817.29s	391.02s
SVM	0.3074s	2.56s	18.43s	2.83s
KNN	0.0041s	0.01s	0.008s	0.02s
LSTM	120.83s	604.9s	808.95s	395.7s
CNN	56.56s	113.35s	75.56s	64.79s
CNN Encoder	73.05s	2.96s	115.32s	33.26s
GTFN	66.9s	135.59s	102.25s	71.9s
Proposed	112.3s	207.06s	152.29s	110.09s

F. *Visual Analysis of Classification Results*

This paper presents a visual representation of the classification effects of various classification methods, accompanied by a qualitative analysis. As illustrated in Figures 7-10, the classification effects on the KSC, Botswana, Houston-2013, and Samson datasets are presented. From the figures, it is evident that the algorithm proposed in this paper exhibits superior performance on the four datasets in comparison to other algorithms. It demonstrates a more comprehensive ability to maintain the clarity and integrity of the region boundaries.

It is evident that traditional algorithms, such as KNN and SVM, are unable to effectively suppress noise and are susceptible to misclassified scatter. In contrast, deep learning-based methods demonstrate superior performance, illustrating the robust feature extraction capacity of convolutional neural networks. However, deep modelling algorithms, such as RNN and LSTM, exhibit suboptimal capability in capturing intricate features and boundary regions, and remain unable to attain enhanced classification outcomes. By integrating CNN and ST, this algorithm effectively minimizes classification noise and extracts comprehensive features and edge features, showcasing robust classification performance

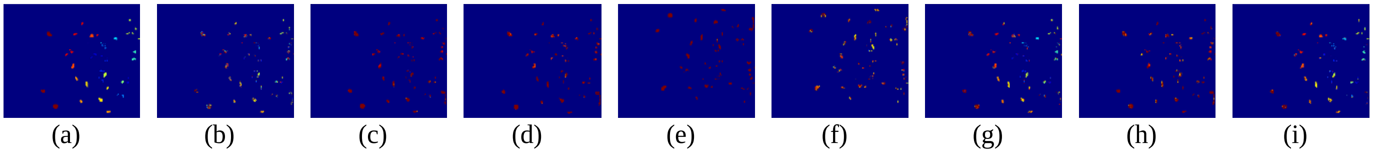

(a) (b) (c) (d) (e) (f) (g) (h) (i)

Fig.8 Prediction map on KSC dataset. (a) Ground truth. (b) KNN. (c) SVM. (d) RNN. (e) LSTM. (f) CNN. (g) CNN Encoder. (h) GTFN. (i) Proposed

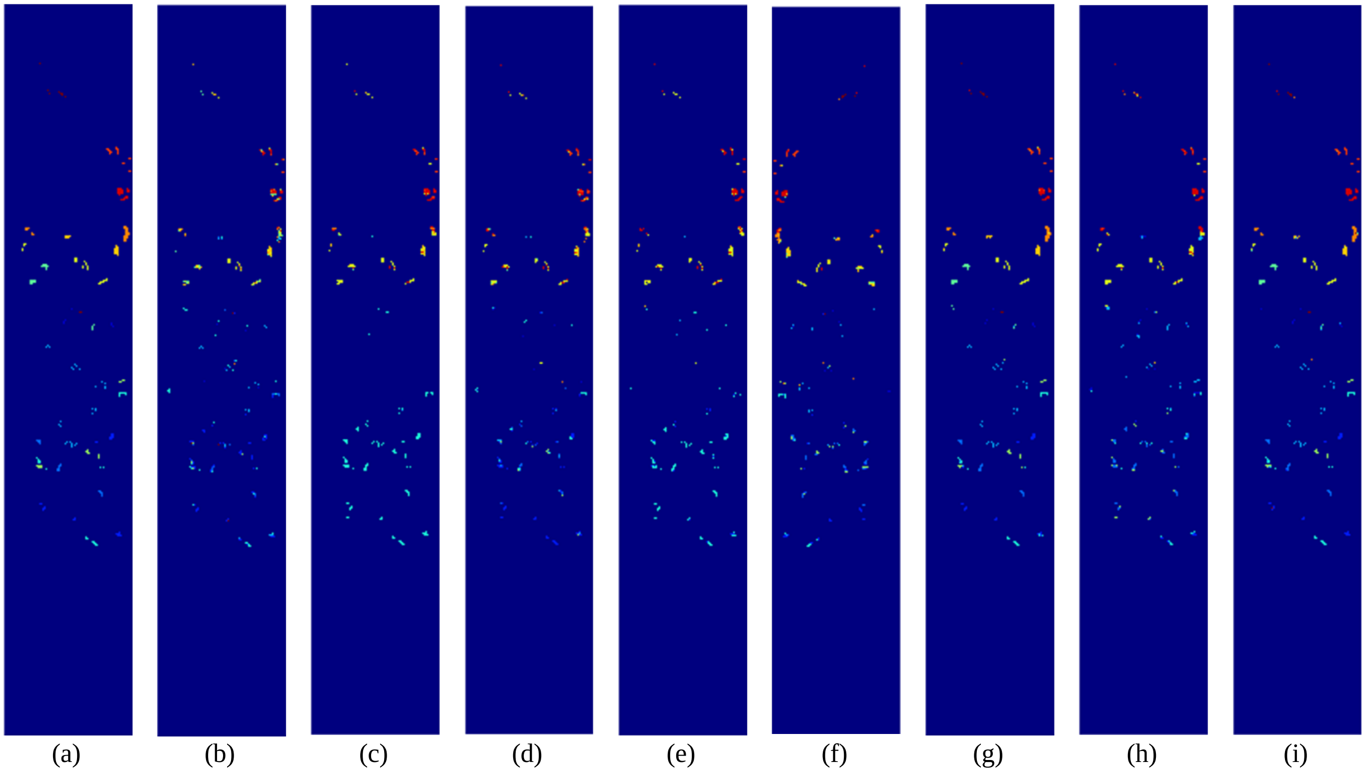

(a) (b) (c) (d) (e) (f) (g) (h) (i)

Fig.9 Prediction map on Botswana dataset. (a) Ground truth. (b) KNN. (c) SVM. (d) RNN. (e) LSTM. (f) CNN. (g) CNN Encoder. (h) GTFN. (i) Proposed

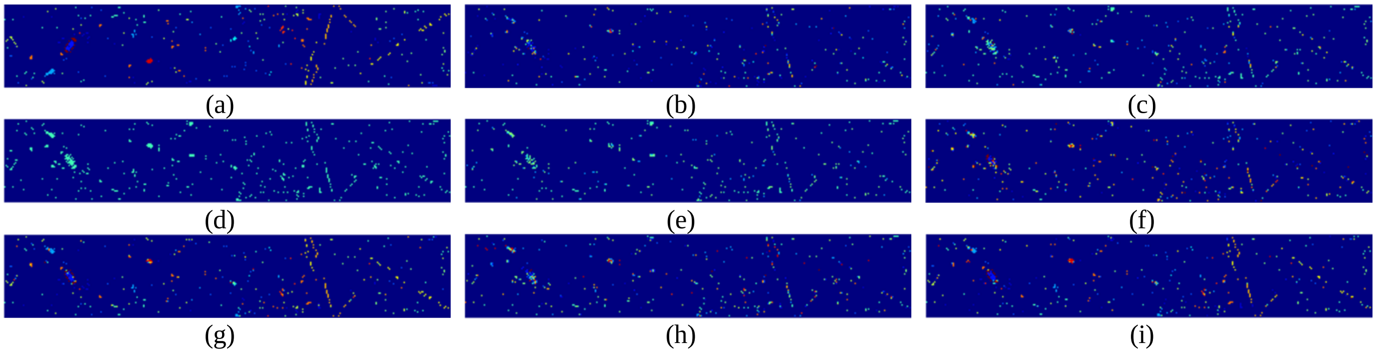

(a) (b) (c)
(d) (e) (f)
(g) (h) (i)

Fig.10 Prediction map on Houston-2013 dataset. (a) Ground truth. (b) KNN. (c) SVM. (d) RNN. (e) LSTM. (f) CNN. (g) CNN Encoder. (h) GTFN. (i) Proposed

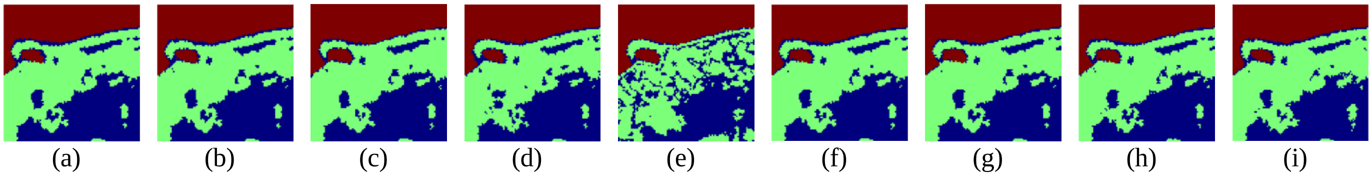

(a) (b) (c) (d) (e) (f) (g) (h) (i)

Fig.11 Prediction map on Samson dataset. (a) Ground truth. (b) KNN. (c) SVM. (d) RNN. (e) LSTM. (f) CNN. (g) CNN Encoder. (h) GTFN. (i) Proposed

G. Ablation Experiment

In order to further validate the effectiveness of the CSTFNet model, this section conducts ablation experiments using the Houston-2013 dataset as an example. The aim of these experiments is to systematically evaluate the contribution of different components in the proposed model by removing individual or combined modules. The results are presented in Table 8, which demonstrates the classification performance of four distinct model configurations with varying training-testing ratios. The proposed CSTFNet model demonstrates consistent accuracy across all training-testing ratios, with a notable decline in performance when any module is removed. In particular, the AA value of CSTFNet is 0.993, the OA value is 0.992, and the Kappa coefficient is 0.991 at a ratio of 90:10, which evinces the superiority of the full model when a larger proportion of training data is available.

As evidenced in Table 8, the removal of the CNN module results in a notable decline in model performance. Specifically, at a training-test ratio of 10:90, both OA and AA drop from 0.996 to 0.969, and the kappa coefficient drops from 0.995 to 0.964. Similarly, the removal of the ST module also results in a decline in performance, although the outcomes are marginally superior in comparison to the model that lacks the CNN module. Specifically, at a training-test ratio of 10:90, AA decreased from 0.996 to 0.968, OA decreased from 0.996 to 0.967, and the kappa coefficient decreased from 0.995 to 0.962. The removal of both the CNN module and the Transformer module resulted in the lowest performance, underscoring the pivotal role these components play in attaining optimal classification accuracy. Specifically, at a training-test ratio of 10:90, both OA and AA decreased from 0.996 to 0.965, and the kappa coefficient decreased from 0.995 to 0.959. These findings corroborate the efficacy and resilience of the integrated model architecture in HSIC.

Table 8. Results of ablation experiments on the Houston-2013 dataset

Ablation Settings		AA				
CNN	ST	10:90	20:80	70:30	80:20	90:10
×	×	0.965	0.973	0.973	0.960	0.948
✓	×	0.968	0.969	0.961	0.962	0.975
×	✓	0.969	0.976	0.975	0.959	0.974
✓	✓	0.996	0.996	0.990	0.991	0.993
		OA				
×	×	0.965	0.971	0.971	0.957	0.945
✓	×	0.967	0.965	0.958	0.959	0.972
×	✓	0.969	0.974	0.972	0.955	0.972
✓	✓	0.996	0.996	0.990	0.990	0.992
		Kappa				
×	×	0.959	0.966	0.966	0.949	0.935
✓	×	0.962	0.960	0.951	0.952	0.968
×	✓	0.964	0.969	0.968	0.947	0.968
✓	✓	0.995	0.995	0.988	0.988	0.991

IV. Conclusion

In order to make full use of the spatial-spectral information of HSI, we proposed the CSTFNet model, which is mainly composed of CNN blocks and DLST blocks. In order to solve the dimensionality problem of HSI, one-dimensional convolution is introduced into the CNN block, and the spectral features of HSI are fully extracted. Since the spatial information of HSI is very rich, they are processed by two ST blocks, and their window-based attention mechanism is fully utilized for hierarchical learning, which effectively captures the spatial information of HSI. The combination of CNN and ST realizes the effective fusion of spectral-spatial features and demonstrates higher classification accuracy than existing algorithms. The classification accuracy of CSTFNet on Houston-2013, Samson, KSC and Botswana datasets is 99.6%, 97.8%, 63.3% and 35.9% respectively.

CSTFNet demonstrates a powerful capability for spectral-spatial feature extraction and holds significant promise for applications in the field of HSIC. However, certain limitations must be acknowledged. The relatively low accuracy on datasets such as KSC and Botswana highlights potential challenges in handling datasets with high spectral complexity or limited training samples. Additionally, the computational cost of integrating CNN and transformer-based architectures, especially for large datasets, requires further exploration and optimization. In future work, we aim to address these limitations by investigating advanced techniques to enhance generalization on complex datasets and reduce computational overhead. Further integration of cutting-edge advancements in CNNs, transformers, and hybrid architectures will be explored to achieve even greater classification performance and broader applicability in hyperspectral image analysis.

REFERENCES

[1] Khan A, Vibhute A D, Mali S, et al. A systematic review on hyperspectral imaging technology with a machine and deep learning methodology for agricultural applications[J]. Ecological Informatics, 2022, 69: 101678.

[2] Bhatti, M. A., Zeeshan, Z., Syam, M. S., Bhatti, U. A., Khan, A., Ghadi, Y. Y., ... & Afzal, T. (2024). Advanced plant disease segmentation in precision agriculture using optimal dimensionality reduction with fuzzy c-means clustering and deep learning. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.

[3] Han, H., Zeeshan, Z., Talpur, B. A., Sadiq, T., Bhatti, U. A., Awwad, E. M., ... & Ghadi, Y. Y. (2024). Studying long term relationship between carbon Emissions, Soil, and climate Change: Insights from a global Earth modeling Framework. International Journal of Applied Earth Observation and Geoinformation, 130, 103902.

[4] Lu B, Dao P D, Liu J, et al. Recent advances of hyperspectral imaging technology and applications in agriculture[J]. Remote Sensing, 2020, 12(16): 2659.

[5] Stuart M B, McGonigle A J S, Willmott J R. Hyperspectral imaging in environmental monitoring: A review of recent developments and technological advances in compact field deployable systems[J]. Sensors, 2019, 19(14): 3071.

[6] Lelewer D A, Hirschberg D S. Data compression[J]. ACM Computing Surveys (CSUR), 1987, 19(3): 261-296.

[7] Keshava N, Mustard J F. Spectral unmixing[J]. IEEE signal processing magazine, 2002, 19(1): 44-57.

[8] Zhao Z Q, Zheng P, Xu S, et al. Object detection with deep learning: A review[J]. IEEE transactions on neural networks and learning systems, 2019, 30(11): 3212-3232.

[9] Wu H, Xian J, Wang J, et al. Missing data recovery using reconstruction in ocean wireless sensor networks[J]. Computer Communications, 2018, 132: 1-9.

[10] Li S, Song W, Fang L, et al. Deep learning for hyperspectral image classification: An overview[J]. IEEE Transactions on Geoscience and Remote Sensing, 2019, 57(9): 6690-6709.

[11] Bhatti, U. A., Huang, M., Neira-Molina, H., Marjan, S., Baryalai, M., Tang, H., ... & Bazai, S. U. (2023). MFFCG–Multi feature fusion for hyperspectral image classification using graph attention network. Expert Systems with Applications, 229, 120496.

[12] Datta D, Mallick P K, Bhoi A K, et al. Hyperspectral image classification: Potentials, challenges, and future directions[J]. Computational intelligence and neuroscience, 2022, 2022(1): 3854635.

[13] Pathak D K, Kalita S K, Bhattacharya D K. Hyperspectral image classification using support vector machine: a spectral spatial feature based approach[J]. Evolutionary Intelligence, 2022: 1-15.

[14] Li R, Li S. Multimedia image data analysis based on knn algorithm[J]. Computational Intelligence and Neuroscience, 2022, 2022(1): 7963603.

[15] Tong F, Zhang Y. Spectral–spatial and cascaded multilayer random forests for tree species classification in airborne hyperspectral images[J]. IEEE Transactions on Geoscience and Remote Sensing, 2022, 60: 1-11.

[16] Saha D, Manickavasagan A. Machine learning techniques for analysis of hyperspectral images to determine quality of food products: A review[J]. Current Research in Food Science, 2021, 4: 28-44.

[17] Ahmad M, Shabbir S, Roy S K, et al. Hyperspectral image classification—Traditional to deep models: A survey for future prospects[J]. IEEE journal of selected topics in applied earth observations and remote sensing, 2021, 15: 968-999.

[18] He L, Li J, Liu C, et al. Recent advances on spectral–spatial hyperspectral image classification: An overview and new guidelines[J]. IEEE Transactions on Geoscience and Remote Sensing, 2017, 56(3): 1579-1597.

[19] Pan Y, Zhang H, Chen Y, et al. Applications of hyperspectral imaging technology combined with machine learning in quality control of traditional chinese medicine from the perspective of artificial intelligence: a review[J]. Critical Reviews in Analytical Chemistry, 2023: 1-15.

[20] Mazzia V, Salvetti F, Chiaberge M. Efficient-capsnet: Capsule network with self-attention routing[J]. Scientific reports, 2021, 11(1): 14634.

[21] Li Y, Wang Q, Zhang J, et al. The theoretical research of generative adversarial networks: an overview[J]. Neurocomputing, 2021, 435: 26-41.

[22] Bo D, Wang X, Shi C, et al. Beyond low-frequency information in graph convolutional networks[C]//Proceedings of the AAAI conference on artificial intelligence. 2021, 35(5): 3950-3957.

[23] Zhang J, Li X, Li J, et al. Rethinking mobile block for efficient attention-based models[C]//2023 IEEE/CVF International Conference on Computer Vision (ICCV). IEEE Computer Society, 2023: 1389-1400.

[24] Lee H, Kwon H. Going deeper with contextual CNN for hyperspectral image classification[J]. IEEE Transactions on Image Processing, 2017, 26(10): 4843-4855.

[25] Zhang M, Li W, Du Q. Diverse region-based CNN for hyperspectral image classification[J]. IEEE Transactions on Image Processing, 2018, 27(6): 2623-2634.

[26] Li Z, Liu F, Yang W, et al. A survey of convolutional neural networks: analysis, applications, and prospects[J]. IEEE transactions on neural networks and learning systems, 2021, 33(12): 6999-7019.

[27] Kiranyaz S, Avci O, Abdeljaber O, et al. 1D convolutional neural networks and applications: A survey[J]. Mechanical systems and signal processing, 2021, 151: 107398.

[28] Li Z, Liu F, Yang W, et al. A survey of convolutional neural networks: analysis, applications, and prospects[J]. IEEE transactions on neural networks and learning systems, 2021, 33(12): 6999-7019.

[29] Ji S, Xu W, Yang M, et al. 3D convolutional neural networks for human action recognition[J]. IEEE transactions on pattern analysis and machine intelligence, 2012, 35(1): 221-231.

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

[31] Roy S K, Krishna G, Dubey S R, et al. HybridSN: Exploring 3-D–2-D CNN feature hierarchy for hyperspectral image classification[J]. IEEE Geoscience and Remote Sensing Letters, 2019, 17(2): 277-281.

[32] Gong H, Li Q, Li C, et al. Multiscale information fusion for hyperspectral image classification based on hybrid 2D-3D CNN[J]. Remote Sensing, 2021, 13(12): 2268.

[33] Hinton G E, Krizhevsky A, Wang S D. Transforming auto-encoders[C]//Artificial Neural Networks and Machine Learning–ICANN 2011: 21st International Conference on Artificial Neural Networks, Espoo, Finland, June 14-17, 2011, Proceedings, Part I 21. Springer Berlin Heidelberg, 2011: 44-51.

[34] Zhang H, Meng L, Wei X, et al. 1D-convolutional capsule network for hyperspectral image classification[J]. arXiv preprint arXiv:1903.09834, 2019.

[35] Hong D, Han Z, Yao J, et al. SpectralFormer: Rethinking hyperspectral image classification with transformers[J]. IEEE Transactions on Geoscience and Remote Sensing, 2021, 60: 1-15.

[36] Mou L, Lu X, Li X, et al. Nonlocal graph convolutional networks for hyperspectral image classification[J]. IEEE Transactions on Geoscience and Remote Sensing, 2020, 58(12): 8246-8257.

[37] Huang X, Dong M, Li J, et al. A 3-d-swin transformer-based hierarchical contrastive learning method for hyperspectral image classification[J]. IEEE Transactions on Geoscience and Remote Sensing, 2022, 60: 1-15.

[38] Han K, Wang Y, Chen H, et al. A survey on vision transformer[J]. IEEE transactions on pattern analysis and machine intelligence, 2022, 45(1): 87-110.

[39] Gu A, Dao T. Mamba: Linear-time sequence modeling with selective state spaces[J]. arXiv preprint arXiv:2312.00752, 2023.

[40] Zhu L, Liao B, Zhang Q, et al. Vision mamba: Efficient visual representation learning with bidirectional state space model[J]. arXiv preprint arXiv:2401.09417, 2024.

[41] Li Y, Luo Y, Zhang L, et al. Mambahsi: Spatial-spectral mamba for hyperspectral image classification[J]. IEEE Transactions on Geoscience and Remote Sensing, 2024.

[42] Wang C, Huang J, Lv M, et al. A local enhanced mamba network for hyperspectral image classification[J]. International Journal of Applied Earth Observation and Geoinformation, 2024, 133: 104092.

[43] Zhang M, Li L, Wenxuan S H I, et al. VmambaSCI: Dynamic Deep Unfolding Network with Mamba for Compressive Spectral Imaging[C]//ACM Multimedia 2024.

[44] He J, Zhao L, Yang H, et al. HSI-BERT: Hyperspectral image classification using the bidirectional encoder representation from transformers[J]. IEEE Transactions on Geoscience and Remote Sensing, 2019, 58(1): 165-178.

[45] Yin J, Qi C, Chen Q, et al. Spatial-spectral network for hyperspectral image classification: A 3-D CNN and Bi-LSTM framework[J]. Remote Sensing, 2021, 13(12): 2353.

[46] Zhou W, Kamata S, Luo Z, et al. Multiscanning strategy-based recurrent neural network for hyperspectral image classification[J]. IEEE Transactions on Geoscience and Remote Sensing, 2021, 60: 1-18.

[47] Yang A, Li M, Ding Y, et al. GTFN: GCN and transformer fusion with spatial-spectral features for hyperspectral image classification[J]. IEEE Transactions on Geoscience and Remote Sensing, 2023.

[48] Sun L, Zhao G, Zheng Y, et al. Spectral–spatial feature tokenization transformer for hyperspectral image classification[J]. IEEE Transactions on Geoscience and Remote Sensing, 2022, 60: 1-14.

[49] Zhou Y, Huang X, Yang X, et al. DCTN: Dual-Branch Convolutional Transformer Network With Efficient Interactive Self-Attention for Hyperspectral Image Classification[J]. IEEE Transactions on Geoscience and Remote Sensing, 2024.

[50] Huang X, Zhou Y, Yang X, et al. Ss-tmnet: Spatial–spectral transformer network with multi-scale convolution for hyperspectral image classification[J]. Remote Sensing, 2023, 15(5): 1206.

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

[52] Yang X, Cao W, Lu Y, et al. Hyperspectral image transformer classification networks[J]. IEEE Transactions on Geoscience and Remote Sensing, 2022, 60: 1-15.

[53] Roy S K, Deria A, Shah C, et al. Spectral–spatial morphological attention transformer for hyperspectral image classification[J]. IEEE Transactions on Geoscience and Remote Sensing, 2023, 61: 1-15.

[54] Ayas S, Tunc-Gormus E. SpectralSWIN: a spectral-swin trans-former network for hyperspectral image classification[J]. In-ternational Journal of Remote Sensing, 2022, 43(11): 4025-4044.

[55] Long Y, Wang X, Xu M, et al. Dual self-attention Swin trans-former for hyperspectral image super-resolution[J]. IEEE Transactions on Geoscience and Remote Sensing, 2023, 61: 1-12.