



Synergistic spectral and spatial feature analysis with transformer and convolution networks for hyperspectral image classification

Dhirendra Prasad Yadav^{1,2} · Deepak Kumar² · Anand Singh Jalal¹ · Ankit Kumar³ · B. Kada⁴

Received: 20 October 2023 / Revised: 15 November 2023 / Accepted: 12 December 2023 / Published online: 1 February 2024
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2024

Abstract

Hyperspectral imaging (HSI) contains several land cover objects with rich spatial and spectral features. By utilizing these features, deep convolution neural networks (CNN) improved HSI classification accuracy. However, shallow CNN lacks global co-relation of the spatial and spectral features. Further, by increasing the convolution layers, trainable parameters also increase. Hence, computation cost significantly increases. In this study, a fusion-based HFTNet model is designed that extracts features via convolution and transformer block to improve classification performance. In the proposed HFTNet, the convolution block extracts local semantic features, and the transformer block captures the attention-based global features. We reduced the computation costs by dividing the query vector into two parts and passing it to convolution and transformer blocks for feature extraction. Finally, features are combined to generate enhanced semantic local and global features. The effectiveness of the proposed method is tested on four datasets and achieved an accuracy of 99.34% (UP), 97.95% (IP), 99.70% (SV), and 84.23% (KSC). We found that HFTNet takes less computation time and achieves much better classification accuracy than other methods.

Keywords Hyperspectral imaging (HSI) · CNN · Semantic features · Classification · Transformer

1 Introduction

Advanced spectrometers capture hyperspectral imaging (HSI) with numerous spectral and spatial characteristics. The

continuous spectral spectrum extends from visible light to infrared, boosting the visibility of ground objects [1]. Categorizing each pixel, HSI has found widespread application in mineral exploration, precise agriculture, and environmental monitoring. HSI has been broadly classified as either spectral, spatial, or hybrid exploitation of spatial and spectral information [2]. Since each ground item has a unique spectral characteristic, the spectrum-based method converts into a short pattern recognition that identifies spectral vectors using a classifier [3]. However, external factors such as lightning, environment, and atmosphere influence the generation of spectral vectors and create noise, or so-called spectral variability, leading to substandard performance [4]. To smoothen the spectral difference of ground objects, the spatial information described in [5] is often considered, and numerous techniques based on joint spectral-spatial information have been published [6–12].

Several handcrafted feature-based HSI classification approaches exist, including the k-nearest neighbor [13, 14], Bayesian estimation approach [15], multinomial logistic regression [16, 17], and support vector machine (SVM)

✉ Dhirendra Prasad Yadav
dhirendra.yadav@gla.ac.in

Deepak Kumar
deepak.kumar@nitm.ac.in

Anand Singh Jalal
asjalal@gla.ac.in

Ankit Kumar
iiiita.ankit@gmail.com

B. Kada
bkada@kau.edu.sa

¹ Department of Computer Engineering and Applications, G.L.A. University, Mathura, U.P., India

² Department of Computer Engineering, NIT Meghalaya, Shillong, Meghalaya, India

³ Department of Information Technology, Guru Ghasidas Vishwavidyalaya, Bilaspur, Chattisgarh, India

⁴ Department of Aerospace Engineering, King Abdulaziz University, Jeddah, Saudi Arabia

[18–21]. These approaches are incapable of noise suppression and lack spatial-spectral characteristics. The spatial variability of spectral information [22] and extracting discriminative and most informative characteristics [23] remain substantial obstacles in HSI. Moreover, several band-reduction-based approaches, such as linear discriminant analysis (LDA) [24–26], independent component analysis (ICA) [27], and principal component analysis (PCA) [28, 29], fail to exploit the spatial correlation between pixels effectively. The use of deep convolutional neural networks, which can automatically extract high-dimensional spatial and spectral characteristics, has allowed researchers to overcome these obstacles.

The spatial and spectral properties of 3D HSI were extracted as 1D features using the stacked autoencoders (SAEs) and a deep belief network (DBN) Chen et al. [30, 31]. This was achieved at the expense of a great many spatial details. The Classification performance was improved using five layers of 1D-CNN to extract spatial information [32]. Before extracting spatial features from HIS with 2D-CNN, principal component analysis (PCA) was used in [33, 34] to minimize the dimension of HIS. By flattening the features, a dual branch of 2D and 1D CNN layers allowed for the joint exploitation of spatial and spectral characteristics [35]. In [36], a 3D convolutional neural network (CNN) model was used to improve classification accuracy using spatial and spectral information. Still, the enormous number of trainable parameters caused the computation cost to skyrocket. Later, [37, 38], using 3D and 2D CNN layers lowered the computational cost.

As shown by [39], dilated convolutional-guided feature filtering can help reduce the model's loss during training and validation. This strategy lowers spatial feature loss without diminishing the receptive field and can obtain distant features that boost classification performance. In [40], the residual connection-based SSRN model was used to exploit the spatial and spectral information, where the residual connection was added to each 3D layer, followed by batch normalization. But, due to the 3D layer and residual block, the computational cost was considerable. Multi-branch 3D CNN was used in [41] with an attention module for HSI object classification. However, more trainable parameters are needed when more 3D layers are used, which raises computation costs.

Recently, a powerful deep learning method called the transformer network was introduced to address natural image categorization from sequential data [42]. Transformer networks are superior at analyzing sequential data because they employ self-attention methods, unlike CNNs and RNNs. This presents a novel approach that can effectively be utilized for the HSI image land cover categorization. It is well-known that the self-attention technique is the central module in transformers and can capture global information by encoding position. Although they address the long-term

dependence of spectrum properties, they lack spatial-spectral integration data at the local level. Although they solve the long-term dependency of spectrum features, they miss local spatial-spectral integration data. In addition, local texture data and positional information loss occur because current transformer networks progressively encode spatial features via the flattening technique and linear projection. A 3D-Swin transformer-based technique was used in [43] to represent semantic-level images. The proposed technique used numerous transformer blocks to improve performance but considerably increased the calculational costs. Later, the authors of [44] used the SSFTT approach to reduce the computing cost of the 3D-Swin transformer-based technique by employing one 3D and two 2D layers followed by a transformer module, allowing extraction of global and local characteristics. Yet, classification performance in several classes may be improved. In [45], the fusion of convolution and transformer block in one technique was applied to improve the classification. The proposed method used parallel linear convolution blocks and transformer blocks to collect local and global data. The HSI cube is first turned into a sequence and then handed to the transformer block to collect the local and global features. This minimized the computational cost but at the expense of diminished performance.

To address the aforementioned issues associated with the HSI classification problem, a novel deep learning model called HFTNet is developed in this paper based on a dual block transformer. HSI's local spectral and spatial information is extracted via a 3D convolution block and 2D convolution layers based on the network architecture. As a result, improved classification performance is achieved by extracting global high-level semantic features using a dual-block transformer network. The significant contributions of the proposed method are as follows.

- (1) Initially, a 3D convolution layer is incorporated to focus on extracting spectral features, followed by implementing network-in-network structured 2D convolution layers explicitly designed to capture and analyze spatial features.
- (2) Integration of a transformer module is critical for spectral and spatial features. This module establishes local and global correlations within the data. The design facilitates a dual-pathway approach that enhances the representation of global semantic features and local pixel-level details.
- (3) The next stage involves sophisticated semantic and pixel pathways integration. This integration strategically distributes self-attention information across both pathways. Furthermore, the transformer's computational costs are reduced to optimize efficiency by splitting the query between the local convolution block and the transformer module.

- (4) The final step in this architecture involves a synergistic fusion of the CNN network with the dual block transformer module. This fusion technique enhances the overall classification accuracy significantly. The efficacy and superiority of HFTNet are experimentally proven through rigorous testing across four distinct datasets.

The rest of the paper is organized as follows.

In Sect. 2, we have discussed the proposed method, whereas Sect. 3 describes the quantitative and visual results. Finally, in Sect. 4, the conclusion is discussed in detail.

2 The proposed model

The proposed method system flow has been illustrated in Fig. 1. Let the hypercube of the HSI is $I \in R^{M \times N \times B}$, where M, N represents width and height, and B is the total bands. Each pixel in I contains the spatial and spectral feature, and their one hot encoding is given by $H = \{h_1, h_2, \dots, h_C\}$, where C is the different objects of land cover. In HSI, several continuous bands containing rich sets of spectral information are available due to the high number of bands, computation cost and redundancy increase. To overcome this problem, principal component analysis (PCA) is applied over band B . At the same time, maintaining the same spatial information. Let after PCA total band is D and the hypercube is represented by $Y \in R^{M \times N \times D}$.

The proposed method extracts the spectral and spatial features using 3D and 2D convolution layers. We added one 3D CNN layer and the 2D CNN layers to extract spectral and spatial features. We have not included several 2D CNN layers since the labeled training data is less, which may lead to overfitting. Therefore, based on the depth-wise separable method [46], CNN layers are utilized, which can enhance the performance and reduce the computation cost. The depth-wise 2D CNN layers filter per input is defined as.

$$Y_{h,w,b}^* = \sigma \left(\sum K_{i,j,b} Y_{h,w,b} + A_{i,j,b} \right), \quad (1)$$

where $Y_{h,w,b}^*$ = Features map, $K_{i,j,b}$ = Convolution kernel, $A_{i,j,b}$ = Bias.

2.1 Vision transformer preliminaries

The concept of vision transformer (ViT) was first used for Natural Language Processing (NLP) [42]. Later, this technique was extended to other fields like image classification, segmentation, object detection and image captioning. In ViT, one sequence is transformed into another with the help of an

encoder and decoder module. The ViT encoder takes an input image and produces output results.

2.1.1 The self-attention encoder module

When connecting various locations within the same series, the self-attention technique could calculate a projection of such an input data sequence [41]. The self-attention network represents the encoded structure and multi-layer perceptron (MLP) block, where each block uses the normalization layer with residual connections. A set of keys, value pairs, and a query are mapped to output using the attention function [47]. The sequence accessibility function and the appropriate key generate the weights assigned to each value, and the output is produced by adding the weighted total of the values. To learn different meanings, three learnable weight matrices M_q, M_k and M_v are created in advance, and tokens are linearly mapped to 3-D-invariant matrices, containing queries q , keys k , and values v . Finally, the attention score of each q and v is calculated using Softmax activation as shown in Eq. (2).

$$\text{SHA} = \text{Attention}(q, k, v) = \text{Soft max} \left(\frac{q \times k^t}{\sqrt{d_K}} \right) \times v \quad (2)$$

The proposed model concurrently attends data from multiple representation subsets located at different locations via multi-headed attention. The computation carried out by the encoder's multi-head self-attention for q, k and v is calculated through the concatenation of each head as follows.

$$\text{MHSA} = \text{Concat}(\text{SHA}_1, \text{SHA}_2, \dots, \text{SHA}_h) M^0 \quad (3)$$

where M is the parametric matrix, h is the total number of heads, and M^0 is the parametric matrix.

$$\text{SHA}_j = \text{Attention}(q M_j^q, k M_j^k, v M_j^v) \quad (4)$$

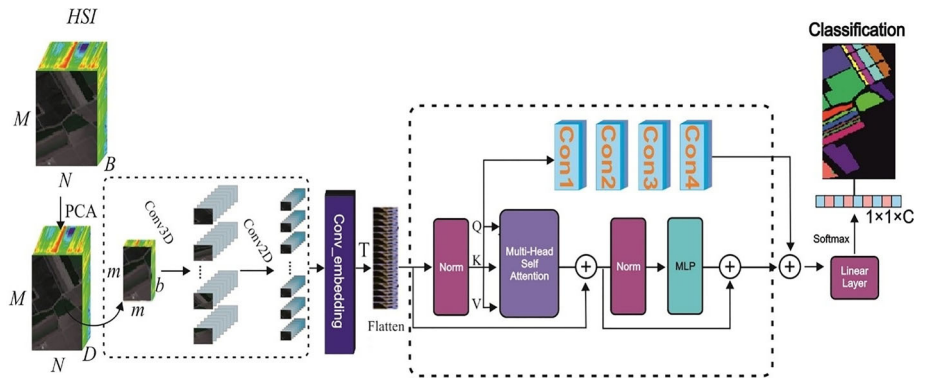
The projection of the parametric matrix is defined as follows.

$$M_j^q \in R^{D_m * d_K}, M_j^k \in R^{D_m * d_K}, M_j^v \in R^{D_m * d_K} \text{ and } M_j^0 \in R^{\text{SHA}_j * d_m}.$$

Once the weight matrix has been learned, it is fed into the MLP block. Two interconnected layers make up the MLP. The nonlinear activation function, the Gaussian error linear unit (GELU), lies between these two layers. Adding an LN after the MLP layer prevents gradient inflating, mitigates vanishing gradient issues, and expedites training. The stacked structure of layers is identical in the model. For instance, let $F \in R^{m \times D}$ be the token features with dimension D and length m . Mathematically, each block can be defined as follows.

$$B = \sigma(\text{FP}), \tilde{B} = s(B), A = \tilde{B}Q \quad (5)$$

Fig. 1 The architecture of the proposed HFTNet



$$b_0 = [X_{\text{class}} : X_p^1 E : X_p^2 \dots : X_p^N] + E_{\text{POS}},$$

$$E \in R^{(p^2 * C) * D}, E_{\text{POS}} \in R^{(N+1) * D} \tag{6}$$

$$b_l^m = \text{MSA}(\text{LN}(b_{l-1})) + b_{l-1}, l = 1, 2, 3 \dots L \tag{7}$$

$$b_l = \text{MLP}(\text{LN}(b_l^m)) + b_l^m, l = 1, 2, 3 \dots L \tag{8}$$

$$y = \text{LN}(b_l^0) \tag{9}$$

F and P stand for the dimensions of the channel’s linear projections, σ stands for an activation function, and s stands for identity mapping.

2.2 Proposed convolutional and transformer block

The FEATURES extracted from the 2D convolutional block generate 2D tokens using Eq. (10). Afterward, tokens are used for input to the dual block. Finally, k , q and v vectors are generated by flattening 2D features y , using Eq. (11).

$$\text{Tokens} = \text{MaxPool}(\text{ReLU}(\text{Conv}2d(y))). \tag{10}$$

$$y_{qkv} = \text{Linear}(\text{Flatten}(y)). \tag{11}$$

The query vector q is split into two parts $q_a \in R^{N \times C/2}$ and $q_b \in R^{N \times C/2}$. The vector q_a is passed to the transformer block and q_b to the convolutional block. By doing this, the computation cost in the transformer block was reduced half due to reduced channel size. The convolution block contains several convolution layers of kernel 3×3 with step size 1 and padding zero shown in Fig. 2. Each convolution is followed by ReLU activation and Local Response Normalization(LRN).

2.2.1 Local response normalization (LRN)

The LRN is a contrast enhancement process for feature maps and reduce the saturation problem of deep CNN. We

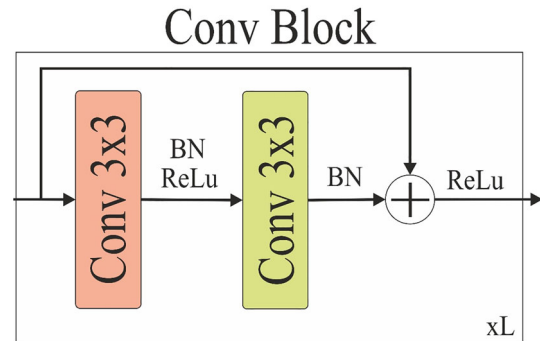


Fig. 2 The architecture of the convolution block

have used RELU activation function in the convolution block that improves neurons learning capability even on small samples. The learning activity of $x_{x,y}^i$ neurons can be evaluated at a place (x,y) through j , for the generalization of the resources. The LRN can be calculated using the formula as shown below.

$$\text{LRN}_{x,y}^i = \text{Ne}_{(x,y)}^i / \left(t + \text{Ne}_{\sum_{j=\max(0, j, n/2)}^{\min(N, 1, i+n/2)}} (\text{Ne}_{(x,y)}^i)^2 \right)^\beta \tag{12}$$

where N = Total numbers of channel and t, x, n, β = hyper-parameters. Before passing q_a to the model, it is reshaped using Eq. (13) to match the dimension with the convolution block. The q_b is reshaped to 2D using Eq. (14), Then fed to the convolution block.

$$\text{Attention}(y) = \text{Reshape} \left(\text{Soft max} \left(\frac{q_a \times k^t}{\sqrt{d_K}} \right) \times v \right). \tag{13}$$

$$\text{Conv}(y) = \text{BatchNormalization}(\text{Conv}2D(q_b)). \tag{14}$$

Finally, global features and local features are obtained through the transformer and convolution block, and these features are concatenated to form a pool of features vector as shown in Eq. (15).

$$F = \text{concat}(\text{Attention}(y), \text{Conv}(y)). \tag{15}$$

In the classical ViT, the query is directly passed to the MHA to attain the global correlation of the features. Due to this, computation costs are high. The proposed HFTNet divides the query vector into two parts to reduce the computation resources. The local and global correlation of the spatial and spectral features is achieved through convolutional and MHA blocks. Further, enhanced features are obtained by fusing the features acquired via convolution and transformer modules. The working of the conventional transformer and proposed dual block transformer is shown in Fig. 3.

The feature vector F and the feature set y is passed to the Softmax function that convert logits into probabilities [48]. The land cover class is determined by setting the value of $k = \{9, 16, 16, 13\}$ for the UP, IP, SV and KSC datasets respectively and labeling is performed using variable L . A bias value w_0y_0 included in each iteration to classify the land covers. The probabilities of class is calculated using Eq. (16).

$$P(y = L | F^{(j)}) = \frac{e^{F^{(j)}}}{\sum_{L=0}^k e_k^{F^{(j)}}} \tag{16}$$

where $F = w_0y_0 + w_1y_1 + w_2y_2 + \dots + w_ky_k$.

The algorithm of the proposed method is shown below.

Algorithm 1: Proposed HFTNet Method

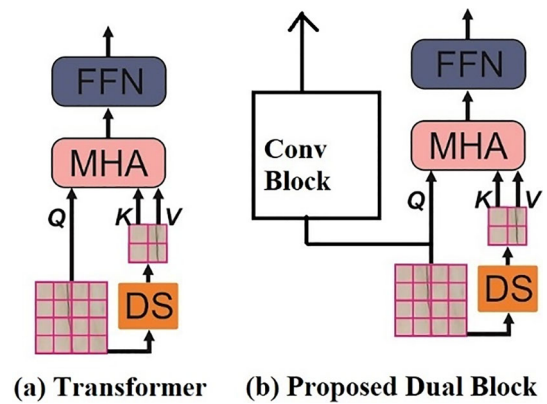


Fig. 3 Working illustration of conventional and proposed dual block transformer

(UP), Indian Pines (IP), Salinas Valley (SV) and Kennedy Space Center (KSC). The first UP dataset was captured using Reflective Optics System Imaging Spectrometer (ROSIS) sensors. It has 115 continuous spectral bands with a spatial resolution of 1.3 m per pixel (mpp) with height and width of 610 and 340, respectively. In the experiment, 103 bands are used after removing the 12 noisy bands. The nine land covers contain 42,776 pixels labeled into nine categories. The

```

INPUT: Hyperspectral image  $I \in \mathbb{R}^{M \times N \times B}$  and ground truth label  $X \in \mathbb{R}^{M \times N}$ 
OUTPUT: Classified label of the test dataset
1. Obtained  $I_{PCA}$  after performing PCA on hypercube and set  $D=30$ 
2. Create patch of the dataset and randomly divide them into training and validation sets.
3. for  $i=1$  to 100 do
  (a) Extract spectral and spatial features through 3D and 2D convolution layer
  (b) Perform 2D convolution embedding and flatten them to 1D vector.
  (c) Generate  $Q, K, V$  from 1D vector and split the  $Q$  vector to pass in convolution block and the transformer block
  (d) concatenate the semantic global and local features obtained from two module
  (e) Pass the features pool to linear layer and apply the Softmax function to predict labels.
  End
4. Use test data to classify objects using trained model.
    
```

3 Result analysis

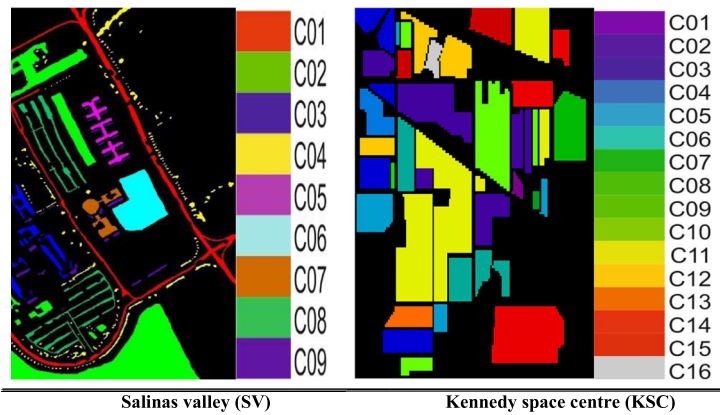
3.1 Dataset description

In the proposed study, we have implemented HFTNet on four benchmark datasets, including the University of Pavia

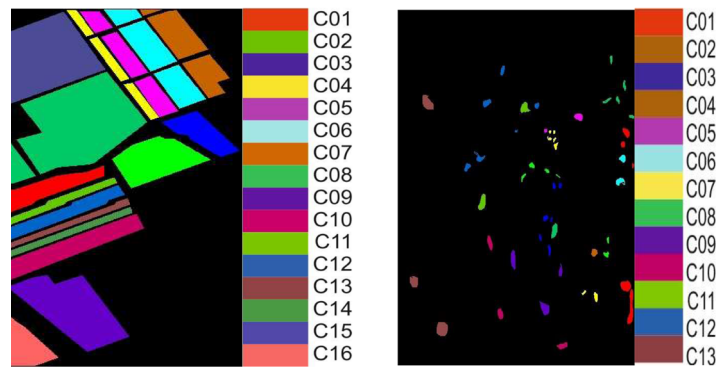
second IP dataset was collected from the Indian Pines test site in North-western Indiana by AVIRIS sensors. The land cover contains 16 types of objects with a spatial resolution of 20 mpp with a size of 145×145 pixels. The 20 water absorption bands (104–108, 150–163, and 220) are removed, and 200 bands are used for the experiment.

Table 1 Details of the sample in each land cover with their ground truth and color map

University of Pavia (UP)			Indian Pines (IP)		
Id	Class	Samples	Id	Class	Samples
1	Asphalt	6631	1	Alfalfa	46
2	Meadows	18649	2	Corn-notill	1428
3	Gravel	2099	3	Corn-mintill	830
4	Trees	3064	4	Corn	237
5	Metal sheets	1345	5	Grass-pasture	483
6	Bare soil	5029	6	Grass-trees	730
7	Bitumen	1330	7	Grass-pasture-mowed	28
8	Bricks	3682	8	Hay-windrowed	478
9	Shadows	947	9	Oats	20
	Total	42766	10	Soybean-notill	972
			11	Soybean-mintill	2455
			12	Soybean-clean	593
			13	Wheat	205
			14	Woods	1265
			15	Buildings-Grass-Trees-Drives	386
			16	Stone-Steel-Towers	93
				Total	10294



Salinas valley (SV)			Kennedy space centre (KSC)		
Id	Class	Samples	Id	Class	Samples
1	Brocoli_green_weeds_1	2009	1	Scrub	761
2	Brocoli_green_weeds_2	3726	2	Willow swamp	243
3	Fallow	1976	3	CP Hammock	256
4	Fallow_rough_plow	1394	4	Slash Pine	252
5	Fallow_smooth	2678	5	Oak/Broadleaf	161
6	Stubble	3959	6	Hardwood	229
7	Celery	3579	7	Swamp	105
8	Grapes_untrained	11271	8	Graminoid marsh	431
9	Soil_vinyard_develop	6203	9	Spartina marsh	520
10	Corn_senesced_green_weeds	3278	10	Cattail marsh	404
11	Lettuce_romaine_4wk	1068	11	Salt marsh	419
12	Lettuce_romaine_5wk	1927	12	Mud flats	503
13	Lettuce_romaine_6wk	916	13	Water	927
14	Lettuce_romaine_7wk	1070		Total	5211
15	Vinyard_untrained	7268			
16	Vinyard_vertical_trellis	1807			
	Total	54129			



The scene of the third SA dataset was also collected using AVIRIS sensors over Salinas Valley, California, which has 224 spectral bands with a spatial resolution of 3.7 mpp. The 20 water-absorbing bands (108–112, 154–167 and 224) are removed, and 204 bands of spatial size 512×217 pixels with 16 classes are utilized in our experiment. The last KSC dataset was captured using AVISIRIS sensor over the Kennedy Space Center (KSC), Florida. The spatial size of 512×614 pixels with a spatial resolution of 20 m is used in the experiment. After removing 48, water absorption and low signal-to-noise ratio 176 bands were adopted for the analysis. A details description of each dataset is provided in Table 1 [51, 52].

3.2 Experimental setting and performance indicators

The proposed method is implemented in Python environment on the window10 operating system (OS) with 128 GB RAM and NVIDIA Geforce TITAN X4000 with a dual GPU of 8 GB. First, bands of each dataset is reduced to 30 using PCA. After that, model was trained for 100 epochs using an Adam optimizer with an initial learning value of 0.0001 and a batch size of 64. For UP, SA, and KSC dataset samples are randomly split and 5% is used for training. Due to few samples in several class of IP dataset 10% samples are used for training.

To evaluate the quantitative performance of the model overall (OA), average accuracy (AA) and Kappa coefficient (Kc) and class-wise classification accuracy of each land cover is calculated based on the confusion matrix $[CM_{tp}]$. Where $[CM_{tp}]$ denotes the number of testing pixels whose true label is t and predicted label is p . $[CM_{tp}]$ can be defined as.

$$CM_{tp} = \sum_{k=1}^K 1(y_k = t)1(y_k^* = p) \quad (17)$$

where K = Total testing samples, y_k = True label and y_k^* = Predicted label.

The OA accuracy refers to the total number of correctly predicted samples and it is formulated by the following equation.

$$OA = \frac{1}{K} \sum_{t=1}^T CM_{tp} \quad (18)$$

The AA is used to calculate the mean accuracy of all per class and it is defined as

$$AA = \frac{1}{K} \sum_{t=1}^T \frac{CM_{tp}}{\sum_{p=1}^P CM_{tp}} \quad (19)$$

The Kc measures proportion of error caused by the ground truth map and final classification map.

$$Kc = \frac{\frac{1}{K} \sum_t CM_{tt} - \frac{1}{K^2} \left(\sum_p CM_{tp} \right) \left(\sum_p CM_{pt} \right)}{1 - \frac{1}{K^2} \left(\sum_p CM_{tp} \right) \left(\sum_p CM_{pt} \right)} \quad (20)$$

3.3 Comparative performance evaluation

To demonstrate the effectiveness of the proposed method seven classical method are selected, namely 1DCNN [32], 2DCNN [33], 3DCNN [36], HybridSN [37], SSRN [49], SSFTT [40] and MBDA [44]. For all the methods experiment is conducted according to the setting and parameters mentioned in the article. The 1DCNN consists of five weighted layers: input, convolution, max pooling, fully connected and classification layer. It contains 20, 1D convolutions kernel with an output size of 128. For the classification of LULC, a Softmax activation function was added on the top layer of the 1D CNN. Following the conventional CNN architecture, 2DCNN is equipped with three convolutional layers of size 8, 16 and 32, followed by a max-pooling layer, batch normalization and ReLU activation. The 3DCNN network consists of 3D convolutional followed by batch normalization and max-pooling layers. The size of 3D convolution blocks is 8, 16, and 32, respectively with a filter of size $3 \times 3 \times 3$. HybridSN consists of three 3D convolution layers of size 8, 16 and 32. After the 3D block, a 2D convolutional layer of size 64 was included in the model. Each 3D and 2D block contains a filter of size $3 \times 3 \times 3$ and 3×3 respectively. In SSRN, separate spectral and spatial blocks contain skip connections of 4 convolutional layers and two identity mapping. After two consecutive 3D convolutional layers, a residual link in the spatial block. The SSFTT network contains one 3D and one 2D convolution block and transformer module. At the top, a Softmax layer is added for the classification.

3.4 Quantitative results

The experimental results of the HFTNet on four datasets are demonstrated in Tables 2, 3, 4, and 5. The performance measures AA, OA and Kappa for every class in each dataset have been evaluated [50]. We can notice in Table 2 that 1DCNN performs poorly in all the classes but slightly improved performance results in 2DCNN. However, a few classes' performances could be more optimal due to missing spectral information. The 3DCNN improves further performance, but the computation cost is high. The HybridSN method used 3DCNN and 2DCNN to extract spectral and spatial features. The Metal Sheet class accuracy 99.62% is highest by HybridSN. The SSRN method exploits spatial and spectral

Table 2 Performance comparison on UP dataset

Id.	1DCNN [32]	2DCNN [33]	3DCNN[36]	HybridSN [37]	SSRN [49]	SSFTT [40]	MBDA[44]	HFTNet
1	92.23	97.67	96.54	98.82	95.64	99.33	99.82	98.78
2	97.63	98.52	98.84	99.17	98.28	100	100	100
3	65.23	85.36	88.92	90.57	87.85	98.51	99.53	98.62
4	94.65	96.82	94.68	95.48	93.61	97.34	99.67	99.87
5	98.24	99.42	99.52	99.62	99.32	98.54	98.84	99.12
6	86.21	98.32	99.12	98.54	98.92	99.02	99.21	99.58
7	87.54	93.74	91.53	88.95	87.67	97.56	98.67	99.82
8	83.64	95.64	92.35	91.26	90.26	98.72	99.17	98.85
9	89.58	88.68	96.36	95.72	93.65	97.58	97.15	99.46
AA	82.33	94.91	95.32	95.35	93.91	98.51	99.12	99.34
OA	84.52	95.63	96.98	96.42	95.82	98.98	99.25	99.56
Kappa	83.48	95.14	94.25	95.38	94.23	98.27	99.18	99.42

Bold indicates the highest accuracy obtained in that class

Table 3 Performance comparison on IP Dataset

Id.	1DCNN [32]	2DCNN [33]	3DCNN [36]	HybridSN [37]	SSRN [49]	SSFTT [40]	MBDA[44]	HFTNet
1	47.24	64.85	82.46	85.82	88.45	95.42	94.15	97.85
2	75.56	80.16	87.52	95.48	97.83	96.74	98.45	96.43
3	68.15	84.62	92.25	97.57	95.26	96.84	96.14	96.78
4	65.42	76.87	85.12	96.13	94.74	95.98	98.24	99.15
5	80.23	82.14	75.58	94.62	92.94	96.36	98.86	99.28
6	85.84	89.52	86.12	99.06	98.53	97.94	96.32	98.58
7	78.86	85.38	98.84	94.72	98.24	99.12	100	100
8	95.36	100	93.46	94.85	100	100	99.82	100
9	48.28	58.04	62.12	88.28	84.52	87.96	83.86	90.63
10	76.53	85.48	79.48	88.58	97.38	97.65	98.74	99.18
11	87.78	92.68	95.72	97.76	98.72	98.06	99.14	97.89
12	78.52	67.92	97.62	96.26	94.32	95.15	96.05	97.17
13	82.76	85.94	97.34	95.32	98.92	98.98	99.12	100
14	91.38	95.86	97.82	99.16	99.23	100	99.18	99.05
15	48.85	89.48	87.42	90.86	95.27	96.28	98.19	99.12
16	85.46	87.68	92.84	97.32	96.63	95.28	93.87	96.10
AA	74.76	82.91	88.23	94.49	95.69	96.74	96.92	97.95
OA	76.23	83.48	92.14	95.74	96.14	96.78	97.52	98.64
Kapa	72.82	79.67	88.73	94.86	95.12	96.52	95.98	97.36

Bold indicates the highest accuracy obtained in that class

features using a 3D convolution layer with residual attention. This model classification accuracy is higher compared to HybridSN model for several classes. The ViT-based model SSFTT outperforms Asphalt and Meadows.

However, other classes' performances need further improvement. The MBDA method computation cost is high due to multiscale 3D-CNN layers, but classification accuracy improvement can be seen in Meadows, Gravel and

Bricks class. The proposed method performance is highest for Meadows, Tress, Bare-soil, Bitumen, and Shadows classes with less computation cost due to the use of only one 3D-CNN layer and Dual block vision transformer. Similarly, in Table 3, we can notice HFTNet, SSRN and MBDA performance is 100% for Hay-Windrowed class. For small sample classes like Alfalfa, Grass-Pasture-Mowed and Oats proposed model achieved the highest accuracy of 97.85%, 100%

Table 4 Performance comparison on SV Dataset

Id.	1DCNN [32]	2DCNN [33]	3DCNN [36]	HybridSN [37]	SSRN [49]	SSFTT [40]	MBDA[44]	HFTNet
1	96.42	97.42	97.12	99.42	100	100	98.65	100
2	97.24	98.15	99.15	99.85	99.95	100	100	100
3	98.32	98.87	99.54	97.84	100	99.26	99.34	100
4	99.21	96.14	99.25	97.52	96.82	95.92	97.18	99.58
5	97.58	99.12	100	100	98.54	98.24	99.23	99.98
6	99.26	99.28	100	100	100	99.86	100	100
7	98.05	98.88	99.58	97.32	98.65	99.53	97.38	99.82
8	87.56	84.35	94.72	91.82	95.42	96.16	100	100
9	99.54	99.65	98.14	96.74	97.08	99.82	97.18	100
10	97.86	93.85	92.89	90.82	98.74	99.78	96.85	98.92
11	97.54	95.74	96.74	94.21	99.42	96.57	99.96	99.98
12	99.46	98.96	99.62	97.85	98.35	98.64	100	100
13	98.15	75.46	97.92	87.98	98.27	98.87	98.24	99.25
14	99.23	96.52	98.34	99.25	98.68	99.92	99.94	99.96
15	88.32	89.75	95.68	96.54	97.86	97.15	98.31	99.14
16	87.48	98.36	100	88.76	94.85	100	100	98.54
AA	96.33	95.03	98.04	95.99	98.29	98.73	98.89	99.70
OA	97.28	96.86	98.25	97.16	99.15	98.87	99.14	99.85
Kappa	95.45	94.78	98.14	96.45	99.87	98.98	99.02	99.83

Bold indicates the highest accuracy obtained in that class

Table 5 Performance comparison on KSC dataset

Id.	1DCNN [32]	2DCNN [33]	3DCNN [36]	HybridSN [37]	SSRN [49]	SSFTT [40]	MBDA[44]	HFTNet
1	75.54	63.34	80.76	68.18	82.46	87.52	64.25	85.27
2	45.86	66.83	57.49	61.43	72.54	78.63	88.76	85.64
3	42.56	35.38	53.24	65.52	78.21	77.81	91.24	90.52
4	53.35	37.84	44.67	48.24	56.83	55.27	72.28	80.62
5	63.24	40.64	65.94	70.16	93.58	87.78	88.67	93.25
6	45.82	42.45	48.58	42.43	57.86	68.46	67.89	75.48
7	30.48	32.94	40.26	45.34	42.32	48.35	51.48	71.34
8	46.53	47.37	56.38	67.82	72.43	65.28	66.78	74.94
9	52.67	68.72	46.86	57.28	54.85	58.92	64.96	88.36
10	48.14	42.85	41.58	45.84	43.76	48.84	46.43	68.74
11	51.34	49.86	58.93	63.76	68.78	78.65	88.75	91.82
12	54.53	65.67	66.37	62.94	72.13	81.52	76.38	94.76
13	72.72	42.92	68.92	65.87	75.52	82.42	95.57	95.10
AA	52.53	48.99	56.14	58.83	66.97	70.73	74.03	84.23
OA	54.34	50.24	57.83	62.17	68.58	72.36	75.54	85.46
Kappa	53.48	49.84	55.98	60.28	67.92	71.13	74.62	84.16

Bold indicates the highest accuracy obtained in that class

Fig. 4 Illustration of patch size on performance for UP, IP, SA and KSC dataset is shown in **a–d** respectively

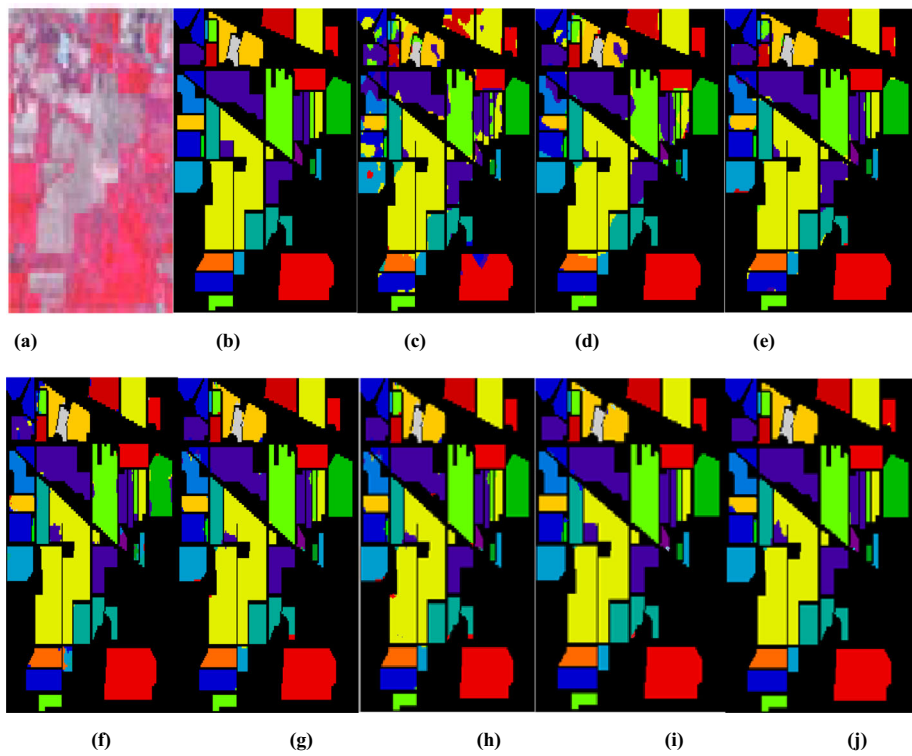
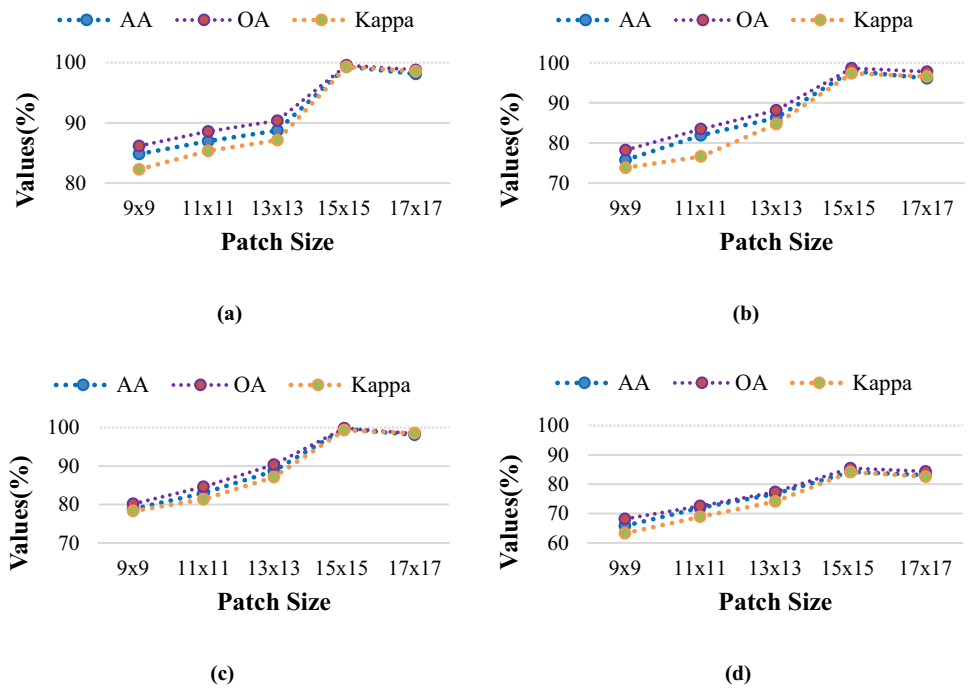
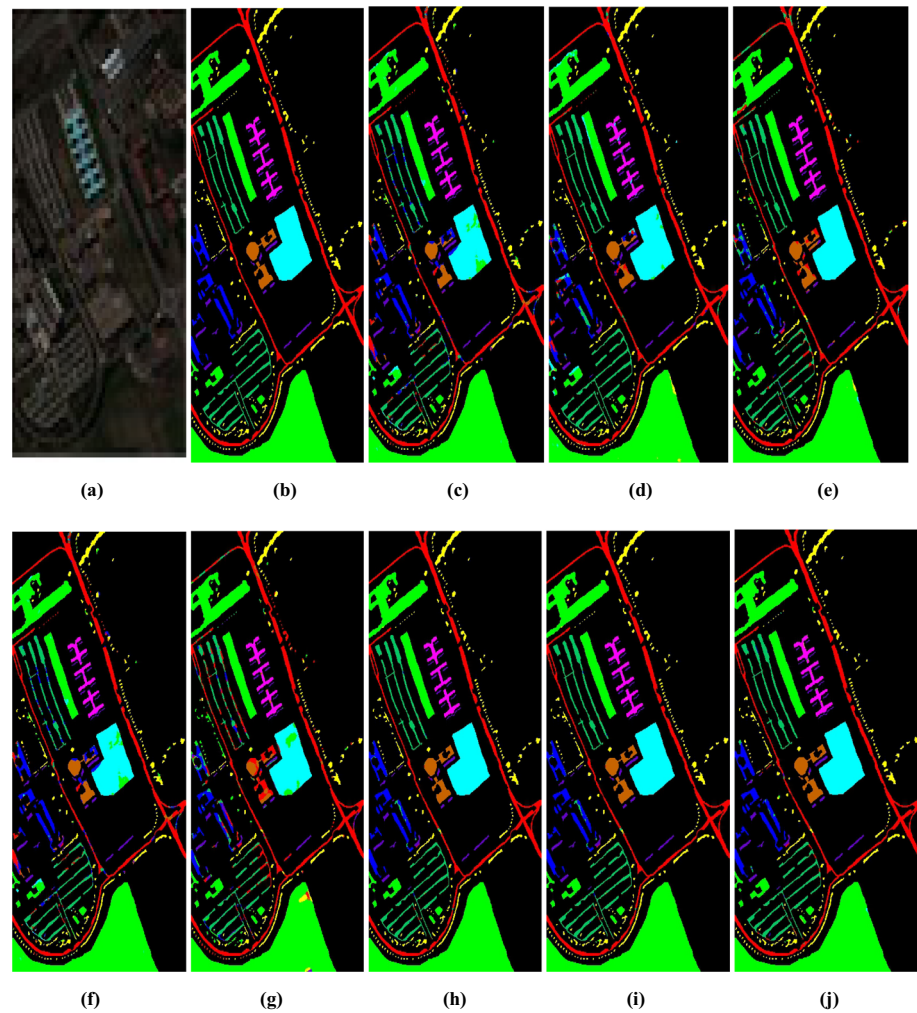


Fig. 5 Classification map visualization of different methods on IP dataset. **a** False color image **b** Ground truth map **c** 1DCNN **d** 2DCNN **e** 3DCNN **f** HybridSN **g** SSRN **h** SSFTT **i** MBDA and **j** HFTNet

Fig. 6 Classification map visualization of different methods on UP dataset. **a** False color image **b** Ground truth map **c** 1DCNN **d** 2DCNN **e** 3DCNN **(f)** HybridSN **g** SSRN **h** SSFTT **i** MBDA and **j** HFTNet



and 90.18%, respectively. In addition, HFTNet obtained very close performance compared to other methods.

In Table 4, SSFTT, MBDA and HFTNet achieved identical performance due to the large sample in each class. In some classes, HybridSN obtained the highest classification accuracy. For the KSC dataset, out of 13 classes, the proposed method achieved the highest classification accuracy in 10 classes.

The SSRN method obtained the highest performance in one class, and the MBDA method achieved the highest in three classes, as shown in Table 5. In summary, the proposed method works well on a small sample for other classes having large sample sizes, and HFTNet achieved identical performance. This confirms that adding a dual block transformer enhanced the feature selection process and improved the classification accuracy with less computation cost.

3.5 Effect of patch size on performance

In the proposed study, we conducted experiment on 9×9 , 11×11 , 13×13 , 15×15 and 17×17 patch size. For smaller patch, 9×9 model performance is poor in all the datasets. However, classification performance improved as the patch size increases. Maximum value of AA, OA and kappa was obtained with the patch size of 15×15 . Further, increasing patch size reduces the classification accuracy as shown in Fig. 4.

3.6 Visual results

The visualization map of several methods is shown in Figs. 5 and 6. In Fig. 5, we can notice classification map of 1DCNN, 2DCNN, 3DCNN and SSRN is poor for the Meadows and

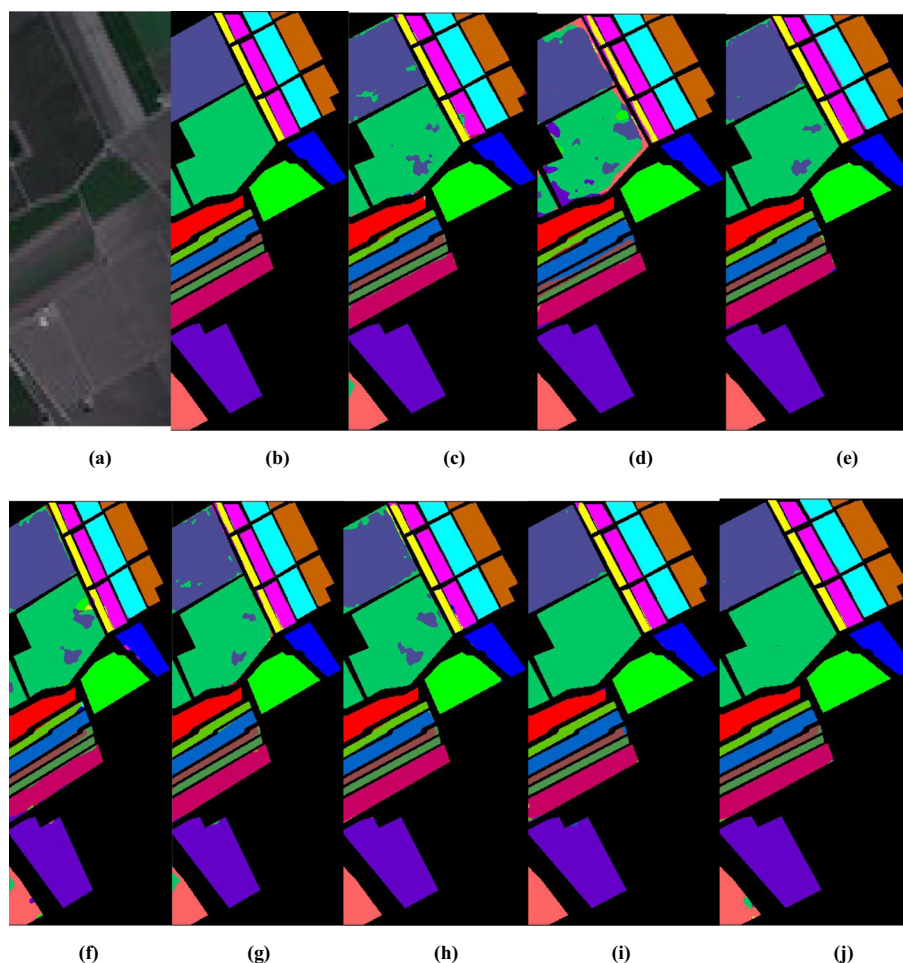


Fig. 7 Classification map visualization of different methods on SV dataset. **a** False color image **b** Ground truth map **c** 1DCNN **d** 2DCNN **e** 3DCNN **f** HybridSN **g** SSRN **h** SSFTT **i** MBDA and **j** HFTNet

Bitumen class in UP dataset. The 1DCNN and 2DCNN contains vast noise due to this objects are not classified accurately in all the datasets. Whereas, SSFTT, MBDA and proposed HFTNet is very close to ground truth map. For Metal Sheets class HybridSN and HFTNet achieved similar visualization map. On IP dataset, again the visual performance of 1DCNN, 2DCNN and 3DCNN is inferior in several classes, as shown in Fig. 7. The Grass-Free land cover class visualization map of 3DCNN, SSRN, MBDA and HFTNet is similar to the ground truth map. Since these methods can suppress the noise. Furthermore, for Alfalfa, Grass-Pasture-Mowed and Oats class, the HFTNet visualization map is closer to ground truth than other methods.

In Fig. 8, we can see the visual representation for classes one, two, three and four of the proposed method is very similar to the ground truth map. Due to the semantic global and

local spatial and spectral features. However, for class five 3DCNN and HybridSN obtained slightly better maps. Again, for classes, six and eight MBDA and HFTNet achieved the best visualization map. For other classes, the proposed method classification map is very close to the ground truth. In the KSC dataset visualization map of class one, HFTNet is much better than other methods. For classes two and three, MBDA achieved the best visual map. The remaining class classification map of the proposed method is close to the ground truth map. In short, the visualization map of the HFTNet on UP, IP, SA and KSC datasets is much better than other methods in several classes due to the improved spatial and spectral features obtained through convolutional and transformer blocks.

Fig. 8 Classification map visualization of different methods on KSC dataset. **a** False color image **b** Ground truth map **c** 1DCNN **d** 2DCNN **e** 3DCNN **f** HybridSN **g** SSRN **h** SSFTT **i** MBDA and **j** HFTNet

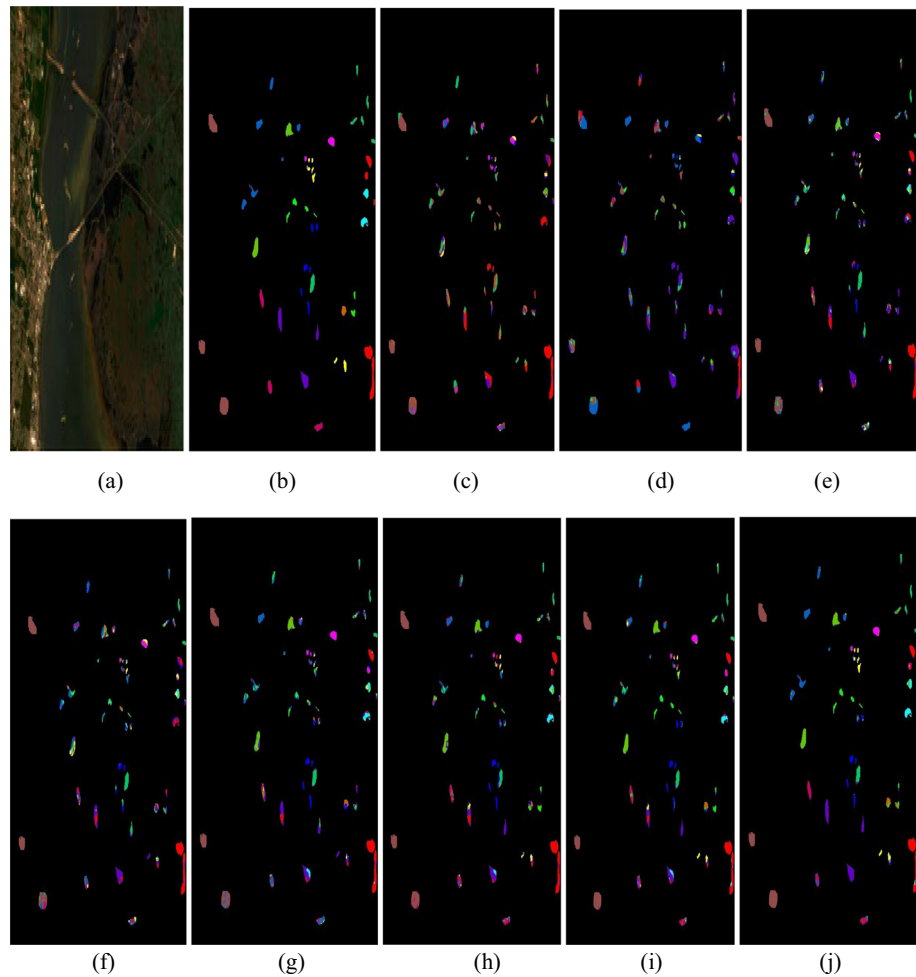


Table 6 Training (m) and test time(s) of the HFTNet on four datasets

Methods	IP		UP		SV		KSC	
	Train	Test	Train	Test	Train	Test	Train	Test
1DCNN [32]	3.26	1.57	7.12	3.02	8.46	4.23	2.54	1.02
2DCNN [33]	7.18	3.24	11.04	7.29	12.37	9.54	4.32	1.21
3DCNN [36]	8.22	3.48	12.13	8.06	14.26	10.52	5.37	2.08
HybridSN [37]	4.45	2.07	9.45	5.51	10.26	7.47	2.39	1.53
SSRN [49]	3.26	2.12	6.51	4.23	11.34	8.26	4.16	1.53
SSFTT [40]	2.98	1.96	4.17	2.97	8.18	4.67	2.35	1.19
MBDA [44]	5.63	3.10	8.97	5.12	12.54	6.31	4.87	2.15
HFTNet	2.46	1.41	3.96	2.53	7.32	4.16	2.01	1.03

3.7 The training and validation time comparison

We investigated the training and test time on four datasets with the same experimental settings. As we can see in Table 6,

the training and validation times of the 3DCNN [36] and MBDA [44] are relatively high. However, 1DCNN [32] and 2DCNN [33] take less computation time, and the SSFTT

takes the least training and test time [40] and HFTNet. However, the training and test time of SSFTT is high compared to HFTNet. In the SSFTT, the query vector is directly passed to the MHA, but in HFTNet, we divided the query into two parts and passed it to the convolution and transformer blocks. This confirms that HFTNet can be used for real-time hyperspectral image processing.

3.8 Industrial applications of the proposed method

HFTNet can be used for automated quality control in the manufacturing industry. It is perfect for identifying minor flaws or irregularities in commodities, including electronics to automobile parts, because it can analyze spatial and spectral properties in-depth. Further, it can be used for early cancer and several severe disease detection. The high-dimension features extracted by the model can help with anomaly detection, early diagnosis, and predictive analytics for patient care. In addition, HFTNet can analyze satellite and aerial imagery in agriculture to track crop health, forecast yields, and identify plant illnesses because of its ability to comprehend the images' local and global features. The proposed approach can be used to analyze geographic and environmental data for environmental applications. This involves maintaining track of land use changes, monitoring deforestation, and evaluating the condition of natural ecosystems. Other applications can be security and surveillance. The sophisticated feature extraction can improve surveillance systems in security applications. It can also be used for facial recognition and crowd analysis to detect security risks.

4 Conclusion

The hyperspectral image (HSI) contains rich spatial and spectral information sets. The traditional CNN based improved the HSI classification performance but lacked semantic global and local features. In addition, computational cost significantly improves due to many trainable parameters. In the proposed study, an HFTNet based on convolution and transformer is proposed that extracts local features from the convolution block and global semantic features from the transformer block. Finally, the features of both blocks are combined, and classification is performed using the Softmax layer. The computational resources are reduced by dividing the query into two parts and passing them through two modules.

Further, the quantitative and visual performances obtained on four datasets are much better than state-of-the-art methods. The effectiveness of the proposed method is tested on four datasets and achieved an accuracy of 99.34% (UP), 97.95% (IP), 99.70% (SV), and 84.23% (KSC). The computation cost of the proposed model is less due to the reduction

of the query in the transformer. The high performance of the proposed methods can be used in several industrial applications.

The computational resource requirements of the HFTNet still need reduction as it involves 3D convolution layers and transformer modules. In addition, the model performance has been evaluated on the open-access datasets, and its effectiveness depends on the quality and volume of the data. The network's performance must be evaluated when data is scarce, noisy, or poorly quality. The sophisticated architecture of HFTNet is beneficial for feature extraction, but it can also increase the risk of overfitting, especially when dealing with limited or particular datasets. Further, the model needs to be tested on the real-time diverse datasets.

In the future study, we will optimize the algorithm to reduce computational requirements, making HFTNet more accessible and efficient for various applications. In addition, model architecture can be refined to improve its ability to handle diverse and limited datasets effectively through advanced data augmentation techniques or transfer learning. Further, the method can be directed toward enhancing the real-time processing capabilities of HFTNet, making it more suitable for applications in dynamic environments. We can utilize the potential to integrate HFTNet with other emerging technologies, such as edge computing and IoT devices, to expand its applicability in real-world scenarios.

Author contributions DPY: Conceptualization, Data curation, Methodology, Writing—review & editing, DK: Validation, Formal analysis, : Project administration, ASJ :Visualization, Investigation. BK: Data curation, AK, Writing—original draft.

Funding No funding available.

Data availability The dataset used in the study can be downloaded from the following open access repository:https://www.ehu.es/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes

Declarations

Conflict of interest All authors declare that he has no conflict of interest. We certify that the submission is original work and that neither the submitted materials nor portions have been published previously or are under consideration for publication elsewhere.

Ethical approval This article contains no studies with human participants or animals performed by authors.

References

1. Lin, K., Guo, Y., Liu, Y., Zhang, X., Xiao, S., Gao, G., Wu, G.: Outdoor detection of the pollution degree of insulating materials based on hyperspectral model transfer. *Measurement* **214**, 112805 (2023)
2. Wang, B., Ren, M., Xia, C., Li, Q., Dong, M., Zhang, C., Guo, C., Liu, W., Pischler, O.: Evaluation of insulator aging status based on

- multispectral imaging optimized by hyperspectral analysis. *Measurement* **205**, 112058 (2022)
3. Castillo, F., Arias, L., Garcés, H.O.: Estimation of temperature, local and global radiation of flames, using retrieved hyperspectral imaging. *Measurement* **208**, 112459 (2023)
 4. Mou, L., Ghamisi, P., Zhu, X.X.: Deep recurrent neural networks for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **55**(7), 3639–3655 (2017). <https://doi.org/10.1109/TGRS.2016.2636241>
 5. Plaza, A., Plaza, J., Martin, G.: Incorporation of spatial constraints into spectral mixture analysis of remotely sensed hyperspectral data. In: *Proceedings of the IEEE International Workshop Machine Learning Signal Processing*, pp. 1–6 (2009)
 6. Jia, S., Lin, Z., Deng, B., Zhu, J., Li, Q.: Cascade superpixel regularized Gabor feature fusion for hyperspectral image classification. *IEEE Trans. Neural Netw. Learn. Syst.* **31**(5), 1638–1652 (2019)
 7. Liu, C., Li, J., He, L., Plaza, A.J., Li, S., Li, B.: Naive Gabor networks for hyperspectral image classification. *IEEE Trans. Neural Netw. Learn. Syst.* **32**(1), 376–390 (2021)
 8. Zhang, Y., Li, W., Zhang, M., Qu, Y., Tao, R., Qi, H.: Topological structure and semantic information transfer network for cross-scene hyperspectral image classification. *IEEE Trans. Neural Netw. Learn. Syst.* (2021). <https://doi.org/10.1109/TNNLS.2021.3109872>
 9. Yang, S., Feng, Z., Wang, M., Zhang, K.: Self-paced learning-based probability subspace projection for hyperspectral image classification. *IEEE Trans. Neural Netw. Learn. Syst.* **30**(2), 630–635 (2019)
 10. Gong, Z., Zhong, P., Hu, W.: Statistical loss and analysis for deep learning in hyperspectral image classification. *IEEE Trans. Neural Netw. Learn. Syst.* **32**(1), 322–333 (2021)
 11. Zhang, B., Sun, X., Gao, L., Yang, L.: Endmember extraction of hyperspectral remote sensing images based on the ant colony optimization (ACO) algorithm. *IEEE Trans. Geosci. Remote Sens.* **49**(7), 2635–2646 (2011)
 12. Zhang, B.: Advancement of hyperspectral image processing and information extraction. *J. Remote Sens.* **20**(5), 1062–1090 (2016)
 13. Ma, L., Crawford, M.M., Tian, J.: Local manifold learning-based k -nearest-neighbor for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **48**(11), 4099–4109 (2010)
 14. Cariou, C., Chehdi, K.: A new k -nearest neighbor density-based clustering method and its application to hyperspectral images. In: *Proceedings of the IEEE International Geoscience Remote Sensing Symposium (IGARSS)*, pp. 6161–6164 (2016)
 15. SahIn, Y. E., Arisoy, S., Kayabol, K.: Anomaly detection with Bayesian Gauss background model in hyperspectral images. In: *Proceedings of the 26th Signal Processing Communication Applications Conference (SIU)*, pp. 1–4 (2018)
 16. Haut, Y. J., Paoletti, M., Paz-Gallardo, A., Plaza, J., Plaza, A.: Cloud implementation of logistic regression for hyperspectral image classification. In: *Proceedings of the 17th International Conference on Computational Mathematical Methods Science Engineering (CMMSE)*, vol. 3, pp. 1063–2321. Costa Ballena, Cádiz, Spain (2017)
 17. Li, J., Bioucas-Dias, J., Plaza, A.: Spectral–spatial hyperspectral image segmentation using subspace multinomial logistic regression and Markov random fields. *IEEE Trans. Geosci. Remote Sens.* **50**(3), 809–823 (2012)
 18. Melgani, F., Bruzzone, L.: Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. Geosci. Remote Sens.* **42**(8), 1778–1790 (2004)
 19. Ye, Q., Huang, P., Zhang, Z., Zheng, Y., Fu, L., Yang, W.: Multiview learning with robust double-sided twin SVM. *IEEE Trans. Cybern.* (2021). <https://doi.org/10.1109/TCYB.2021.3088519>
 20. Ye, Q., et al.: L1-norm distance minimization-based fast robust twin support vector k -plane clustering. *IEEE Trans. Neural Netw. Learn. Syst.* **29**(9), 4494–4503 (2017)
 21. Chen, Y.-N., Thaipisutikul, T., Han, C.-C., Liu, T.-J., Fan, K.-C.: Feature line embedding based on support vector machine for hyperspectral image classification. *Remote Sens.* **13**(1), 130 (2021)
 22. Hong, D., Yokoya, N., Chanussot, J., Zhu, X.: An augmented linear mixing model to address spectral variability for hyperspectral unmixing. *IEEE Trans. Image Process.* **28**(4), 1923–1938 (2019)
 23. Shabbir, S., Ahmad, M.: *Hyperspectral Image Classification-Traditional to Deep Models: A Survey for Future Prospects*. 2021. <https://arxiv.org/abs/2101.06116>
 24. Ye, Q., Yang, J., Liu, F., Zhao, C., Ye, N., Yin, T.: L1-norm distance linear discriminant analysis based on an effective iterative algorithm. *IEEE Trans. Circuits Syst. Video Technol.* **28**(1), 114–129 (2018)
 25. Fu, L., et al.: Learning robust discriminant subspace based on joint L2, p - and L2, s -norm distance metrics. *IEEE Trans. Neural Netw. Learn. Syst.* **33**(1), 130–144 (2022)
 26. Bandos, T.V., Bruzzone, L., Camps-Valls, G.: Classification of hyperspectral images with regularized linear discriminant analysis. *IEEE Trans. Geosci. Remote Sens.* **47**(3), 862–873 (2009)
 27. Villa, A., Benediktsson, J.A., Chanussot, J., Jutten, C.: Hyperspectral image classification with independent component discriminant analysis. *IEEE Trans. Geosci. Remote Sens.* **49**(12), 4865–4876 (2011)
 28. 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.* **9**(3), 447–451 (2012)
 29. Prasad, S., Bruce, L.M.: Limitations of principal components analysis for hyperspectral target recognition. *IEEE Geosci. Remote Sens. Lett.* **5**(4), 625–629 (2008)
 30. Chen, Y., Lin, Z., Zhao, X., Wang, G., Gu, Y.: Deep learning-based classification of hyperspectral data. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **7**(6), 2094–2107 (2014)
 31. Chen, Y.S., Zhao, X., Jia, X.: Spectral–spatial classification of hyperspectral data based on deep belief network. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **8**(6), 2381–2392 (2014)
 32. Hu, W., Huang, Y., Wei, L., Zhang, F., Li, H.: Deep convolutional neural networks for hyperspectral image classification. *J. Sensors* **2015**, 1–12 (2015)
 33. Shao, W., Du, S.: Spectral–spatial feature extraction for hyperspectral image classification: a dimension reduction and deep learning approach. *IEEE Trans. Geosci. Remote Sens.* **54**(8), 4544–4554 (2016)
 34. Yue, J., Zhao, W., Mao, S., Liu, H.: Spectral–spatial classification of hyperspectral images using deep convolutional neural networks. *Remote Sens. Lett.* **6**(6), 468–477 (2015)
 35. Yang, J., Zhao, Y.-Q., Chan, J.C.-W.: Learning and transferring deep joint spectral–spatial features for hyperspectral classification. *IEEE Trans. Geosci. Remote Sens.* **55**(8), 4729–4742 (2017)
 36. Chen, Y., Jiang, H., Li, C., Jia, X., Ghamisi, P.: Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Trans. Geosci. Remote Sens.* **54**(10), 6232–6251 (2016)
 37. 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.* **17**(2), 277–281 (2020)
 38. Ahmad, M., Shabbir, S., Raza, R.A., Mazzara, M., Distefano, S., Khan, A.M.: Artifacts of different dimension reduction methods on hybrid CNN feature hierarchy for hyperspectral image classification. *Optik* **246**, 167757 (2021)
 39. Liu, R., Cai, W., Li, G., Ning, X., Jiang, Y.: Hybrid dilated convolution guided feature filtering and enhancement strategy for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **19**, 1–5 (2022). <https://doi.org/10.1109/LGRS.2021.3100407>

40. Sun, L., Zhao, G., Zheng, Y., Wu, Z.: Spectral–spatial feature tokenization transformer for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–14 (2022)
41. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. *Adv. Neural Inf. Process. Syst.* **30** (2017)
42. Vaswani, A., et al.: Attention is all you need. In: *Proceedings of the Advances in Neural Information Processing Systems*, vol. 30 (2017)
43. Huang, X., Dong, M., Li, J., Guo, X.: A 3-D-swin transformer-based hierarchical contrastive learning method for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–15 (2022). <https://doi.org/10.1109/TGRS.2022.3202036>
44. Yin, J., Qi, C., Huang, W., Chen, Q., Qu, J.: Multibranch 3D-dense attention network for hyperspectral image classification. *IEEE Access* **10**, 71886–71898 (2022)
45. Yang, L., Yang, Y., Yang, J., Zhao, N., Wu, L., Wang, L., Wang, T.: FusionNet: a convolution-transformer fusion network for hyperspectral image classification. *Remote Sens.* **14**(16), 4066 (2022)
46. Chollet, F.: Xception: deep learning with depthwise separable convolutions. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1251–1258 (2017)
47. Alayrac, J.B., Recasens, A., Schneider, R., Arandjelović, R., Ramapuram, J., De Fauw, J., Smaira, L., Dieleman, S., Zisserman, A.: Self-supervised multimodal versatile networks. *Adv. Neural. Inf. Process. Syst.* **33**, 25–37 (2020)
48. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778. Las Vegas, NV, USA (2016)
49. Zhong, Z., Li, J., Luo, Z., Chapman, M.: Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework. *IEEE Trans. Geosci. Remote Sens.* **56**(2), 847–858 (2018)
50. Yu, H., Xu, Z., Zheng, K., Hong, D., Yang, H., Song, M.: MSTNet: a multilevel spectral–spatial transformer network for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–13 (2022). <https://doi.org/10.1109/TGRS.2022.3186400>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.