Local Information Interaction Transformer for Hyperspectral and LiDAR Data Classification

Yuwen Zhang, *Student Member, IEEE*, Yishu Peng, *Member, IEEE*, Bing Tu, *Member, IEEE*, and Yaru Liu, *Student Member, IEEE*

***Abstract*—The multisource remote sensing classification task has two main challenges. 1) How to capture hyperspectral image (HSI) and light detection and ranging (LiDAR) features cooperatively to fully mine the complementary information between data. 2) How to adaptively fuse multisource features, which should not only overcome the imbalance between HSI and LiDAR data but also avoid the generation of redundant information. The local information interaction transformer (LIIT) model proposed herein can effectively address these above issues. Specifically, multibranch feature embedding is first performed to help in the fine-grained serialization of multisource features; subsequently, a local-based multisource feature interactor (L-MSFI) is designed to explore HSI and LiDAR features together. This structure provides an information transmission environment for multibranch features and further alleviates the homogenization processing mode of the self-attention process. More importantly, a multisource feature selection module (MSTSM) is developed to dynamically fuse HSI and LiDAR features to solve the problem of insufficient fusion. Experiments were carried out on three multisource remote-sensing classification datasets, the results of which show that LIIT has more performance advantages than the state-of-the-art CNN and transformer methods.**

***Index Terms*—Feature fusion, local information interaction transformer (LIIT), multisource data classification, transformer.**

I. Introduction

The classification of remote-sensing images, a pixel-level classification task, is the process of recognizing unmarked areas by learning to obtain prior knowledge [1], [2], [3], [4], and applied in various practices [5], [6]. Hyperspectral (HSI) data is obtained by imaging spectrometer, which can provide a large amount of narrowband spectral information from the visible spectrum to the infrared spectrum for each pixel [7]. Due to its rich spectral information content, HSI data have more detailed surface feature description ability than other remote-sensing data, and is one of the most suitable remote sensing classification task data sources [8]. However, it is worth noting that there are often similar spectral curves with different types of ground objects in real ground objects; furthermore, the same type of ground objects can show different spectral curves due to differences in their regional distribution. Therefore, single HSI data has limitations when dealing with the task of classifying complex terrain scenes. With the development of remote-sensing imaging technology, each sensor can obtain various remote-sensing data with different physical characteristics from the same geographical space [9], [10], [11]. Light detection and ranging (LiDAR) point cloud data carries distance information between sensors and ground objects, and can be converted into its image version DSM through preprocessing. Images can describe the elevation information of the figure by the size of the gray value [12], [13], [14]. Since LiDAR data are not easily affected by the external atmospheric medium and weather, its elevation information also provides strong support for accurate surface-feature classification tasks.

To overcome the shortcomings of single HSI data derived from classification tasks, many researchers try to use integration strategies to combine the unique features of HSI and LiDAR data to develop complementary advantages and carry out collaborative feature recognition of multimodal data [15], [16], [17]. Mattia et al. [18] initially obtained the extended attribute profile (EAP) of HSI and LiDAR data from the morphology perspective. After feature concatenation and fusion, they were sent to the classifier to complete the classification. However, a simple fusion method will generate feature redundancy and a the existence of large number of dimensions after stacking will easily cause the Hughes phenomenon. In this regard, Behnood et al. [19] introduced the kernel principal component analysis (KPCA) method to reduce the dimensions of features after obtaining the extended profile of HSI and LiDAR data; and realized feature fusion with the help of orthogonal TV component analysis (OTVCA). Jia et al. [20] used superpixel-guided KPCA to preprocess HSI data. Then, the 2-D and 3-D Gabor filter is used to extract the features of LiDAR and processed HSI data, respectively, so as to obtain the identifiable multisource Gabor features with the magnitude and phase information. Zhang et al. [21] tried

Manuscript received 17 October 2022; revised 2 December 2022; accepted 21 December 2022. Date of publication 30 December 2022; date of current version 10 January 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 61977022 and Grant 62271200, in part by the Foundation of Department of Water Resources of Hunan Province under Grant XSKJ2021000-12, Grant XSKJ2021000-13, and Grant XSKJ2022068-48, in part by the Natural Science Foundation of Hunan Province under Grant 2021JJ40226, in part by the Foundation of Education Bureau of Hunan Province under Grant 21B0590, Grant 21B0595, and Grant 20B062, and in part by the Scientific Research Fund of Education Department of Hunan Province under Grant 19A200. *(Corresponding author: Yishu Peng.)*

Yuwen Zhang, Yishu Peng, and Yaru Liu are with the School of Information Science and Engineering, Hunan Institute of Science and Technology, Yueyang 414000, China (e-mail: yuwen_zhang@vip.hnist.edu.cn; lovepys@hnist.edu.cn; liuyarua@foxmail.com).

Bing Tu is with the School of Information Science and Engineering, Hunan Institute of Science and Technology, Yueyang 414000, China, and also with the Guangxi Key Laboratory of Cryptography and Information Security, Guilin University of Electronic Technology, Guilin 541000, China (e-mail: tubing@hnist.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2022.3232995

using various classifiers, i.e., SVM, RF, and KNN, to obtain preliminary HSI and LiDAR classification results, and to achieve more robust classification of multisource remote sensing data from the decision level through majority weighted voting. Although the above classical methods fuse HSI and LiDAR data from multiple perspectives, the processing of multisource data is incomplete and the information is not fully utilized.

The emergence of deep neural network (DNN) helps to extract deeper semantic information from remote-sensing data [22], [23], [24], [25]. By placing each modal data in different structural branches, data information can be more fully mined. Xu et al. [26] designed different network structures for the feature extraction of HSI and LiDAR data, and proposed a two-branch CNN model. One branch uses a 2-D and 1-D hybrid CNN structure to capture the spectral − spatial features of HSI and the other branch designs a cascade-based CNN to explore the elevation information of LiDAR data. Zhao et al. [27] applied a weight contribution mechanism to the dual-branch structure and proposed the coupled CNN model, that model not only alleviates the calculation pressure of the model, but also guides the learning process of the dual branches, strengthening the feature consistency by sharing the last two convolutions of the HSI and LiDAR branches. Hong et al. [28] applied the full connection layer model to multisource remote sensing classification tasks, and designed additional feature reconstruction structures after the encoding process of HSI and LiDAR data, with the aim of promoting feature fusion more compactly. In [29], multiscale PToP CNN was designed to obtain HSI and LiDAR features at different scales to make full use of multisource remote sensing information. However, the conventional deep learning methods have defects in dealing with the interference of spatial edge pixels. The CNN-based methods are even more due to the limitation of the fixed convolution kernel size, leading to the introduction of extra classes of pixels, which affects the training effect of the model [30], [31], [32].

Therefore, researchers start to designing a network module with attention capability [33], [34], [35]. By adaptively identifying the importance of features and giving them corresponding weight values, it can highlight important information and weaken the function of secondary information to enhance the feature recognition. The transformer is designed with an attention module as the basic framework and has made remarkable achievements in natural-language processing tasks [36], [37], [38], [39]. Alexey et al. [40] introduced a transformer into the image field for the first time by serializing images, and proposed vision transformer (ViT), that can model features at the global level both simply and effectively, and established dependencies between the sequence data. Through the introduction and improvement of the ViT model, the joint classification task of HSI and LiDAR data has been further broken through. Dong et al. [41] proposed an effective multibranch feature fusion network with self- and cross-guided attention. This method started by obtaining the weight graph of LiDAR and is used to guide the self-attention of LiDAR data and the cross-attention of HSI data, respectively. Xue et al. [42] used deep hierarchical vision transformer (DHViT) to extract sequence features of HSI and LiDAR data, and fused various modal sequences after the cross-attention module. Although the above methods effectively capture the heterogeneous features of HSI and LiDAR, they do not take the information imbalance between HSI and LiDAR data into account during the fusion process.

From the perspective of multimodal information interaction and feature screening, we propose a local information interaction transformer (LIIT) model to capture and fuse multimodal remote sensing data dynamically. Specifically, the dual-branch transformer was first designed to fully extract the sequence features of HSI and LiDAR. In this process, local based multisource feature interactor (L-MSFI) is developed to endow the global-based transformer model with a local spatial feature information interaction ability. In addition, a multisource feature selection module (MSFSM) is introduced to give weight to each modal data to realize the dynamic multimodal data filtering function and solve the imbalance problem between features. Subsequently, the fused feature is put into the convolutional transformer module to help with further training and classification. Compared to state-of-the-art methods in several open multisource remote sensing datasets, LIIT can achieve better classification performance. The main contributions of this article are as follows.

1) A local-based multisource feature interactor (L-MSFI) is designed to provide an information interaction environment for HSI and LiDAR features, avoid independent feature extraction process, and guide features to learn from each other.
2) The convolution module is added to the self-attention, which overcomes the process of the gradual homogenization of different features due to them having the same operation in the self-attention, and makes its description of features more detailed.
3) An MSFSM has been developed to solve the balance problem of HSI and LiDAR features in the fusion and reduce the generation of redundant information by dynamically filtering source components in the sequence features.

The remainder of this article is organized as follows. Section II describes the proposed LIIT method, Section III introduces the design of parameters in the LIIT method and the comparison with multisource remote sensing classification methods on the Houston, MUUFL, and Trento datasets, and Section IV concludes this article.

II. Description of the Proposed Approach

This section first gives an overall introduction to the proposed LIIT method. On this basis, the functions and importance of L-MSFI and MSFSM are analyzed and explained in detail.

A. *Overview of the Proposed Method*

The joint classification task of HSI and LiDAR data aims to make full use of their respective outstanding features, complement each other's advantages, and break through the performance bottleneck when using a single data source. The main challenges are as follows. 1) How to effectively capture the semantic features of multisource data and maximize the retention

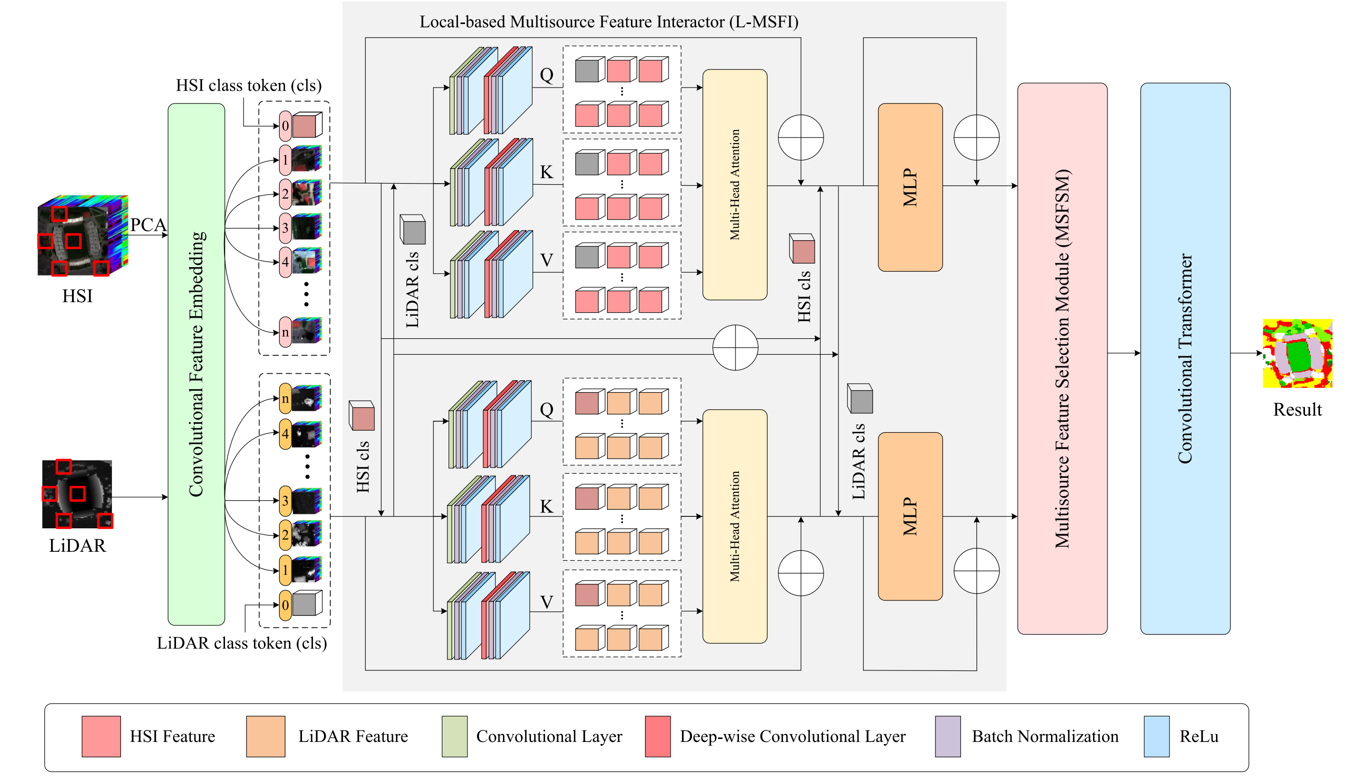

Local-based Multisource Feature Interactor (L-MSFI)
HSI class token (cls)
PCA
HSI
LiDAR
Convolutional Feature Embedding
0
1
2
3
4
n
LiDAR cls
HSI cls
Q
K
V
Multi-Head Attention
MLP
Multisource Feature Selection Module (MSFSM)
Convolutional Transformer
Result
LiDAR class token (cls)
HSI Feature
LiDAR Feature
Convolutional Layer
Deep-wise Convolutional Layer
Batch Normalization
ReLu

Fig. 1. Framework of the proposed LIIT method, where the upper branch is used to extract HSI feature, and the lower is the respective LiDAR branch to extract elevation feature.

of data information without Hughes phenomenon. 2) How to avoid information redundancy and overcome the problem of data imbalance combined with the heterogeneous features between multimodes, due to the imbalance of the importance between HSI and LiDAR data for classification.

The LIIT model is proposed for the above analysis, and Fig. 1 shows its structural framework. Specifically, the dual-branch transformer is first adopted to obtain the semantic features of HSI and LiDAR data from the level of sequence global dependency. In this process, L-MSFI is designed to mine the information of multisource features from the local level, while also forming information interactions between multimodal features to avoid the closed state of the feature extraction process of each branch. MSFSM is used to dynamically filter HSI and LiDAR features; its adaptive feature fusion method effectively avoids feature imbalance and fusion redundancy. Finally, the fusion features are further trained by the transformer and the final classification is completed.

B. Embedding for HSI and LiDAR Data

Feature embedding, which aims to perform sequence mapping of image data and establish the interdependence of sequence features from the global level is the initial step of the transformer [43], [44], [45]. For the vanilla ViT, the convolution layer with the same step size and kernel size is usually used to perform nonoverlapping blocking operations on the image, and then flatten each block to form sequence data [46], [47]. However, this process is obviously coarse-grained and there is a lack of information transfer between blocks.

For the feature embedding in the LIIT method, using a principal component analysis (PCA) algorithm reduces the dimensions of the HSI data $\mathbf{X}_H \in \mathbb{R}^{H\times W\times C}$ and preextracts its spectral features. Then, the convolution module helps conduct tokenization for HSI and LiDAR data. Specifically, the convolution process is Conv-BN-ReLU-DWConv-BN-ReLU

$$\begin{aligned}
\mathbf{X}_{\mathbf{H}}^{1} &= \mathrm{ReLU}(\mathrm{BN}(\mathrm{Conv}(\mathbf{X}_H))) \\
\mathbf{X}_{L}^{1} &= \mathrm{ReLU}(\mathrm{BN}(\mathrm{Conv}(\mathbf{X}_L))) \\
\mathbf{X}_{H}^{2} &= \mathrm{ReLU}\left(\mathrm{BN}\left(\mathrm{DWConv}\left(\mathbf{X}_{H}^{1}\right)\right)\right) \\
\mathbf{X}_{L}^{2} &= \mathrm{ReLU}\left(\mathrm{BN}\left(\mathrm{DWConv}\left(\mathbf{X}_{L}^{1}\right)\right)\right)
\end{aligned} \tag{1}$$

where multisource data are put into different convolution branches. This process is fine-grained and ensures dimensional alignment between multimodal features. After features have been serialized, add class token ($\mathbf{E}_{cls}^{H}$, $\mathbf{E}_{cls}^{L}$) for classification and position embeddings ($\mathbf{E}_{pos}^{H}$, $\mathbf{E}_{pos}^{L}$) for encoding sequence sequence to multisource data. The process is as follows:

$$\begin{aligned}
\mathbf{S}_{\mathbf{H}} &= \left[\mathbf{X}_{H}^{2}; \mathbf{E}_{cls}^{H}\right] + \mathbf{E}_{pos}^{H} \\
\mathbf{S}_{L} &= \left[\mathbf{X}_{L}^{2}; \mathbf{E}_{cls}^{L}\right] + \mathbf{E}_{pos}^{L}
\end{aligned} \tag{2}$$

where ; is the concatenation process, $\mathbf{S}_H, \mathbf{S}_L \in \mathbb{R}^{(N+1)\times D}$ is the output sequences of HSI and LiDAR data, N is the number of sequences, and D is the number of channels.

After the embedding process, the sequences will put into the transformer for feature extraction.

C. Local-Based Multisource Feature Interactor (L-MSFI)

It is worth noting that the self-attention process can help each token in the sequence to transmit information, but it will also make different tokens homogeneous because of the similar operation. Therefore, the use of a convolutional module helps feature focus on local information, thus avoiding the over-smoothing of self-attention. Specifically, the class token ($\mathbf{E}_{cls}^H, \mathbf{E}_{cls}^L$) of each modal feature is stripped first, and the rest of the features are reshaped to a spatial feature block whose size is consistent with the original input. Then, the convolution module is designed to establish local correlation for the spatial neighborhood intra of each source data, and the process is shown as follows:

$$\begin{aligned} &Q_H, K_H, V_H \\ &\quad = \boldsymbol{Conv}_{Q1}\left(\mathbf{X}_H^2\right), \boldsymbol{Conv}_{K1}\left(\mathbf{X}_H^2\right), \boldsymbol{Conv}_{V1}\left(\mathbf{X}_H^2\right) \\ &Q_L, K_L, V_L \\ &\quad = \boldsymbol{Conv}_{Q2}\left(\mathbf{X}_L^2\right), \boldsymbol{Conv}_{K2}\left(\mathbf{X}_L^2\right), \boldsymbol{Conv}_{V2}\left(\mathbf{X}_L^2\right) \end{aligned} \quad (3)$$

where $\boldsymbol{Conv}$ is represent the convolution module, which is consistent with the embedding process, and includes common convolutional layer and deep-wise convolutional layer. Among them, the common convolutional layer is used for information transfer between channel dimension features, while deep-wise convolution can relieve the pressure of model parameters.

The information interaction between multisource data and feature extraction is the key to the joint classification task, which makes the feature extraction process of each branch not isolated, and conducive to the expression of each source feature. Considering that the class token is the classification representation in each branch, global dependency can be established with the branch feature in self-attention. To this end, we flatten the HSI and LiDAR features after convolution, and concatenate each branch's class token onto another branch, which is shown in Fig. 1, and the expression is shown as follows:

$$\begin{aligned} Q_H, K_H, V_H &= \left[Q_H; \mathbf{E}_{cls}^L\right], \left[K_H; \mathbf{E}_{cls}^L\right], \left[V_H; \mathbf{E}_{cls}^L\right] \\ Q_L, K_L, V_L &= \left[Q_L; \mathbf{E}_{cls}^H\right], \left[K_L; \mathbf{E}_{cls}^H\right], \left[V_L; \mathbf{E}_{cls}^H\right]. \end{aligned} \quad (4)$$

As the attention process proceeds, the class token continuously learns the semantic information of another branch

$$\begin{aligned} \mathbf{Z}_H &= \boldsymbol{Attention}(Q_H, K_H, V_H) \\ &= \boldsymbol{Softmax}\left(\frac{Q_H {K_H}^T}{\sqrt{d_k}}\right) V_H \\ \mathbf{Z}_L &= \boldsymbol{Attention}(Q_L, K_L, V_L) \\ &= \boldsymbol{Softmax}\left(\frac{Q_L {K_L}^T}{\sqrt{d_k}}\right) V_L \end{aligned} \quad (5)$$

where $\mathbf{Z}_H$ and $\mathbf{Z}_L$ are the output of the attention for HSI and LiDAR branches. After completion, class tokens are returned to the original branch and sent to the MLP module along with the original branch features to further capture the self-source features

$$\begin{aligned} \mathbf{S}'_{\mathbf{H}} &= \boldsymbol{MLP}_H\left[\mathbf{X}'_H; \mathbf{E}_{cls}^H\right] \\ \mathbf{S}'_L &= \boldsymbol{MLP}_L\left[\mathbf{X}'_L; \mathbf{E}_{cls}^L\right] \end{aligned} \quad (6)$$

where $\mathbf{X}'_H$ and $\mathbf{X}'_L$ are the features after interactive attention for HSI and LiDAR branches. Consistent with the vanilla self-attention, the residual structure is also retained to achieve the information aggregation of the original features and the processed features.

In L-MSFI, each data source feature can conduct local-based feature learning on another source feature by exchanging class tokens, and further aggregate global features with self-attention. After the class token returns to the original branch and the subsequent feature extraction has completed, each branch feature realizes the information interaction based on the class token.

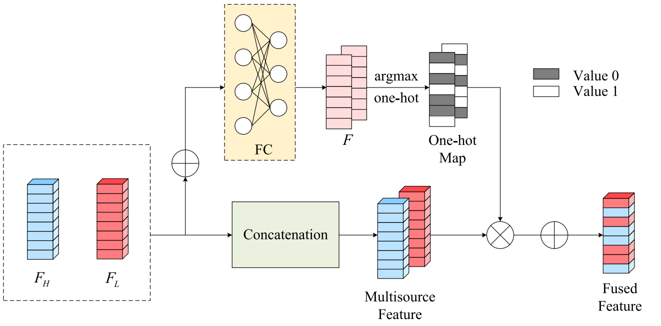


Fig. 2. Structure of MSFSM, which fuses multisource features by adaptively filtering HSI and LiDAR heterogeneous features.

D. Multisource Feature Selection Module (MSFSM)

Conventional linear fusion methods fuse HSI and LiDAR features indiscriminately, and lack the modulation of features, which introduces unnecessary redundant features. Beside, the relative importance of HSI and LiDAR data for collaborative classification is unequal. Generally speaking, HSI data have a larger information load because it occupies the main feature in the fusion feature, so adopting a simple linear fusion method limits the expression ability of the HSI feature. In this regard, we learned about the SCN [48] module, introduced it into the fusion of HSI and LiDAR sequence features, and proposed MSFSM, as is shown in Fig. 2. Specifically, for each branch feature $F_H \in \mathbb{R}^{N \times C}$ and $F_L \in \mathbb{R}^{N \times C}$ (where N represents the number of sequences and C represents the number of channels) after the joint feature extraction by L-MSFM, an fully connected (FC) layer is first used to combine the features and map them to $F \in \mathbb{R}^{N \times 2}$; the results are as follows:

$$F = \mathbf{FC}(F_H + F_L). \quad (7)$$

After F has been obtained, the probability map is obtained using the softmax function, whose probability indicates which source component the feature most likely resembles. To complete feature filtering, one-hot data features are acquired through

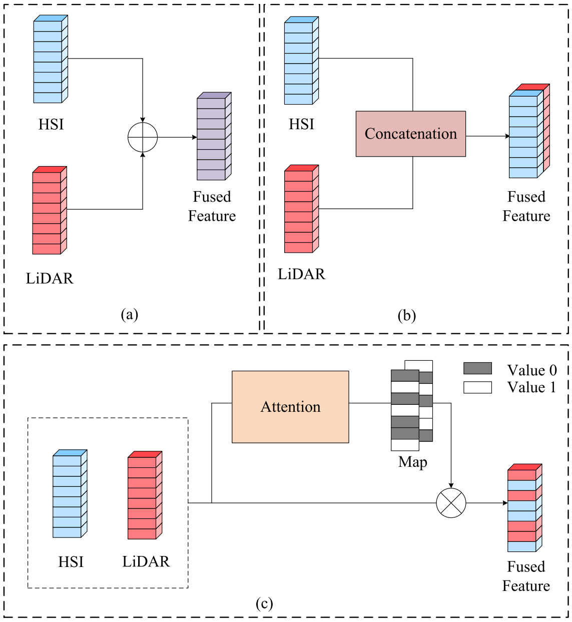


Fig. 3. Comparison of three fusion modes of multisource features. (a) Linear addition fusion. (b) Linear concatenation fusion. (c) MSFSM.

the hard sampling process of token-wise. Then, the one-hot features are mapped into concatenated multisource feature to finish the fusion work

$$\omega_i = \mathrm{argmax}(f_1, f_2)$$
$$F_f = \sum_{i=1}^{2} \omega_i [Concat(F_H, F_L)]_i \tag{8}$$

where f_1, f_2 are each channel of F, ω_i is the one-hot map, and F_f is the fused feature.

Through MSFSM, multisource features can be effectively combined into a mixed feature, and any token of sequence is described as one of the source data that is more important for classification performance. Fig. 3 shows the comparison of different fusion methods. It can be observed that MSFSM is not only a process of fusing multisource features but also a feature screening process; it is worth noting that MSFSM is a dynamic feature screening process. When training the model, the FC layer is continuously optimized by adopting the reparameterization method gumbel softmax, which allows the discrete sampling process to propagate gradients.

After the fusion feature is obtained, a convolution transformer is adopted to further optimize the fusion feature. Finally, the class token of the feature is separated and is sent to the classification module to obtain the classification result of the fused feature.

III. Experimental and Analysis

In this section, three well-known datasets (i.e., Houston, MUUFL, and Trento) and three evaluation metrics [i.e., average accuracy (AA), overall accuracy (OA), and kappa coefficient (Kappa)] are applied to analyze the parameters of LIIT and compare the performance of various state-of-the-art CNNs and transformers. The LIIT method is implemented based on Python 3.7. and its deep learning framework is built by PyTorch, which is proposed by Facebook. It not only provides a convenient deep learning system, but also supports the code running process for GPU acceleration. All our experiments are performed on a PC with Windows 10 OS, Intel Core i7-7800X CPU, 32-GB RAM, and an NVIDIA GeForce RTX 1080 Ti GPU. The implementation process of CASST is presented in Algorithm 1.

Algorithm 1: LIIT.

```
Input: The raw HSI data $\mathbf{X}_H \in \mathbb{R}^{H\times W\times C}$, LiDAR data
  $\mathbf{X}_L \in \mathbb{R}^{H\times W}$, and ground truth $\mathbf{Y} \in \mathbb{R}^{H\times W}$.
Output: Classification result and relative visualization
  map for three datasets.
1:  Reduce the dimension of HSI $\mathbf{X}_H$ and along with
    LiDAR data $\mathbf{X}_L$ to expand each pixel into spatial
    patch.
2:  Obtain the training set, validation set, and testing set,
    then build their respective dataloaders
3:  Set train batchsize $b = 64$, optimizer Adam with the
    learning rate $lr = 0.001$, and train epoches $e = 150$.
4:  for $e = 1$ to 150 do
5:    Embedding multisource data by convolutional
      feature embedding module.
6:    Perform the L-MSFI.
7:    Perform the MSFSM module.
8:    Extract the class token in the fused feature.
9:    Classify the fused class token using the MLP Head
      and SoftMax.
10: end for
11: Save the trained model to classify the testing set, and
    plot the visualization map.
```

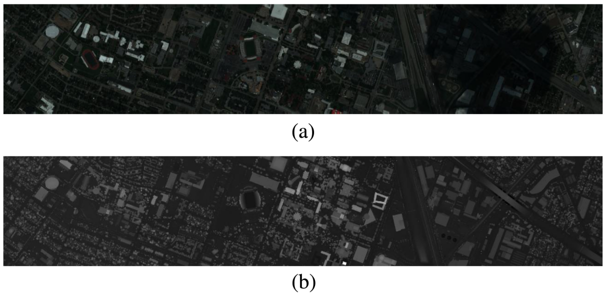


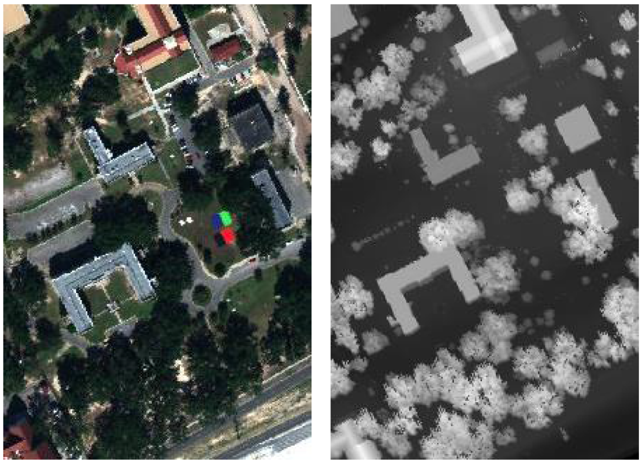
Fig. 4. Houston dataset. (a) Pseudocolor image for HSI. (b) LiDAR DSM map.

A. Datasets

1) Houston Dataset: The Houston 2013 dataset was taken over the University of Houston and its neighboring cities, which was initially used in the 2013 IEEE GRSS data fusion contest. This dataset contains HSI and LiDAR DSM data with a spatial size of 349 × 1905 and a spatial resolution of 2.5 m. The HSI bands range from 0.38–1.05 μm and contain 144 available bands. It has 15 categories and 15 029 sample pixels. Fig. 4 shows the HSI pseudocolor map and LiDAR DSM map of the dataset, and

TABLE I
NUMBER OF TRAINING, VALIDATION, AND TESTING SAMPLES FOR THE HOUSTON DATASET

Class		Number of samples		
No	Name	Training	Validation	Testing
1	Health grass	198	158	895
2	Stressed grass	190	160	904
3	Synthetic grass	192	75	430
4	Trees	188	158	898
5	Soil	186	158	898
6	Water	182	21	122
7	Residential	196	160	912
8	Commercial	191	158	895
9	Road	193	159	900
10	Highway	191	155	881
11	Railway	181	158	896
12	Parking lot 1	192	156	885
13	Parking lot 2	184	43	242
14	Tennis court	181	27	220
15	Running track	187	28	445
	Total	2832	1774	10423



(a) (b)

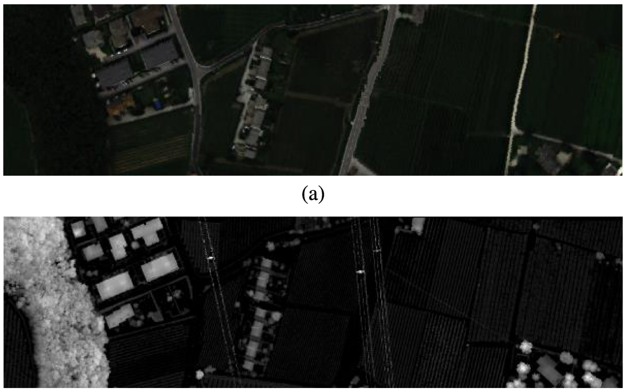
Fig. 5. MUUFL dataset. (a) Pseudocolor image for HSI. (b) LiDAR DSM map.

Table I shows the sample allocation when the dataset is used for the experiment, in which all samples are selected randomly among the categories.

2) MUUFL Dataset: The MUUFL dataset was collected at the International University of Southern Mississippi campus and contains LiDAR DSM data acquired by the Gemini LiDAR and HSI data captured by the CASI-1500. The spatial size of the HSI and LiDAR data is 325 × 220, the spatial resolution of HSI data is 0.54 × 1.0 m, and it contains a total of 64 available bands from 375−1050 nm. There are 11 classes and 53 687 sample pixels. Fig. 5 shows the HSI pseudocolor map and LiDAR DSM map of the dataset, and Table II shows the sample allocation when the dataset is used for experimental comparison.

TABLE II
NUMBER OF TRAINING, VALIDATION, AND TESTING SAMPLES FOR THE MUUFL DATASET

Class		Number of samples		
No	Name	Training	Validation	Testing
1	Trees	100	3472	19674
2	Mostly grass	100	625	3545
3	Mixed ground	100	1017	5765
4	Dirt and sand	100	259	1467
5	Roads	100	988	5599
6	Water	100	55	311
7	Building shadows	100	320	1813
8	Buildings	100	921	5219
9	Sidewalks	100	193	1092
10	Yellow curbs	100	12	71
11	Cloth panels	100	25	144
	Total	1100	7887	44700



(a)

(b)

Fig. 6. Trento dataset. (a) Pseudocolor image for HSI. (b) LiDAR DSM map.

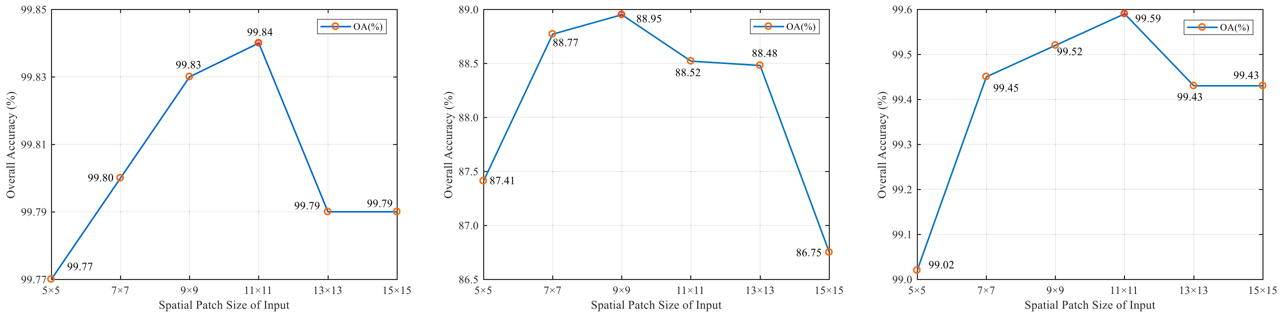
TABLE III
NUMBER OF TRAINING, VALIDATION, AND TESTING SAMPLES FOR THE TRENTO DATASET

Class		Number of Samples		
No	Name	Training	Validation	Testing
1	Apple trees	72	594	3368
2	Buildings	69	425	2409
3	Ground	58	63	358
4	Wood	86	1355	7682
5	Vineyard	102	1560	8839
6	Roads	68	466	2640
	Total	455	4463	25296

3) Trento Dataset: The Trento dataset was acquired in a rural area in southern Trento, Italy. The spatial size of the HSI and LiDAR DSM data is 166 × 600 and the spatial resolution is 1 m. The number of available spectral bands for HSI data is 63. It contains six object categories with a total of 30 214 sample pixels. Fig. 6 shows the HSI pseudocolor map and LiDAR DSM map of the dataset, and Table III shows the number of samples in the training set, validation set, and testing set when the dataset is used for experimental comparison. The sampling method is a random sampling of each category.

B. Experimental Setup

In this part, we conduct a series of parameter experiments to analyze and confirm which parameters are most beneficial to the performance expression of LIIT, including the number of heads of the self-attention and the patch size of the network input. In addition, some parameters are set by default based on experience. 1) The training epochs are 150, and the training batch size and test batch size are set to 64 and 1000, respectively.



(a) (b) (c)

Fig. 7. Influence of spatial patch size as network input on classification performance. (a) Houston dataset. (b) MUUFL dataset. (c) Trento dataset.

TABLE IV
Overall Accuracy (%) With Different Number of Heads for Proposed LIIT on Three Datasets

Number of Heads	Datasets		
	Houston	MUUFL	Trento
1 heads	99.77	87.85	99.31
2 heads	99.84	88.46	99.48
4 heads	99.83	87.90	99.56
8 heads	99.78	88.02	99.59
16 heads	99.81	87.41	99.51
32 heads	99.77	86.54	99.51

TABLE V
Overall Accuracy (%) With Different Number of PCs After PCA for Proposed LIIT on Three Datasets

Number of PCs	Datasets		
	Houston	MUUFL	Trento
1 PC	96.38	79.50	99.53
10 PCs	99.73	89.30	99.60
20 PCs	99.80	88.73	99.58
30 PCs	99.84	88.95	99.59
40 PCs	99.82	88.30	99.54
50 PCs	99.79	88.38	99.53

2) The gradient optimization algorithm uses adaptive moment estimation (Adam), and the learning rate is 0.001. 3) The number of feature channels during the transformer is 64.

The number of heads determines the effectiveness of the self-attention process to a certain extent. By assigning features to different subspaces for attention calculation and then aggregating them, this process makes the data processing more refined and the processed features more discriminative. Table IV shows the comparison of the classification performance of the LIIT method on three common datasets with different numbers of heads. LIIT is most suitable when the number of heads is 2 on the Houston and MUUFL datasets, while the number of heads is 8 on the Trento dataset. In addition, it can still be observed that with the increasing of the number of heads, the classification performance decreases instead of increasing. This is mainly due to the continuous growth of the number of heads, resulting in mutual redundancy of subspace information, which inhibits the performance expression.

In the fusion classification task of HSI and LiDAR data, the sufficiency of spatial information interaction is the key to improving the classification accuracy of the model. Selecting a more appropriate input patch size can further stimulate the potential performance of the model. Fig. 7 shows the comparison results of the classification performance of LIIT when the spatial input patch size is 5 to 15. The results show that the optimal size is 11 × 11 under the Houston and Trento datasets, and 9 × 9 under the MUUFL dataset. As the patch size grows, the introduction of different categories of pixels will further interfere with the expression of the original features of the patch and exacerbate this phenomenon through the self-attention process. It is worth noting that the spatial patch used in the MUUFL dataset is smaller than that in the Houston and Trento datasets due to its complex land cover distribution.

PCA process initially affects the quality of the HSI feature. Appropriate number of PCs can retain important feature information as well as remove redundant information. Table V shows how the number of PCs affects the model accuracy of LIIT methods. The results show that when PCs are ten, the accuracy is best on MUUFL and Trento datasets and 30 on Houston datasets.

C. Experimental Comparison With Competitive Methods

To verify the effectiveness and excellent classification performance of the proposed LIIT method, we compare the algorithm performance with some traditional methods and the state-of-the-art method on the Houston, MUUFL, and Trento datasets. The traditional methods include SVM and EMAP based on the morphological algorithm. The state-of-the-art methods include representative CNN-based methods, such as 3-DCNN, TBCNN, and CPCNN, as well as transformer-based ViT and SpectralFormer. The selection of all samples in this process is shown in Section III-A. Each method has been tested ten times, and its mean value is represented as its final classification result.

1) SVM [49] algorithm obtains the surface feature classification labels by analyzing the HSI spectral features. The implementation of the algorithm is based on LIBSVM toolbox of MATLAB. With Gaussian RBF kernel function, the model is trained by fivefold cross-validation.
2) The EMAP [50] method fully captures the morphological features of multimodal features by obtaining the expanded multiattribute profiles of HSI and LiDAR. In the implementation process, HSI data is reserved to three

TABLE VI
CLASSIFICATION PERFORMANCE OF THE SVM, EMAP, 3-DCNN, CPCNN, TBCNN, VIT, SPECTRALFORMER, AND LIIT CLASSIFICATION METHOD ON THE HOUSTON DATASET IN TERMS OF OA, AA, AND KAPPA

No	Class	**Classification performance of various methods**							
		SVM	EMAP	3-DCNN	CPCNN	TBCNN	ViT	SpectralFormer	LIIT
🟩 1	Healthy grass	98.02	99.62	98.81	90.69	**99.62**	98.17	97.91	98.78
🟩 2	Stressed grass	98.24	99.06	98.98	99.72	88.06	99.32	98.68	**100.0**
🟩 3	Synthetic grass	99.80	**100.0**	100.0	99.80	100.0	99.95	99.80	99.77
🟩 4	Trees	98.19	99.71	99.14	99.91	99.15	98.78	98.86	**99.89**
🟫 5	Soil	98.24	99.90	99.85	99.91	99.62	99.92	99.81	**100.0**
🟦 6	Water	99.79	100.0	99.93	100.0	100.0	99.18	98.60	**100.0**
⬜ 7	Residential	95.01	99.34	98.64	98.13	99.16	97.69	97.48	**100.0**
🟪 8	Commercial	97.42	99.62	97.48	97.34	95.35	99.15	97.05	**99.89**
🟥 9	Roads	86.33	95.47	94.59	89.90	98.96	96.21	91.78	**100.0**
🟥 10	Highway	90.94	98.46	99.37	91.99	91.02	99.56	99.71	**100.0**
🟦 11	Railway	90.13	99.04	99.87	89.37	87.95	99.64	98.58	**100.0**
🟨 12	Parking Lot 1	93.66	97.74	98.91	93.85	88.28	98.00	95.87	**99.89**
🟧 13	Parking Lot 2	88.85	98.93	99.59	96.84	**100.0**	97.15	87.72	99.17
🟪 14	Tennis Court	97.22	97.62	100.0	99.19	99.60	100.0	100.0	**100.0**
🟧 15	Running Track	98.72	100.0	100.0	100.0	100.0	99.95	100.0	**100.0**
	OA (%)	94.94	98.88	98.75	95.63	95.44	98.75	97.59	**99.84**
	AA (%)	95.38	98.97	99.01	96.39	96.45	98.85	97.46	**99.83**
	$\kappa \times 100$	94.50	98.79	98.64	95.26	95.05	98.64	97.38	**99.68**

The bold entities indicate the best classification accuracy of each category and each evaluation metric.

channels through PCA method, and morphological algorithm is used to expand HSI and LiDAR data to 60-band profiles and 15-band profiles, respectively.

3) Three groups of 3 × 3 × 3 3-D convolutional layers, batch normalization layers, ReLUs, and max pooling layers are used in the 3-DCNN method [51]. In addition, the spatial patch size of the input feature is set to 11 × 11 on the three datasets.
4) Coupled CNN optimizes the drawbacks of the traditional two-branch CNN model used for multimodal data classification tasks. By sharing the network weight of dual branches, it can guide the mutual communication between features, help feature fusion, and reduce training time. In the experiment, the input feature spatial patch of HSI and LiDAR is set to 11 × 11.
5) TBCNN adopts the model design of tow-branch CNN to extract the spatial-spectral features of HSI and the elevation features of LiDAR data, respectively. During the implementation, the parameters shall be consistent with the code provided in the original paper. The training epochs are set to 100, and PCs are 30.
6) ViT method is the first time to introduce transformer model into the field of computer vision. In the multimodal classification task, the spatial input is set to 9 × 9, and the patch embedding process adopts a nonoverlapping 3 × 3 convolutional process.
7) SpectralFormer [52] improves the input mode of ViT. By using band grouping input, the model can extract local-based HSI spectral features. The experimental parameters and epochs required for training are consistent with the code provided in the article.
8) The parameter design of the LIIT method proposed in this article can be seen in part B of Section III, and its parameters are set to the values with the best classification performance on various datasets. 150 epochs are required for model training.

Note that all the traditional methods are implemented on MATLAB, and all the methods based on deep learning are coded using Python 3.7.

Tables VI–VIII summarize the classification performance [i.e., OA (%), AA (%), and Kappa] for various classification methods on the Houston, MUUFL, Trento datasets. As shown in the Tables, LIIT outperforms other method on the three datasets.

1) Houston Dataset: Table VI shows the comparison of the classification accuracy of each experimental method on the Houston dataset. As we can see, the SVM algorithm that fails to describe the spatial surface structure can only obtain 94.94% OA. The TBCNN model based on spatial blocks has excellent local space coding ability. On the premise of HSI spectral dimension feature extraction, it effectively captures and fuses heterogeneous features between HSI and LiDAR data, with 95.44% OA. CPCNN promotes the information consistency among multimodal features through the introduction of weight contribution mechanism, and further improves the classification performance. The implantation of the attention module makes the communication between features closer and more recognizable. As a typical attention-based model, ViT obtains the contextual semantic information of spatial blocks, establishes global dependencies, and adaptively filters important features. It has excellent classification performance on the Houston dataset. However, due to the lack of exploring the local spatial features, the ViT model obviously has a performance bottleneck. The LIIT method proposed in this article solves the above problems. With the use of MSFSM, the amount of data information is increased and the feature recognition is enhanced. Compared with ViT, the classification accuracy of LIIT in this dataset is improved by 1.1%. As shown in Fig. 8, the full pixel classification result map of each method on the Houston dataset, and the map presented by LIIT, is more accurate for the description of ground objects.

2) MUUFL Dataset: Table VII and Fig. 9, respectively, show the classification accuracy comparison and classification results

TABLE VII
CLASSIFICATION PERFORMANCE OF THE SVM, EMAP, 3-DCNN, CPCNN, TBCNN, VIT, SPECTRALFORMER, AND LIIT CLASSIFICATION METHOD ON THE MUUFL DATASET IN TERMS OF OA, AA, AND KAPPA

No	Class	**Classification performance of various methods**							
		SVM	EMAP	3-DCNN	CPCNN	TBCNN	ViT	SpectralFormer	LIIT
1	Trees	97.93	**98.94**	81.24	92.84	83.44	84.51	90.46	91.85
2	Mostly Grass	53.02	75.33	72.61	43.02	**85.64**	77.22	81.58	82.54
3	Mixed Ground	79.84	**86.26**	69.26	82.03	71.35	68.05	67.75	77.71
4	Dirt and Sand	78.10	88.79	87.75	**98.73**	88.24	92.32	95.89	95.28
5	Roads	86.78	**92.41**	80.90	77.24	91.74	72.08	84.15	86.27
6	Water	60.89	56.52	98.88	**100.0**	100.0	99.45	98.63	98.88
7	Building Shadows	50.83	53.41	84.38	88.84	**90.76**	80.74	89.87	88.37
8	Buildings	94.82	94.90	84.39	**98.32**	90.68	92.81	95.77	93.87
9	Sidewalks	50.44	62.47	57.35	37.35	74.16	45.45	65.68	**75.21**
10	Yellow Curbs	35.26	27.09	**87.09**	43.37	78.31	75.21	80.72	85.77
11	Cloth Panels	72.80	86.60	99.38	98.22	96.45	96.11	98.22	**99.38**
OA (%)		82.25	88.31	79.28	84.85	84.32	80.50	86.27	**88.46**
AA (%)		69.19	74.79	82.11	78.18	86.43	80.36	86.25	**88.65**
$\kappa \times 100$		77.16	84.84	73.53	80.04	79.90	74.98	82.12	**84.88**

The bold entities indicate the best classification accuracy of each category and each evaluation metric.

TABLE VIII
CLASSIFICATION PERFORMANCE OF THE SVM, EMAP, 3-DCNN, CPCNN, TBCNN, VIT, SPECTRALFORMER, AND LIIT CLASSIFICATION METHOD ON THE TRENTO DATASET IN TERMS OF OA, AA, AND KAPPA

No	Class	**Classification performance of various methods**							
		SVM	EMAP	3-DCNN	CPCNN	TBCNN	ViT	SpectralFormer	LIIT
1	Apple Trees	70.12	96.27	98.46	99.20	**99.72**	97.39	91.09	99.17
2	Buildings	96.40	96.73	85.82	97.60	91.67	97.83	97.67	**98.71**
3	Ground	76.40	99.04	94.08	**99.33**	91.92	97.82	95.49	98.91
4	Wood	99.76	99.81	99.79	99.99	95.32	99.97	98.64	**100.0**
5	Vineyard	94.19	99.57	99.52	99.89	97.07	99.39	98.98	**100.0**
6	Roads	94.24	97.31	93.05	97.04	98.55	96.39	91.82	**98.55**
OA (%)		91.71	98.68	97.40	99.30	96.46	98.84	96.91	**99.59**
AA (%)		88.52	98.12	95.12	98.84	95.71	98.13	95.62	**99.22**
$\kappa \times 100$		89.02	98.24	96.52	99.07	95.28	98.41	95.82	**99.46**

The bold entities indicate the best classification accuracy of each category and each evaluation metric.

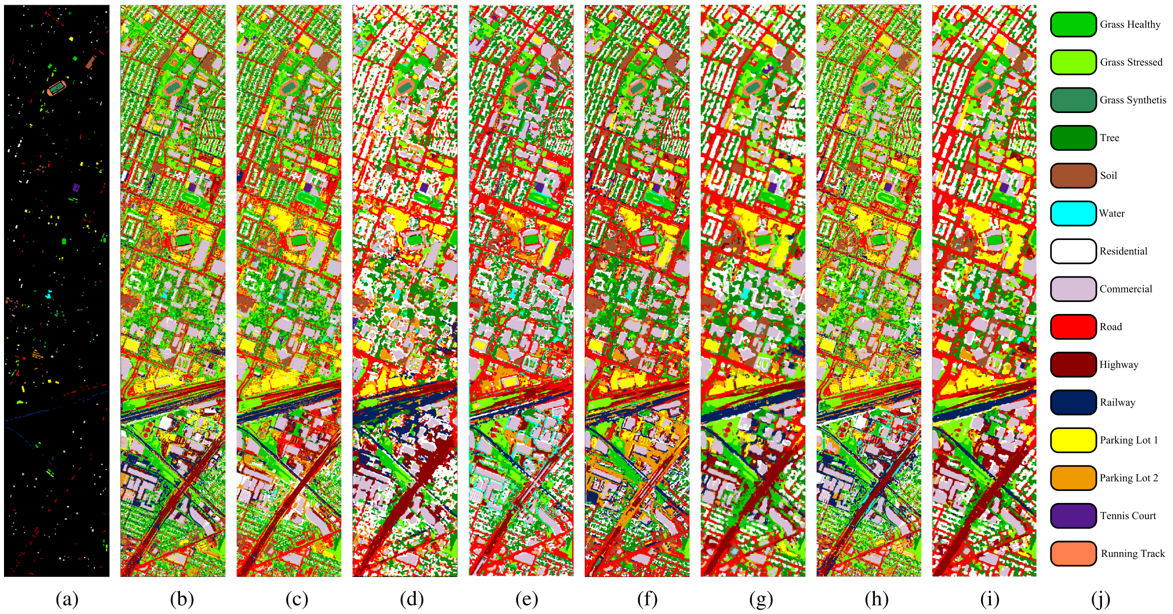


Fig. 8. Classification result maps for different comparison methods on the Houston dataset. (a) Ground truth, (b) SVM (OA = 94.94%), (c) EMAP (OA = 98.88%), (d) 3-DCNN (OA = 98.75%), (e) CPCNN (OA = 95.63%), (f) TBCNN (OA = 95.44%), (g) ViT (OA = 98.75%), (h) SpectralFormer (OA = 97.59%), (i) LIIT (OA = 99.84%), and (j) color map.

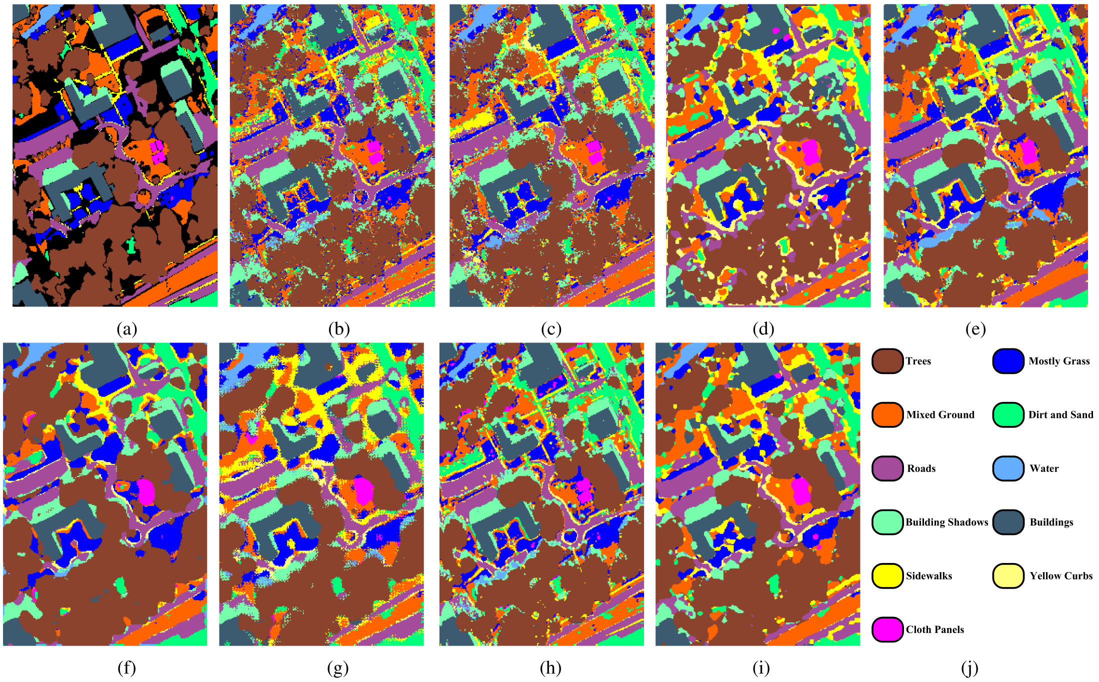


Fig. 9. Classification result maps for different comparison methods on the MUUFL dataset. (a) Ground truth, (b) SVM (OA = 82.25%), (c) EMAP (OA = 88.31%), (d) 3-DCNN (OA = 79.28%), (e) CPCNN (OA = 84.85%), (f) TBCNN (OA = 84.32%), (g) ViT (OA = 80.50%), (h) SpectralFormer (OA = 86.27%), (i) LIIT (OA = 88.46%), and (j) color map.

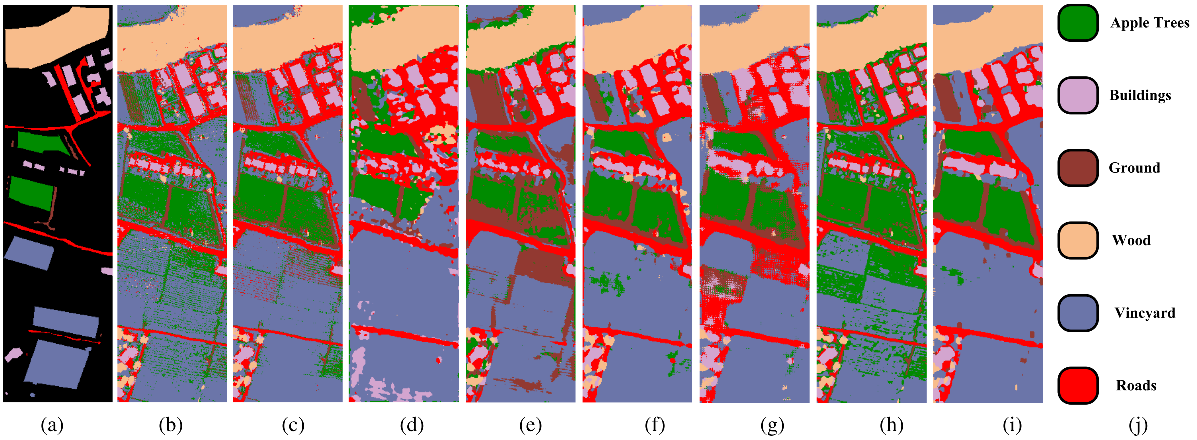
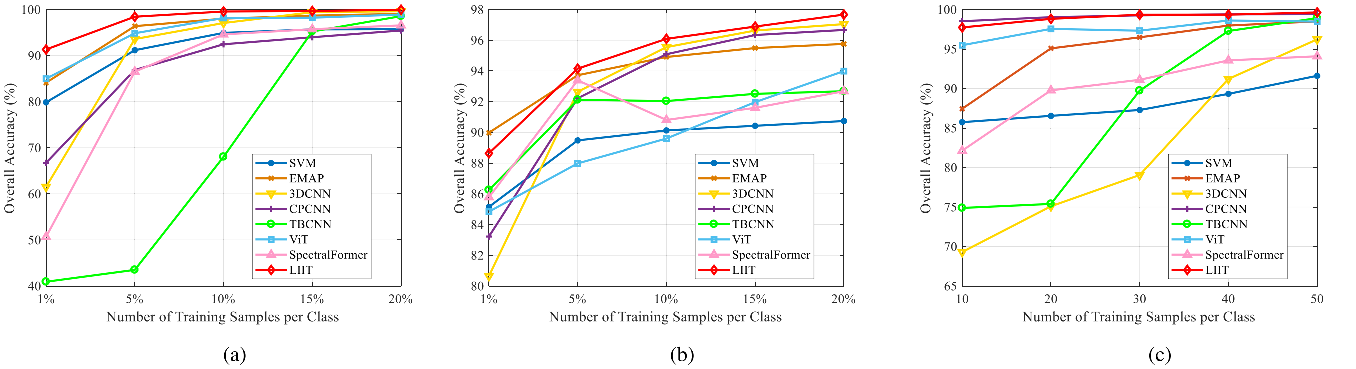


Fig. 10. Classification result maps for different comparison methods on the Trento dataset. (a) Ground truth, (b) SVM (OA = 91.71%), (c) EMAP (OA = 98.68%), (d) 3DCNN (OA = 97.40%), (e) CPCNN (OA = 99.30%), (f) TBCNN (OA = 96.46%), (g) ViT (OA = 98.84%), (h) SpectralFormer (OA = 96.91%), (i) LIIT (OA = 99.59%), and (j) color map.

of each classification method on the MUUFL dataset. It can be observed that because the CNN-based method focuses on the description of local space, its description of ground objects is smoother and mostly blocky. Others are rougher, showing more map noise. The LIIT method has better classification performance, and the generated map has fewer noise pixels and fewer misclassification phenomena. Compared with the ViT model, the introduction of convolution layer enables LIIT to have stronger spatial feature-encoding capability and more reasonable image description; Compared with TBCNN and CPCNN, LIIT is more effective in fusing HSI and LiDAR data, and more detailed in expressing pixel boundaries.

3) Trento Dataset: Table VIII and Fig. 10, respectively, show the classification comparison of each classification algorithm on the Trento dataset. The Trento dataset is relatively easy to be classified due to its orderly distribution of ground objects and blocky space, but it still has certain challenges in some categories. For example, in the buildings and roads categories, the similarity between the two on the spectral curve has caused difficulties in the classification of single HSI data. Therefore, how to effectively combine HSI and LiDAR data is the key to overcoming this problem. CPCNN introduced weight contribution into feature extraction to promote the consistency between multimodal data, and the classification accuracy on both categories exceeded 97%



(a)

(b)

(c)

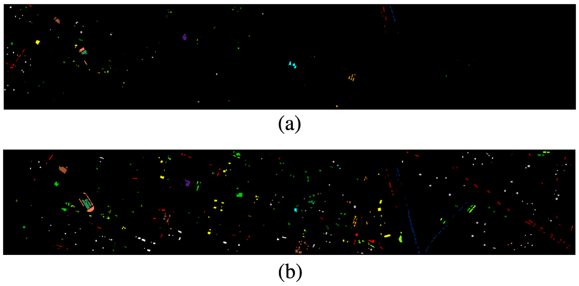
Fig. 11. Classification performance of various methods with different number of training samples on three datasets. (a) Houston dataset. (b) MUUFL dataset. (c) Trento dataset.

TABLE IX
Classification Accuracy of the Proposed LIIT Method Using Different Ablation Strategies on the Three Multisource Datasets

HSI	LiDAR	L-MSFI	MSFSM	Houston			MUUFL			Trento		
				OA (%)	AA (%)	$\kappa \times 100$	OA (%)	AA (%)	$\kappa \times 100$	OA (%)	AA (%)	$\kappa \times 100$
√	×	×	×	66.44	72.14	63.82	72.21	63.61	65.05	95.21	87.02	93.63
×	√	×	×	46.30	48.11	42.31	53.57	34.90	39.55	89.19	73.90	85.23
√	√	×	×	69.25	73.68	66.77	73.02	63.83	63.93	97.37	95.50	96.50
√	√	√	×	70.60	74.45	68.22	74.19	64.56	66.70	97.78	94.22	97.04
√	√	√	√	72.63	75.75	70.08	76.62	68.67	70.16	98.09	97.09	97.46

OA. The LIIT proposed in this article promotes the communication between multimodal data through the design of an interactive structure. The use of MSFSM adaptively estimates the importance of features and further overcomes the redundancy problem of fusion features. The classification accuracy of this method in buildings and roads categories exceeds 98.5% OA, which is superior to each state-of-the-art method, which proves the rationality and effectiveness of LIIT in feature extraction and data fusion.

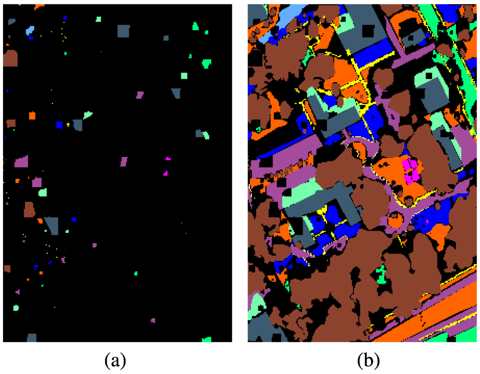
In order to analyze the performance robustness of LIIT when the number of samples changes, we further carried out the classification results comparison experiment between LIIT and various methods when the number of samples changes from less to more. Among them, the selection of training samples on the Houston and MUUFL datasets is 1%−20% of the total samples and there are 10−50 samples on the Trento dataset. The experimental results are shown in Fig. 11. Not only in large sample size, but also in small sample size, the classification performance of the LIIT method is still better than those of the advanced CNNs and transformers. This experiment shows that the LIIT method has strong robustness and excellent classification performance. In addition, the most outstanding classification results are obtained on all datasets, which also proves that the method is strong in universality and can be applied to various multimodal remote sensing data classification scenarios.



(a)

(b)

Fig. 12. Housotn dataset. (a) Training labels for ablation experiment. (b) Testing labels for ablation experiment.



(a)

(b)

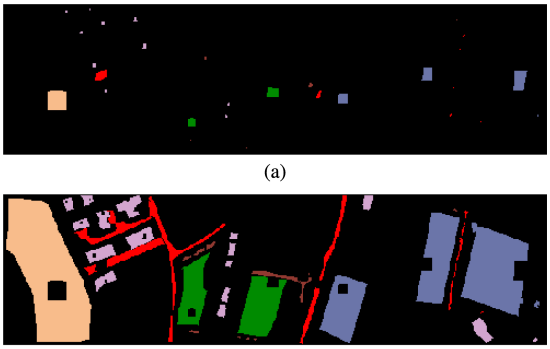
Fig. 13. MUUFL dataset. (a) Training labels for ablation experiment. (b) Testing labels for ablation experiment.

D. Ablation Experiments for the Proposed Method

In order to observe the impact of each module on the model performance more intuitively, we reselected and allocated the samples on the three datasets in block allocation mode, making them more challenging for classification. Figs. 12–14 show the distribution of training samples and test samples for Houston, MUUFL, and Trento datasets, respectively. For Table IX, the results of ablation experiments show that in the joint classification task of HSI and LiDAR, the use of a single HSI data

TABLE X
COMPARISON OF RUNNING TIME OF EACH METHOD ON DIFFERENT DATASETS

		Running time of various methods							
		SVM	EMAP	3-DCNN	CPCNN	TBCNN	ViT	SpectralFormer	LIIT
Houston	Train(s)	223.09	154.76	211.41	411.85	111.08	76.21	256.65	411.03
	Test(s)	2.43	0.98	1.41	0.21	0.28	1.31	2.12	10.83
MUUFL	Train(s)	26.83	35.48	139.51	245.07	31.57	53.59	206.24	124.49
	Test(s)	2.16	4.29	12.17	0.95	0.99	5.08	3.64	13.52
Trento	Train(s)	5.27	6.51	57.89	110.65	108.38	22.51	107.85	54.16
	Test(s)	0.58	1.23	5.71	0.51	0.54	3.16	1.98	8.23


(a)

(b)

Fig. 14. Trento dataset. (a) Training labels for ablation experiment. (b) Testing labels for ablation experiment.

has superior classification performance, indicating that HSI data plays a leading role in the classification process, and the LiDAR data is used as a supplement to the information level. With the introduction of LiDAR data, the classification accuracy of each dataset has significantly increased, which shows that the reasonable use of LiDAR data can help it to achieve a more detailed description of ground objects. L-MSFI is designed to extract the local-global semantic features of multisource data, and its interactive information transmission structure helps to achieve communication between features. The results show that both the Houston and MUUFL datasets help improve OA by about 1.1%. The introduction of MSFSM replaces the traditional linear addition and concatenation fusion actions. With the dynamic model training process, the high-weight features in the multimode are adaptively selected. This process avoids the generation of fusion redundancy while ensuring sufficient information. As shown in the table, the impact of MSFSM on performance is also critical. The results of ablation experiments show that all modules and modal data in the LIIT play an indispensable role in breaking through the model performance bottleneck. With the introduction of each component, the model classification accuracy continues to rise, which is sufficient to prove the rationality of each component and the effectiveness of performance improvement.

E. Analysis of Running Time

Time complexity is another important index to describe the model. Table X shows the comparison of training and testing time required for each method to complete a classification process. It can be seen that CNN-based methods generally have high time complexity. The transformer-based models also has many network parameters due to its embedded self-attention module. The LIIT method is not superior to other methods in terms of time due to the use of transformer and convolutional layer. It is worth noting that the number of samples has a significant impact on the running time. With the increase of the number of samples, the running time of the model increases nonlinearly, especially for the complex model LIIT, which is reflected in Table X that the training time of the LIIT method on the Houston dataset is significantly longer than the other two datasets. In addition, the dual branch model has more structural parameters than the single branch model, but it has a more adequate feature extraction process, and the benefit of its performance improvement is considerable.

IV. CONCLUSION

In this article, an LIIT model is proposed to solve the problems of incomplete HSI and LiDAR data collaborative feature capture and insufficient multisource feature fusion. Specifically, a local-based multisource feature interactor (L-MSFI) is designed. Its local-based feature modeling process alleviates the feature homogeneity of self-attention. Meanwhile, a HSI and LiDAR data interactive feature coding environment has been created, which promotes mutual learning between multisource features. In addition, an MSFSM is developed to dynamically filter multimodal features, overcoming the balance problem of HSI and LiDAR features in fusion. The comparative analysis experiment was carried out on three multisource remote sensing classification datasets (Houston, MUUFL, and Trento). Compared with the state-of-the-art CNNs and transformers, LIIT has more performance advantages. In the future, lightweight multisource remote sensing classification model is our goal to better balance performance and network complexity.

REFERENCES

[1] M. E. Paoletti, J. M. Haut, R. Fernandez-Beltran, J. Plaza, A. J. Plaza, and F. Pla, "Deep pyramidal residual networks for spectral-spatial hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 740–754, Feb. 2019.

[2] C. Yu, R. Han, M. Song, C. Liu, and C.-I. Chang, "A simplified 2D-3D CNN architecture for hyperspectral image classification based on spatial-spectral fusion," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 2485–2501, Apr. 2020.

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

[4] L. Mou, X. Lu, X. Li, and X. X. Zhu, "Nonlocal graph convolutional networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 12, pp. 8246–8257, Dec. 2020.

[5] X. Zhang, Y. Sun, K. Shang, L. Zhang, and S. Wang, "Crop classification based on feature band set construction and object-oriented approach using hyperspectral images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 9, pp. 4117–4128, Sep. 2016.

[6] T. Chen, X. Zheng, R. Niu, and A. Plaza, "Open-pit mine area mapping with Gaofen-2 satellite images using U-Net+," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 3589–3599, Apr. 2022.

[7] D. Hong, N. Yokoya, J. Chanussot, and X. X. Zhu, "Cospace: Common subspace learning from hyperspectral-multispectral correspondences," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 4349–4359, Jul. 2019.

[8] Z. Lin, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Generative adversarial networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 9, pp. 5046–5063, Sep. 2018.

[9] H. Aytaylan and S. E. Yuksel, "Fully-connected semantic segmentation of hyperspectral and LiDAR data," *IET Comput. Vis.*, vol. 13, no. 3, pp. 285–293, 2019.

[10] Z. Wen, B. Hu, L. Jing, M. E. Woods, and P. Courville, "Automatic forest species classification using combined LiDAR data and optical imagery," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2008, pp. III-134–III-137.

[11] S. Morsy, A. Shaker, and A. El-Rabbany, "Multivariate Gaussian decomposition for multispectral airborne LiDAR data classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2018, pp. 8741–8744.

[12] M. Dalponte, H. O. Orka, T. Gobakken, D. Gianelle, and E. Naesset, "Tree species classification in boreal forests with hyperspectral data," *IEEE Trans. Image Process.*, vol. 51, no. 5, pp. 2632–2645, May 2012.

[13] T. Matsuki, N. Yokoya, and A. Iwasaki, "Hyperspectral tree species classification of japanese complex mixed forest with the aid of LiDAR data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens*., vol. 8, no. 5, pp. 2177–2187, May 2015.

[14] J. Vauhkonen, T. Hakala, J. Suomalainen, S. Kaasalainen, O. Nevalainen, and M. Vastaranta, "Classification of spruce and pine trees using active hyperspectral LiDAR," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 5, pp. 1138–1141, Sep. 2013.

[15] P. Ghamisi, B. Höfle, and X. X. Zhu, "Hyperspectral and LiDAR data fusion using extinction profiles and deep convolutional neural network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 6, pp. 3011–3024, Jun. 2017.

[16] Q. Cao, Y. Zhong, A. Ma, and L. Zhang, "Urban land use/land cover classification based on feature fusion fusing hyperspectral image and LiDAR data," in *Proc. IEEE Int. Geosci. Remote Sens. Symp*., 2018, pp. 8869–8872.

[17] Y. Zhang, H. L. Yang, S. Prasad, E. Pasolli, J. Jung, and M. Crawford, "Ensemble multiple kernel active learning for classification of multisource remote sensing data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 2, pp. 845–858, Feb. 2015.

[18] M. Pedergnana, P. R. Marpu, M. Dalla Mura, J. A. Benediktsson, and L. Bruzzone, "Classification of remote sensing optical and LiDAR data using extended attribute profiles," *IEEE J. Sel. Topics Signal Process.*, vol. 6, no. 7, pp. 856–865, Nov. 2012.

[19] B. Rasti, P. Ghamisi, and R. Gloaguen, "Hyperspectral and LiDAR fusion using extinction profiles and total variation component analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3997–4007, Jul. 2017.

[20] S. Jia et al., "Multiple feature-based superpixel-level decision fusion for hyperspectral and LiDAR data classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 2, pp. 1437–1452, Feb. 2021.

[21] C. Zhang, M. Smith, and C. Fang, "Evaluation of Goddard's LiDAR, hyperspectral, and thermal data products for mapping urban land-cover types," *GIScience Remote Sens.*, vol. 55, no. 1, pp. 1–20, 2017.

[22] Y. Chen, Z. Lin, Z. Xing, W. Gang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2094–2107, Jun. 2017.

[23] P. Zhong, Z. Gong, S. Li, and C. B. Schonlieb, "Learning to diversify deep belief networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 6, pp. 3516–3530, Jun. 2017.

[24] X. Zhang, Y. Sun, J. Kai, L. Chen, L. Jiao, and H. Zhou, "Spatial sequential recurrent neural network for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 11, pp. 4141–4155, Nov. 2018.

[25] S. Jia et al., "3-D gabor convolutional neural network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–16, Jun. 2022, Art. no. 5509216.

[26] X. Xu, W. Li, Q. Ran, Q. Du, L. Gao, and B. Zhang, "Multisource remote sensing data classification based on convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 937–949, Feb. 2018.

[27] R. Hang, Z. Li, P. Ghamisi, D. Hong, G. Xia, and Q. Liu, "Classification of hyperspectral and LiDAR data using coupled CNNs," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 4939–4950, Jul. 2020.

[28] D. Hong, L. Gao, R. Hang, B. Zhang, and J. Chanussot, "Deep encoder-decoder networks for classification of hyperspectral and LiDAR data," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, Aug. 2020, Art. no. 5500205.

[29] M. Zhang, W. Li, Q. Du, L. Gao, and B. Zhang, "Feature extraction for classification of hyperspectral and LiDAR data using patch-to-patch CNN," *IEEE Trans. Cybern.*, vol. 50, no. 1, pp. 100–111, Jan. 2020.

[30] B. Tu, W. He, W. He, X. Ou, and A. Plaza, "Hyperspectral classification via global-local hierarchical weighting fusion network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 184–200, Dec. 2021.

[31] X. Zhao, R. Tao, W. Li, W. Philips, and W. Liao, "Fractional Gabor convolutional network for multisource remote sensing data classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–18, Mar. 2022, Art. no. 5503818.

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

[33] H.-C. Li, W.-S. Hu, W. Li, J. Li, Q. Du, and A. Plaza, "A3 CLNN: Spatial, spectral and multiscale attention ConvLSTM neural network for multisource remote sensing data classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 2, pp. 747–761, Feb. 2022.

[34] L. Mou and X. X. Zhu, "Learning to pay attention on spectral domain: A spectral attention module-based convolutional network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 1, pp. 110–122, Jan. 2020.

[35] M. Zhu, L. Jiao, F. Liu, S. Yang, and J. Wang, "Residual spectral-spatial attention network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 449–462, Jan. 2021.

[36] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[37] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," 2021, *arXiv:2103.14030*.

[38] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, "Segmenter: Transformer for semantic segmentation," 2021, *arXiv:2105.05633*.

[39] X. Chen, B. Yan, J. Zhu, D. Wang, and H. Lu, "Transformer tracking," 2021, *arXiv:2103.15436*.

[40] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, and N. Houlsby, "An image is worth 16 × 16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[41] W. Dong, T. Zhang, J. Qu, S. Xiao, T. Zhang, and Y. Li, "Multibranch feature fusion network with self- and cross-guided attention for hyperspectral and LiDAR classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–12, Jun. 2022, Art. no. 5530612.

[42] Z. Xue, X. Tan, X. Yu, B. Liu, A. Yu, and P. Zhang, "Deep hierarchical vision transformer for hyperspectral and LiDAR data classification," *IEEE Trans. Image Process.*, vol. 31, pp. 3095–3110, Apr. 2022.

[43] X. He, Y. Chen, and Z. Lin, "Spatial-spectral transformer for hyperspectral image classification," *Remote Sens.*, vol. 13, 2021, Art. no. 498.

[44] Y. Qing, W. Liu, L. Feng, and W. Gao, "Improved transformer net for hyperspectral image classification," *Remote Sens.*, vol. 13, no. 11, 2021, Art. no. 2216.

[45] L. Sun, G. Zhao, Y. Zheng, and Z. Wu, "Spectral-spatial feature Tokenization transformer for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, Jan. 2022, Art. no. 5522214.

[46] C. F. Chen, Q. Fan, and R. Panda, "Crossvit: Cross-attention multi-scale vision transformer for image classification," 2021, *arXiv:2103.14899*.

[47] A. Jaegle, F. Gimeno, A. Brock, A. Zisserman, and J. Carreira, "Perceiver: General perception with iterative attention," 2021, *arXiv:2103.03206*.

[48] H. Liu, X. Jiang, X. Li, Z. Bao, D. Jiang, and B. Ren, "Nommer: Nominate synergistic context in vision transformer for visual recognition," 2021, *arXiv:2111.12994*.

[49] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, Aug. 2004.

[50] K. Wang, H. Rui, and S. Qian, "Spectral-spatial hyperspectral image classification using extended multi attribute profiles and guided bilateral filter," in *Proc. Int. Conf. Comput. Sci. Mech. Autom.*, 2015, pp. 235–239.

[51] M. Kanthi, T. H. Sarma, and C. S. Bindu, "A 3D-deep CNN based feature extraction and hyperspectral image classification," in *Proc. IEEE India Geosci. Remote Sens. Symp.*, 2020, pp. 229–232.

[52] D. Hong et al., "SpectralFormer: Rethinking hyperspectral image classification with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–15, Nov. 2022, Art. no. 5518615.

Yuwen Zhang (Student Member, IEEE) received the B.S. degree in electrical engineering and automation from the Hunan Institute of Science and Technology, Yueyang, China, in 2020, where he is currently working toward the M.S. degree in information and communication engineering.

His research interests include image processing, classification of multisource remote sensing data, and object detection.

Yishu Peng (Member, IEEE) received the B.S., M.S., and Ph.D. degrees from Northeastern University, Shenyang, China, in 2009, 2011, and 2017, respectively, all in mechanical design and theory.

From 2017 to 2019, he was with the School of Mechanical and Engineering, Hunan Institute of Science and Technology, Yueyang, China, and since 2019, he has been with the School of Information Science and Technology. His research interests include the image processing, object detection, and target tracing.

Bing Tu (Member, IEEE) received the M.S. degree in control science and engineering from the Guilin University of Technology, Guilin, China, in 2009, and the Ph.D. degree in mechatronic engineering from the Beijing University of Technology, Beijing, China, in 2013.

From 2015 to 2016, he was a Visiting Researcher with the Department of Computer Science and Engineering, University of Nevada, Reno, NV, USA, which is supported by the China Scholarship Council. Since 2018, he had been an Associate Professor with the School of Information Science and Engineering, Hunan Institute of Science and Technology, Yueyang, China. He is currently a Full Professor with the School of Information Science and Engineering, Hunan Institute of Science and Technology, Yueyang, China. His research interests include sparse representation, pattern recognition, and analysis in remote sensing.

Dr. Tu is an Associate Editor of the *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*.

Yaru Liu (Student Member, IEEE) received the B.S. degree in communication engineering from the Lanzhou University of Technology, Lanzhou, China, in 2016. She is currently working toward the M.S. degree in information and communication engineering with the Hunan Institute of Science and Technology, Yueyang, China.

Her research interests include hyperspectral image processing, computer vision, and deep learning.