MSLAENet: Multiscale Learning and Attention Enhancement Network for Fusion Classification of Hyperspectral and LiDAR Data

Yingying Fan, Yurong Qian, Yugang Qin, Yaling Wan, Weijun Gong, Zhuang Chu, and Hui Liu

***Abstract*—The effective use of multimodal data to obtain accurate land cover information has become an interesting and challenging research topic in the field of remote sensing. In this article, we propose a new method, multiscale learning and attention enhancement network (MSLAENet), to implement hyperspectral image (HSI) and light detection and ranging (LiDAR) data fusion classification in an end-to-end manner. Specifically, our model consists of three main modules. First, we design the composite attention module, which adopts self-attention to enhance the feature representations of HSI and LiDAR data, respectively, and cross-attention to achieve cross-modal information enhancement. Second, the proposed multiscale learning module combines self-calibrated convolutions and hierarchical residual structure to extract different scales of information to further improve the representation capability of the model. Finally, the attention-based feature fusion module fully considers the complementary information properties between different modalities and adaptively fuses heterogeneous features from different modalities. To test the performance of MSLAENet, we conduct experiments on three multimodal remote sensing datasets and compare them with the state-of-the-art fusion model, which demonstrates the effectiveness and superiority of the model.**

***Index Terms*—Attention mechanism, fusion classification, hyperspectral image (HSI) and light detection and ranging (LiDAR) data, multiscale feature, self-calibrated convolutions.**

Manuscript received 29 March 2022; revised 20 September 2022 and 16 October 2022; accepted 6 November 2022. Date of publication 10 November 2022; date of current version 30 November 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61966035, in part by the National Science Foundation of China under Grant U1803261, in part by the Xinjiang Uygur Autonomous Region Innovation Team under Grant XJEDU2017T002, in part by the Autonomous Region Science and Technology Department International Cooperation Project under Grant 2020E01023, in part by Tianshan Innovation Team Plan Project of Xinjiang Uygur Autonomous Region under Grant 202101642, in part by the Xinjiang University of Finance and Economics scientific research fund project under Grant 2017XYB015, and in part by Sichuan Regional Innovation Cooperation Project under Grant 2020YFQ0018. *(Corresponding author: Yurong Qian.)*

Yingying Fan, Weijun Gong, and Hui Liu are with the College of Information Science and Engineering, Xinjiang University, Urumqi 830008, China, also with the Key Laboratory of Signal Detection and Processing in Xinjiang Uygur Autonomous Region, Urumqi 830014, China, and also with the Key Laboratory of Software Engineering, Urumqi 830008, China (e-mail: fyy0327@stu.xju.edu.cn; 1617812854@qq.com; liuhui@stu.xju.edu.cn).

Yurong Qian, Yugang Qin, Yaling Wan, and Zhuang Chu are with the College of Software, Xinjiang University, Urumqi 830008, China, also with the Key Laboratory of Signal Detection and Processing in Xinjiang Uygur Autonomous Region, Urumqi 830014, China, and also with the Key Laboratory of Software Engineering, Urumqi 830008, China (e-mail: qyr@xju.edu.cn; 107552104366@stu.xju.edu.cn; wyl@stu.xju.edu.cn; 1617812854@qq.com).

Digital Object Identifier 10.1109/JSTARS.2022.3221098

I. Introduction

LAND cover classification has important applications in the fields of agricultural monitoring, production layout, and urban planning. Compared with traditional ground survey methods, remote sensing technology can acquire land cover information with a broader perspective and faster speed [1]. With the continuous development of remote sensing sensor technology, multiplatform and multimodal remote sensing data for the same area are continuously generated, making it possible to use multisensor data to jointly describe land cover information. Different sensors can provide remote sensing data with different advantages and complementary characteristics, for example, hyperspectral image (HSI) can achieve simultaneous acquisition of spatial and spectral information for the observed targets and are now widely used in land cover classification tasks, but the strong spectral resolution and weak spatial resolution characteristics presented by HSI to some extent limit a large number of applications oriented to spatial resolution and sensitive characteristics of radiation information [2]. Unlike HSI, light detection and ranging (LiDAR) images are acquired through active sensing techniques, are less subject to atmospheric interference, contain rich height and shape information, and can provide complementary information for HSI images [3]. Therefore, the land cover classification effect can be further improved by combining different modal remote sensing data and making full use of the complementary advantages of multisource information.

In order to extract effective information from multisource remote sensing data and perform classification, scholars have proposed many methods. Filtering approaches is an early and commonly used method to fuse multimodal remote sensing data for classification, which effectively extracts contextual and spatial features from remote sensing images by reducing redundant spatial information, and uses these features to complete the classification task [4]. Typical filtering approaches algorithms are morphological profiles (MPs) [5], attribute profiles (APs) [6], and extinction profiles (EPs) [7]. Liao [8] et al. used MPs to extract HSI and LiDAR data features, used support vector machines for feature-level classification, and finally joint decision-level fusion for classification. Ghamisi [9] et al. used Aps to extract spatial features from HSI and LiDAR data, and achieved better classification results by concatenating the extracted features. To further improve the classification effect, Ghamisi [10] et al. proposed to use EPs to extract spatial features in HSI and elevation information in LiDAR data. Another more commonly used method is based on subspace learning, Liao et al. [11] proposed a graph-based subspace embedding method to combine spectral, spatial, and elevation information for classification; Yan et al. [12] proposed an angle-based discriminant analysis approach based on Euclidean, classifying multisource features by composite kernel-based subspace learning. Hong et al. [13] proposed a cross-modal feature learning framework based on a common subspace to achieve joint classification of HSI and multispectral image (MSI). Although these traditional shallow models are successfully applied in multimodal remote sensing classification, the above methods usually choose shallow algorithms such as support vector machines [14] and random forests [15] as classifiers, and their intrinsic relationships are more complex due to the different ways of imaging for multimodal remote sensing data, which makes it difficult for shallow algorithms to use these features in an integrated way, causing the traditional feature-level fusion classification methods to exhibit some shortcomings. For example, HSI contain very complex information and have nonlinear characteristics, the traditional feature extraction methods destroy the original spatial and spectral structure in the image, making it difficult to extract these features comprehensively, thus ignoring a large amount of implicit and effective information. In addition, the number of features of HSI is large, and if combined with the features of multisource remote sensing data, it will lead to an even larger feature scale.

In recent years, deep learning techniques have been widely used in the field of computer vision and have shown excellent feature extraction capabilities, so some researchers have started to apply them to the field of remote sensing [16], [17]. Many studies have shown that deep learning has achieved remarkable results in the fields of single-source remote sensing image (e.g., HSI, MSI, LiDAR, etc.) classification [18], [19], [20], semantic segmentation [21], [22], and super-resolution [23]. In order to fully utilize the complementary information of multimodal remote sensing images, many excellent deep learning methods have been proposed, and typical models include convolutional neural networks (CNN), recurrent neural networks, and autoencoder networks. Among them, since CNN can better extract features from 2-D image data, many researchers have adopted CNN as the backbone model for classification of multimodal remote sensing data. Chen [24] et al. first used two-branch CNN to extract features from MSI/HSI and LiDAR data, respectively, and stitched these heterogeneous features to achieve joint classification of multisource remote sensing data. Based on the two-branch network, in order to achieve joint HSI-LiDAR classification, Feng [25] et al. introduced residual connection and adaptive fusion mechanism in the network. Xu [26] et al. designed a CNN with cascaded blocks, Hang [27] et al. proposed a coupled CNN network to reduce model complexity and improve classification performance through weight sharing. Zhang [28] et al. designed an unsupervised feature extraction framework based on CNN, some scholars also introduced 3DCNN in the HSI branch to better extract the spatial spectral information of HSI [29]. Different from the CNN approach, Hong [30] et al. built a deep network based on autoencoder for classification of hyperspectral and LiDAR data. Although these methods can achieve better classification results compared to shallow algorithms, they still suffer from limited feature extraction and insufficient utilization of complementary information. To solve this problem, many methods based on attention mechanisms have been proposed. FusAtNet [31] uses self-attention to enhance the feature representation of each modal data and cross-attention to assign the spatial mask of LiDAR to HSI, enhancing the spatial feature representation of HSI by LiDAR data. A3 CLNN [32] constructs a spatial, spectral, and multiscale attention mechanism and designed an efficient fusion strategy to fully fuse multisource data features.

However, there are still some problems with HSI-LiDAR fusion classification. First, in complex scenes, multiscale information is crucial to the representation of multimodal data, while existing studies pay less attention to multiscale information and have limitations in extracting multiscale features in remote sensing images. Second, how to further accurately extract spectral and spatial information from HSI and LiDAR data by using attention mechanism and fully utilize the spatial information of LiDAR data in cooperation with HSI, it remains a question to be further studied. More importantly, the feature fusion (FF) approach based on simple feature stitching often fails to achieve better classification performance because it ignores the complementarity between multimodal data, and this approach will further increase the feature dimensionality, which may lead to dimensional disaster.

To address these problems, this article proposes the multiscale learning and attention enhancement network (MSLAENet), specifically, the network adopts a two-branch CNN structure, based on self-calibrated convolutions and hierarchical residual structure to build a multiscale learning (MSL) module to extract spectral and spatial information at different scales, which enriches the feature representation of multisource data, two attention mechanisms (self-attention and cross-attention) in the network enhance the spatial and spectral feature representation and intermodal information interaction in each branch, the attention-based FF module can better achieve fusion classification of HSI and LiDAR data. Experiments are conducted on three real hyperspectral and LiDAR datasets, and the effectiveness of the method is demonstrated by comparison with existing models.

In summary, the main contributions of this article are summarized as threefold.

1) To improve the classification performance of multimodal remote sensing data by using multiscale information, a MSL module is constructed by combining self-calibrated convolutions and hierarchical residual networks, which can extract spatial and spectral information of different receptive fields and enhance the multiscale information representation capability of the whole model.
2) Considering the rich spectral and spatial information in HSI and LiDAR data, composite attention (CA) is constructed to obtain enhanced representations of spectral and spatial. Specifically, the spatial information representations in LiDAR data and spectral information representations in HSI are adaptively learned and enhanced by self-attention, and cross-attention is used to achieve

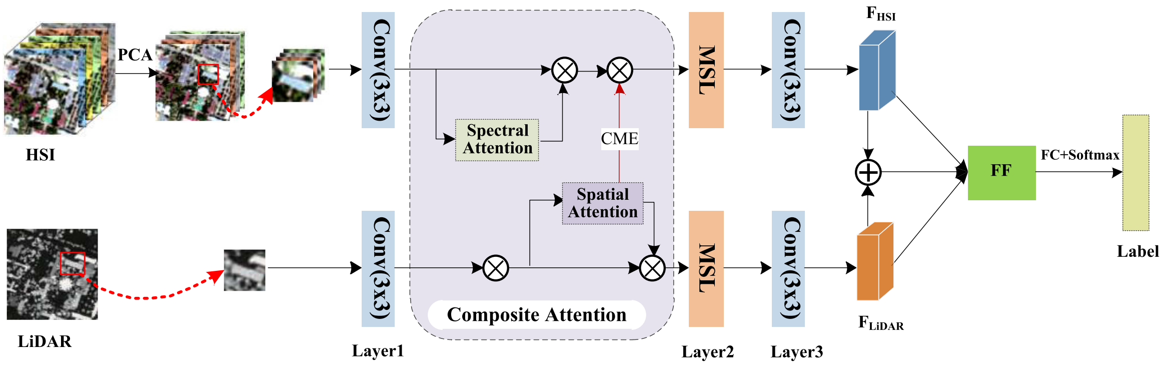


Fig. 1. Architecture of the proposed dual-channel MSLAENet model.

cross-modal information enhancement (CME) to achieve complementary utilization of different modal information.

3) A new attention-based FF module is proposed to take location information into account and fully consider the information complementarity between two modal data to achieve efficient fusion of multimodal features.

II. Methods Description

A. Architecture Overview

The overall framework of the proposed MSLAENet is shown in Fig. 1, which uses CNN as the backbone network and contains three layers. Unlike the traditional CNN network, we build a multiscale feature learning module based on self-calibrated convolution in the second layer of the network to extract multiscale information, and in order to obtain enhanced spectral and spatial feature representations, we add an attention mechanism to the network and realize the information interaction between modalities by CME method. In addition, to avoid the problem of high feature dimensionality and insufficient FF caused by traditional FF using concatenation operation, we propose a novel attention-based FF method that can effectively fuse heterogeneous features between different modalities. It is worth noting that we set the padding and stride parameters of the convolution operation to keep the feature map size constant during the operation, and add batch normalization and rectified linear unit (ReLU) after the convolution operation in each of the three layers in turn for accelerating training and learning nonlinear representation.

The input of the network consists of HSI and LiDAR covering the same area, and fixed-size image blocks are selected for network training and testing centered on the pixel points to be classified, and the input of HSI and LiDAR branches can be expressed as $X_h \in R^{h\times w\times b_h}$ and $X_l \in R^{h\times w\times b_l}$, the b_h and b_l denote the number of HSI and LiDAR bands, respectively, and h and w represent the height and width of the input image, respectively. Considering the redundancy of high-dimensional spectral information in the hyperspectral data, principal component analysis is used to reduce the dimensionality of the input data, and the reduced-dimensional input can be expressed as $X_h' \in R^{h\times w\times b_p}$, p is the number of bands after dimensionality reduction.

B. CA Module

Inspired by the human vision system, attention mechanisms have been introduced into computer vision systems, and more and more deep learning models based on attention mechanisms have been proposed and improved feature representation in many research areas (e.g., image classification, object detection, image generation, etc.) [33]. With the rapid development of deep learning techniques, attention mechanisms are now widely used in remote sensing image classification tasks [34], [35], [36]. In this article, the attention mechanism will be used to guide deep learning networks to learn more accurate feature representations.

1) Spectral Attention for HSI: HSI contain rich spectral information, and the effective use of this spectral information can improve the performance of multimodal remote sensing land cover classification. At present, a number of attention mechanisms have been used to enhance the spectral representation of HSI. But most of them only consider the internal channel information, thus ignoring the location information. However, in the HSI classification task, location information is crucial to capture the structure of the objects. Channel attention with embedded location information, coordinate attention, is a simple and efficient attention model that captures not only cross-channel information, but also orientation-aware and location-sensitive information, which helps the model to locate and identify objects of interest more accurately [37]. To obtain a more discriminative representation of spectral information in HSI, we use a spectral attention model with embedded location information in the HSI branch. As shown in Fig. 2, this module uses global average pooling to encode features along the horizontal and vertical coordinate directions for each channel of the input features, allowing the attention module to capture not only channel information, but also orientation-aware and position-aware information, which in turn improves the classification results. The coordinate attention consists of two steps: coordinate information embedding and coordinate attention generation.

The first step is coordinate information embedding. Assume that the output obtained from the HSI branch after layer1 is $X_{h1} \in R^{h\times w\times c_1}$. The average pooling kernel of size $(h, 1)$ and

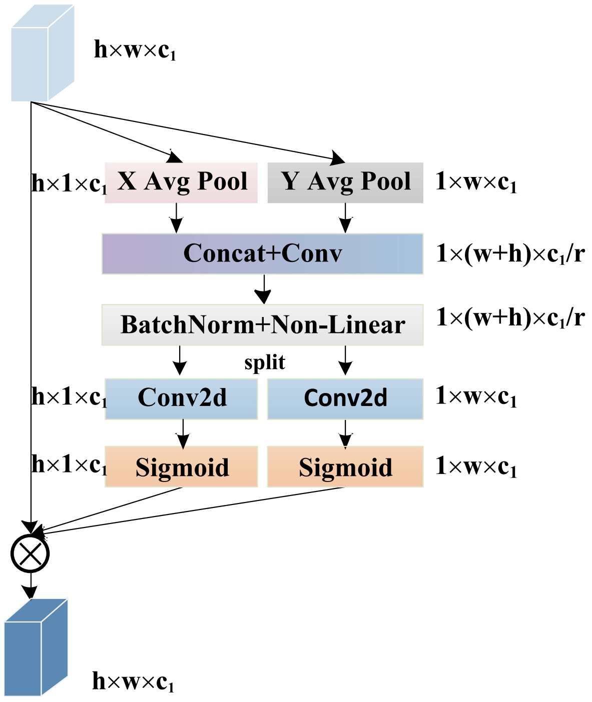


Fig. 2. Structure of spectral attention module.

(1, w) is used to encode each channel along the horizontal and vertical directions, respectively, so that the output after encoding along the horizontal direction is

$$z^h = \sum_{0\le i<w} x_{h1}^c(h,i). \tag{1}$$

Output after encoding along the vertical direction is

$$z^w = \sum_{0\le j<h} x_{h1}^c(j,w). \tag{2}$$

The second step is coordinate attention generation. To make better use of the representation with global receptive field and precise location information generated by the coordinate information embedding module, the coordinate attention generation module is designed to generate channel attention map. First, connecting z^h and z^w, then use the shared 1×1 convolution operation F_1 to perform the feature transformation.

$$f = \delta\left(F_1\left([z^h, z^w]\right)\right). \tag{3}$$

"$[\cdot,\cdot]$" represents the concatenate operation, δ is the nonlinear activation function. $f \in R^{c/r\times(h+w)}$ is the intermediate feature map that encodes spatial information in both the horizontal direction and the vertical direction. r is the reduction ratio for controlling the block size. f is then split along the spatial dimension into $f^h \in R^{c/r\times h}$ and $f^w \in R^{c/r\times w}$, by using two 1×1 convolution F_h and F_w to expand the number of channels of f^h and f^w by r times, so that they are the same number of channels as x_{h1}

$$g^h = \delta\left(F_h\left(f^h\right)\right) \tag{4}$$

$$g^w = \delta\left(F_w\left(f^w\right)\right). \tag{5}$$

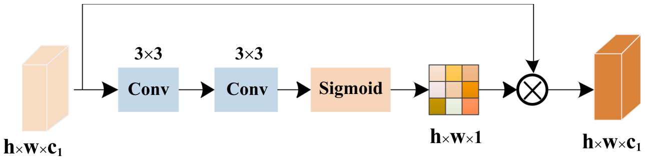


Fig. 3. Structure of spatial attention module.

"δ" is the sigmoid function. Finally, g^h and g^w are used as attention weights to calibrate the weight of input x_{h1}. The final output is

$$Y = x_{h1} \times g^h \times g^w. \tag{6}$$

2) Spatial Attention for LiDAR Data: LiDAR data contains rich elevation information and can convey rich information in the spatial domain. In this article, we use spatial attention to generate spatial attention weights to enhance the feature representation of LiDAR branch. Considering that in the remote sensing image classification task, the pixels to be classified are often more correlated with their surrounding pixels due to the limitation of image resolution, unlike the general spatial attention mechanism that uses global pooling operation to obtain the attention map, we use two consecutive convolution operations to obtain more accurate spatial attention weights. The adopted spatial attention structure diagram is shown in Fig. 3, assuming that the output obtained by LiDAR branch after layer1 is$X_{l1} \in R^{h\times w\times c_1}$, the spatial attention module consists of two convolution operations and a sigmoid function, which first uses two 3×3 convolution to generate a nonnormalized attention map of size $h\times w\times 1$ and then use the sigmoid function to generate the attention weight map, and finally achieve feature enhancement for input x_{l1} of the module by residual skip connections, the process can be formulated as

$$w = \delta\left(f_1\left(f_2\left(x_{l1}\right)\right)\right) \tag{7}$$

$$x'_{l1} = x_{l1} \times w \tag{8}$$

where x_{l1} denotes the output of the spatial attention module, $f_1(-)$, $f_2(-)$ and "δ" are two convolution operations and sigmoid function, respectively.

3) Cross-Modal Enhancement: The rich spatial information in LiDAR data can assist HSI to obtain more accurate classification results. Therefore, we enrich the spatial information of HSI through this cross-modal spatial attention enhancement mechanism by assigning the attention weights obtained from the LiDAR branch to the HSI branch. Thus, after the CA module, the output of the HSI branch can be expressed as

$$x'_{h1} = Y \times w. \tag{9}$$

C. MSL Module

1) Self-Calibrated Convolutions: Traditional CNNs are limited by the size of predefined convolutional kernels and lack large receptive fields, which make it difficult to capture enough high-level semantic information in remote sensing images, thus

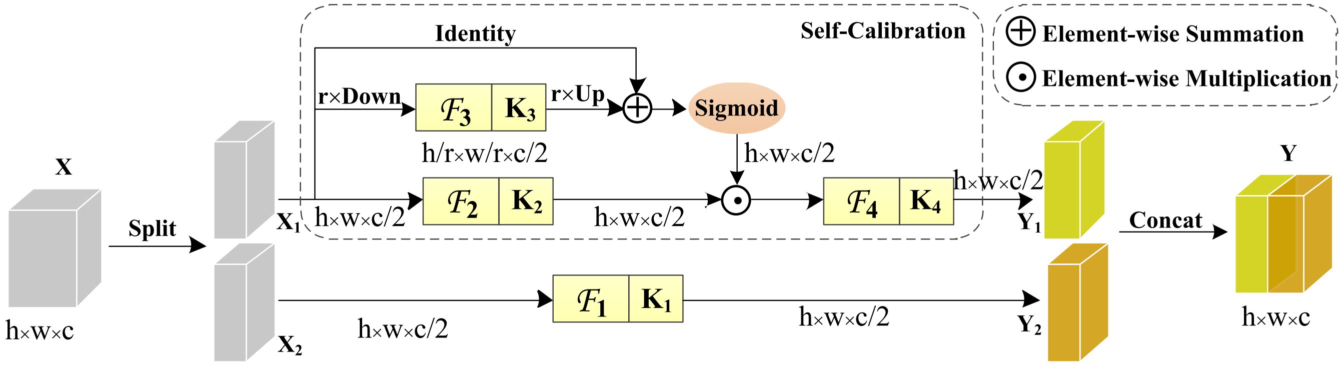


Fig. 4. Illustration of the self-calibrated convolution.

leading to less discriminative feature maps [38]. To obtain more discriminative feature representations, Liu [39] et al. proposed self-calibrated convolutions, which differs from the traditional convolutional uniformly performing convolutional operations on the original input, self-calibrated convolutions perform convolutional features transformation on the input data in two different scale spaces, that is, the original scale space and the downsampled latent space with larger receptive fields. This approach allows each spatial location to adaptively encode contextual information in distant regions, thus breaking the tradition of performing convolution in small regions (e.g., 3 × 3) to produce more discriminative features.

The workflow of the self-calibrated convolutions is illustrated in Fig. 4. First, the input feature map $X \in R^{c \times h \times w}$ is split into X_1 and X_2 with size of $c/2 \times h \times w$, the convolutional transformations are performed on the pairs X_1 and X_2 in the self-calibrated branch and the traditional convolutional branch, respectively, to collect different types of contextual information. Then, given four filters $\{K_1, K_2, K_3, K_4\}$, in the self-calibrated branch, using $\{K_2,\ K_3,\ K_4\}$ to perform the self-calibration operation on X_1 to obtain Y_1; in the conventional convolution branch, use K_1 to performs a simple convolution operation X_2 to obtain $Y_2 = f_1(X_2) = X_2 * K_1$. Finally, the Y_1 and Y_2 are concatenated as the final input Y. The self-calibration process is described as follows.

Given the input X_1, we implement downsampling using averaging pooling, expanding the receptive field at each spatial location.

$$T_1 = \text{AvgPool}_r\,(X_1)\,. \tag{10}$$

r represents the downsampling rate and strides in the pooling operation. Next, using K_2 perform feature transformation on T_1, and use the bilinear interpolation operator $Up(-)$ to perform upsampling by r times, the feature map is restored to the original scale size

$$X_1' = Up\,(f_2\,(T_1)) = Up\,(T_1 * K_2)\,. \tag{11}$$

Then, the self-calibration operation can be described as follows:

$$Y_1' = f_3\,(X_1) * \delta\,(X_1 + X_1') = (X_1 * K_3)\,\delta\,(X_1 + X_1') \tag{12}$$

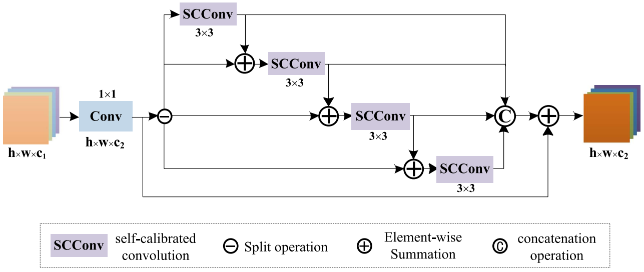


Fig. 5. Illustration of the MSL module.

where "·" is element-wise multiplication, and $\delta(-)$ represents the sigmoid activation function. Therefore, the final result of the self-calibrated branch Y_1 can be written as

$$Y_1 = f_4\,(Y_1') = Y_1' * K_4. \tag{13}$$

2) MSL Module: Different land cover types in remote sensing images exist at different scale sizes, and representing features from multiple scales is crucial for multimodal remote sensing image classification tasks. However, classification models that use uniform scale to extract features can no longer meet the demand for multiscale information for classification tasks. Inspired by the Res2Net [40], the hierarchical residual structure can represent multiscale features at a finer granularity and increase the receptive field of the network. Self-calibrated convolution implements feature transformation from different scale spaces and can obtain a rich feature representation; therefore, this article combines the idea of hierarchical residuals and self-calibrated convolution to build a MSL module.

The structure of the MSL module is shown in Fig. 5. Taking the HSI branch as an example, Let x'_{h1} and x_{h2} denote the input and output of the MSL Module, for the input x'_{h1}, first go through 1×1 convolution for dimensional transformation to get x''_{h1}, then, the feature map x''_{h1} is equally split into m feature subsets by channel, denoted as x''_{h1i}, where $i \in \{1, 2, ..., m\}$, so that each feature subset x''_{h1i} has the same feature dimension, and each x''_{h1i} corresponds to a self-calibrated convolution operation K_i, and the output after the K_i transformation is defined as y_i, in order to obtain a larger receptive field, in addition to x''_{h1i}, we add $y_(i-1)$ to x''_{h1i}, and then fed into K_i, so that y_i can be

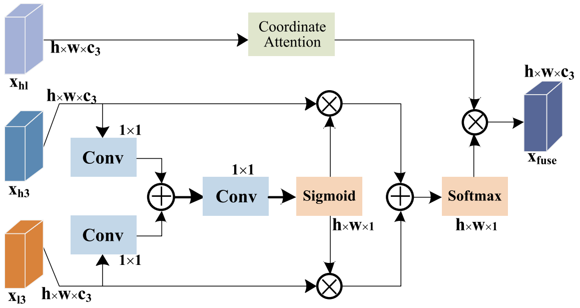


Fig. 6. Illustration of the FF module.

described as

$$y_i = \begin{cases} K_i\left(x''_{h1i}\right), i = 1 \\ K_i\left(x''_{h1i} + y_{i-1}\right), 1 < i \leq m. \end{cases} \quad (14)$$

Finally, the outputs y_i obtained from each layer are joined together to obtain Y'

$$Y' = [y_1, y_2, ..., y_m]. \quad (15)$$

"[-]" means concatenate operation, to avoid the problem of gradient disappearance in the network, we add a residual connection to the 1×1 convolution, map features x''_{h1} obtained before the hierarchical residual learning module are transferred to the deeper layers of the network. Thus, the output of the MSL module x_{h2} can be expressed as

$$x_{h2} = x''_{h1} + Y'. \quad (16)$$

D. FF Module

After obtaining the feature representations of HSI and LiDAR data, how to combine them for classification tasks remains a critical issue. Most existing approaches choose to use the concatenate operation to aggregate them together, however, this approach not only increases the feature dimensionality, but also ignores the contextual information, making the fusion ineffective. Inspired by the visual attention mechanism, we propose an attention-based FF module, as shown in Fig. 6. The FF module contains three inputs, the x_{h3} and x_{l3} denote the HSI features and LiDAR data features obtained after layer3, respectively. x_{hl} is the result after doing element-wise summation operation on HSI and LiDAR data features. As mentioned in Section B, location information is crucial to the multimodal remote sensing image classification task, so we use coordinate attention to x_{hl} to perform feature enhancement. Considering that the direct summation of two feature maps cannot maximize the interaction between feature maps, inspired by attentional FF [41], in order to achieve complementary utilization of HSI and LiDAR data features in the FF stage, we use two 1×1 convolution operations f_1 that f_2, respectively, to reduce the dimension of x_{h3} and x_{l3} to half, and then sum the two modal features in this low-dimensional feature space, followed immediately by using a 1×1 convolution to change the dimension of the feature map to 1, and then use the sigmoid activation function to obtain the non-normalized attention map, they are multiplied with the original input, respectively, and added. So far, we can obtain the feature representation x'_{hl} that fully considers the relationship between HSI and LiDAR data features, finally, the features are normalized by the softmax function and multiplied with $CA(x_{hl})$ to obtain the fused features, which contain location information and maximize feature interaction, This process can be formulated as:

$$w = \delta\left(f_3\left(f_1\left(x_{h3}\right) + f_2\left(x_{l3}\right)\right)\right) \quad (17)$$

$$x'_{hl} = w \times x_{h3} + (1 - w) \times x_{l3} \quad (18)$$

$$x_{fuse} = \theta\left(x'_{hl}\right) \times CA\left(x_{hl}\right) \quad (19)$$

where $\delta(-), \theta(-)$ are the sigmoid function and the softmax function, respectively. $CA(-)$ is the coordinate attention.

TABLE I
NUMBER OF TRAINING AND TEST SAMPLES IN EACH CLASS OF THE HOUSTON DATA

Class no.	Class name	Training	Test
1	Healthy grass	198	1053
2	Stressed grass	190	1064
3	Synthetic grass	192	505
4	Tree	188	1056
5	Soil	186	1056
6	Water	182	143
7	Residential	196	1072
8	Commercial	191	1053
9	Road	193	1059
10	Highway	191	1036
11	Railway	181	1054
12	Parking lot 1	192	1041
13	Parking lot 2	184	285
14	Tennis court	181	247
15	Running track	187	473
-	Total	2832	12197

III. EXPERIMENTS AND ANALYSIS

A. Data Description

To test the effectiveness of our proposed MSLAENet, we conducted experiments on three widely used hyperspectral and LiDAR fusion datasets.

1) Houston Dataset: this dataset was acquired by the Center for Airborne Laser Mapping, funded by the National Science Foundation at the University of Houston, in June 2012 in the University of Houston campus and surrounding area [42]. Both HSI and LiDAR modal data were included, with a band count of 144 and 1, respectively, both containing 349 × 1905 pixels with a spatial resolution of 2.5 m. There are 15 different classes and the pseudocolor images of HSI, grayscale maps of LiDAR data and ground truth maps are shown in Fig. 10(a)–(c), respectively. Table I shows the detailed classes and the number of samples used for training and testing for each category.

2) Trento Dataset: This dataset also contains similarly one HSI and one LiDAR data, collected from a rural area south of the city of Trento, Italy. HSI data are collected by the AISA Eagle sensor with 63 bands; LiDAR data are collected by the Optech ALTM 3100EA sensor. Both types of data contain 166 × 600 pixels with a spatial resolution of 1 m, containing

TABLE II
NUMBER OF TRAINING AND TEST SAMPLES IN EACH CLASS OF THE TRENTO DATA

Class no.	Class name	Training	Test
1	Apple trees	129	4034
2	Buildings	125	2903
3	Ground	105	479
4	Wood	154	9123
5	Vineyard	184	10501
6	Roads	122	3174
–	Total	819	30214

TABLE III
NUMBER OF TRAINING AND TEST SAMPLES IN EACH CLASS OF THE MUUFL DATA

Class no.	Class name	Training	Test
1	Trees	150	23246
2	Mostly Grass	150	4270
3	Mixed Ground Surface	150	6882
4	Dirt and Sand	150	1826
5	Road	150	6687
6	Water	150	466
7	Building Shadow	150	2233
8	Building	150	6240
9	Sidewalk	150	1385
10	Yellow Curb	150	183
11	Cloth Panels	150	269
–	Total	1650	53687

a total of six different classes. Fig. 11 in (a), (b), and (c) shows the pseudocolor image of HSI, the grayscale map of LiDAR data, and the ground truth map, respectively. Table II shows the detailed categories and the number of samples used for training and testing for each category.

3) MUUFL Dataset: This data was collected over the University of Southern Mississippi Gulf Park, both the HSI and LiDAR data contain 325 × 220 pixels. HSI initially contained 72 bands, however, initial and final four bands are removed due to noise issues, the remaining 64 bands were used for the experiment, and the LiDAR data contained two bands. There are 11 different classes and the pseudocolor images of HSI, grayscale maps of the first band of LiDAR data and ground truth maps are shown in Fig. 12(a)–(c), respectively. Table III shows the detailed classes and the number of samples used for training and testing for each category.

B. Parameter Tuning

Our network is implemented in the Pytorch framework, all experiments of this article were conduced on a person computer configured with Intel Xeon W- 2133 CPU, 32 GB RAM, NVIDIA GeForce RTX 2080 graphics card and Windows 10. In the model training process, Adam algorithm is used to optimize our network, and cross-entropy is used as the loss function of the network. Meanwhile, we choose three commonly used evaluation metrics to assess the classification performance, namely Overall accuracy (OA), Average accuracy (AA), and Kappa coefficient.

TABLE IV
OVERALL ACCURACY WITH DIFFERENT VALUES OF M

M	Houston	Trento	MUUFL
{8,16,32}	94.62	98.11	91.49
{16,32,64}	**96.47**	**99.03**	92.14
{32,64,128}	95.80	98.97	**92.31**
{64,128,256}	95.35	98.80	92.18

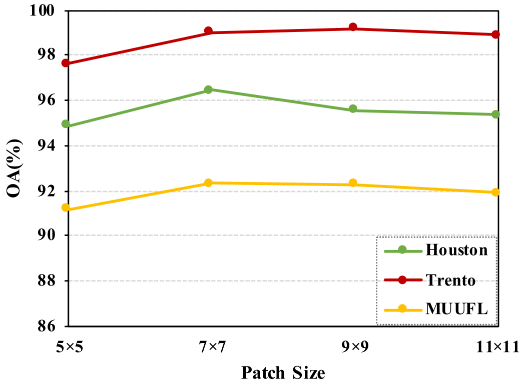


Fig. 7. Overall accuracy with different input patch size.

The setting of deep learning network parameters has a great influence on the model performance. In this section, we will discuss the performance of the model by changing the parameters of the network, which include the input patch size ($s \times s$), the number of principal components (P), the number of feature maps (M), the learning rate (lr). We set the default values of s, P, M, and lr to 7, 30, {16, 32, 64}, and 0.001, respectively, when analyzing the effect of a parameter, default values were taken for the remaining parameters, except as otherwise noted. We empirically set the default values of batch size and epoch to 64 and 200, respectively.

1) Analysis on the Number of Feature Maps: As shown in Fig. 1, the proposed model contains three layers, where layer 1 and layer 3 are CNNs and layer 2 is a MSL module. The parameter M represents the number of feature graphs obtained after passing through each layer, we determine the optimal number of feature maps by using four different combinations of M values. As shown in Table IV, For Houston, Trento, and MUUFL dataset, the optimal number of feature mappings are {16, 32, 64}, {16, 32, 64}, and {32, 64, 128}, respectively.

2) Analysis on the Input Patch Size: Different input patch size contains different amount of information, in order to evaluate the impact of this parameter on the model performance, we compare the classification results of different input patch size on three datasets. For all datasets, we set M as their optimal number of feature mappings, and we fixed the other parameters as default values, and considering that too large patch size will increase the learning time of the network, we selected the value of s from the candidate set {5, 7, 9, 11} to evaluate the impact of this parameter. As can be seen in Fig. 7, the size of the input patch has a significant impact on the model performance, especially on the Houston dataset. and the optimal patch sizes for Houston, Trento, and MUUFL dataset are 7 × 7, 9 × 9, and 7 × 7, respectively.

3) Analysis on the Number of Principal Components: The number of principal components determines the dimensionality

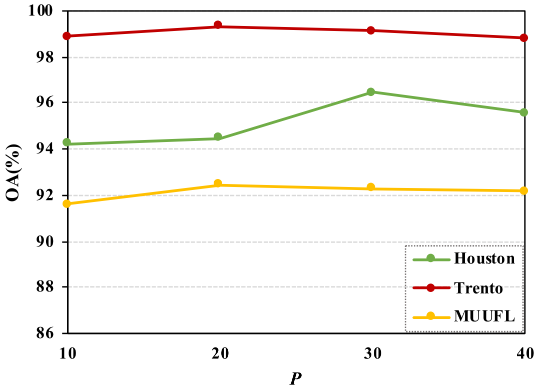


Fig. 8. Overall accuracy with different number of principal components.

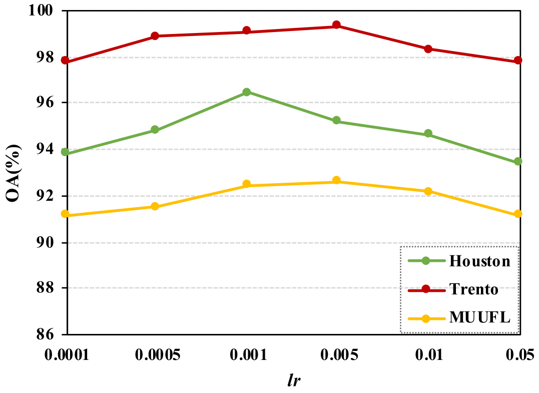


Fig. 9. Overall accuracy with different learning rate.

of the input HSI, and to evaluate the impact of this parameter, we conduct experiments on three datasets. For all datasets, we set M and s to the optimal values, respectively, with other parameters set to default values, and then evaluate the impact of this parameter by selecting the values of P from the candidate sets $\{10, 20, 30, 40\}$. Fig. 8 shows the overall accuracy achieved when setting different values of P for different datasets, and it can be seen that the optimal P values for the Houston, Trento, and MUUFL dataset are 30, 20, and 20, respectively.

4) Analysis on the Learning Rate: The learning rate of the deep learning network can guide the network to adjust the weights of the network through the gradient of the loss function, which has a large impact on the model performance. To evaluate the impact of the learning rate on the performance of MSLAENet, for different datasets, we fix the other parameters as the optimal values and set the learning rate candidates as {0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05} to select the best learning rate by experiment. Fig. 9 reports the overall accuracy achieved when setting different lr values for different datasets. The best learning rate values for Houston, Trento, and MUUFL datasets are 0.001, 0.005, and 0.005, respectively.

C. *Classification Performance*

To highlight the superiority of MSLAENet, we selected seven classification methods for comparison, including two traditional machine learning algorithms SVM [43] and ELM [44], and five state-of-the-art deep learning methods, which are the contextual deep CNN model CDCNN [45], the two-branch CNN model TBCNN [26], the encoder–decoder structure-based fusion network EndNet [30], dual attention-based spectral spatial fusion network FusAtNet [31], and spatial-spectral cross-modal enhancement network S2Enet [46], Among them, CDCNN network is the classical network used for HSI classification. For the conventional methods and CDCNN model, we used LiDAR data and HSI for data layer fusion as the input to the network. For a fair comparison, we used the same training and test sets in all methods.

1) Quantitative Comparison: Tables V–VII show the OA, AA, Kappa, and category accuracies obtained using different methods on the Houston, Trento, and MUUFL datasets, respectively, and the bold values in the tables represent the optimal values of the corresponding rows. From the table, we can draw the following conclusions.

The performance of deep learning-based methods is generally higher than the performance of traditional methods, for example, for Houston, the highest OA value achieved by traditional methods is 5% lower than the lowest OA achieved by deep learning methods, which is due to the stronger feature representation capability of deep learning methods compared to traditional methods, and the fact that traditional methods fuse multimodal data at the data level and then input them into the network for classification, this method cannot effectively fuse the information across modalities.

Among all deep learning-based methods, our proposed network obtains the best classification performance. Specifically, for the Houston, Trento, and MUUFL datasets, we achieved 96.47%, 99.33%, and 92.62% OA, respectively, and the AA and Kappa metrics were higher than the other comparison algorithms. For the Houston dataset, our proposed method achieves a more significant improvement, the OA is 9.55%, 8.49%, 7.95%, 6.49%, and 2.28% higher compared to CDCNN, TBCNN, EndNet, FusAtNet, and S2Enet, respectively. Comparing other methods, it is not difficult to find, the CDCNN method stacks HSI and LiDAR data as input to the network, and this data-level fusion ignores the differences between different modal data and cannot effectively fuse the information of each modality. In TBCNN, the information of each branch cannot be effectively fused by a simple feature cascade; The learning ability of encoder decoder-based feature representation in EndNet is still limited; FusAtNet is the first proposal to be used in multimodal remote sensing classification task using cross-attention approach to achieve enhancement from one modality to another, and S2Enet proposes cross-modal interaction learning before FF to enhance the information representation of each modality. However, all of them do not consider the multiscale information in remote sensing images and lack efficient FF methods. On the one hand, our model fully extracts the spatial and spectral information in multimodal remote sensing data through the attention mechanism and achieves spatial enhancement of HSI data through the cross-modal attention mechanism. On the other hand, the introduction of multiscale information can extract more scale-related information that helps classification.

TABLE V
CLASSIFICATION RESULT OBTAINED BY DIFFERENT METHODS ON HOUSTON DATASET

NO	SVM	ELM	CDCNN	TBCNN	EndNet	FusAtNet	S2ENet	MSLAENet
C1	82.43	83.10	99.32	83.10	81.58	83.10	82.91	**99.48**
C2	82.05	83.70	87.56	84.10	83.65	96.05	**100**	93.31
C3	99.80	**100**	98.80	**100**	**100**	**100**	**100**	99.61
C4	92.80	91.86	97.48	93.09	93.09	93.09	96.88	**97.76**
C5	98.48	98.86	99.81	**100**	99.91	99.43	99.91	97.42
C6	95.10	95.10	99.31	99.30	95.10	**100**	**100**	**100**
C7	75.47	80.04	73.23	92.82	82.65	93.53	95.15	**99.40**
C8	46.91	68.47	88.65	82.34	81.29	92.12	**93.92**	93.39
C9	77.53	84.80	82.34	84.70	88.29	83.63	**91.31**	90.49
C10	60.04	49.13	75.81	65.44	89.00	64.09	**92.95**	91.41
C11	81.02	80.27	72.10	88.24	83.78	90.13	94.69	**99.14**
C12	85.49	79.06	85.39	89.53	90.39	91.93	89.43	**98.98**
C13	75.09	71.58	94.29	92.28	82.46	88.42	83.16	**100**
C14	**100**	99.60	83.62	96.76	**100**	**100**	**100**	**100**
C15	98.31	98.52	99.55	99.79	98.10	99.15	**100**	**100**
OA	80.49	81.92	86.92	87.98	88.52	89.98	94.19	**96.47**
AA	83.37	84.27	89.15	90.11	89.95	94.65	94.69	**97.36**
Kappa	78.98	80.45	85.80	86.98	87.59	89.13	93.69	**96.16**

TABLE VI
CLASSIFICATION RESULT OBTAINED BY DIFFERENT METHODS ON TRENTO DATASET

NO	SVM	ELM	CDCNN	TBCNN	EndNet	FusAtNet	S2ENet	MSLAENet
C1	88.62	95.81	99.76	98.07	88.19	99.54	**99.85**	99.80
C2	94.04	96.97	96.40	95.21	**98.49**	**98.49**	98.17	94.40
C3	93.53	96.66	99.44	93.32	95.19	99.73	**100**	97.61
C4	98.90	99.39	97.75	99.93	99.30	**100**	99.42	99.97
C5	88.96	82.24	97.32	98.78	91.96	99.90	99.65	**100**
C6	91.75	86.52	93.27	89.98	90.14	93.32	90.83	**99.59**
OA	92.77	91.32	97.29	97.92	94.17	99.06	98.54	**99.33**
AA	92.63	92.93	97.32	96.19	93.88	98.50	97.99	**98.56**
Kappa	95.85	90.42	96.39	96.81	92.22	98.75	98.06	**99.11**

TABLE VII
CLASSIFICATION RESULT OBTAINED BY DIFFERENT METHODS ON MUUFL DATASET

NO	SVM	ELM	CDCNN	TBCNN	EndNet	FusAtNet	S2ENet	MSLAENet
C1	98.35	94.89	98.35	98.11	96.86	**98.77**	98.16	**98.77**
C2	55.52	62.23	74.90	**83.38**	72.09	82.27	81.64	81.51
C3	72.89	83.15	82.42	81.77	80.24	**91.64**	90.55	85.70
C4	39.53	57.88	75.66	84.66	73.67	87.99	83.02	**94.46**
C5	80.12	93.33	95.47	96.34	96.56	96.84	94.50	**97.32**
C6	30.92	68.32	80.62	83.36	64.89	87.76	72.14	**92.09**
C7	54.40	47.01	63.93	70.29	66.52	76.50	79.46	**84.08**
C8	86.37	77.58	95.84	**98.77**	95.41	97.92	97.93	97.89
C9	18.61	32.15	62.88	**73.52**	60.45	65.43	65.45	70.61
C10	7.47	0.00	**52.05**	33.15	47.03	18.62	33.40	27.90
C11	0.00	78.29	71.73	60.77	**83.49**	73.34	80.18	82.66
OA	80.74	83.10	88.96	90.85	87.75	91.77	91.68	**92.62**
AA	49.47	63.17	77.62	78.56	76.11	79.74	79.67	**83.00**
Kappa	72.89	77.42	85.67	88.06	84.06	89.28	89.15	**90.35**

In addition, our proposed fusion method will introduce location information and fully consider the relationship between HSI and LiDAR data, which can effectively integrate the complementary information between different modalities and improve the classification accuracy.

2) Visual Comparison: In addition, to better demonstrate the classification performance of different methods, Figs. 10–12 show the classification maps obtained by different classification methods using Houston, Trento, and MUUFL datasets, respectively, and for comparison, we also list the ground truth, in which different colors represent different land cover types. It is obvious that the classification maps obtained by MSLAENet show the fewer error markers, which are more similar to the corresponding ground truth, especially in Houston, where the classification accuracy for categories C1, C7, C11, C12, and C13 far exceeds that of other comparison algorithms, which further validates the advantages of the model in this article.

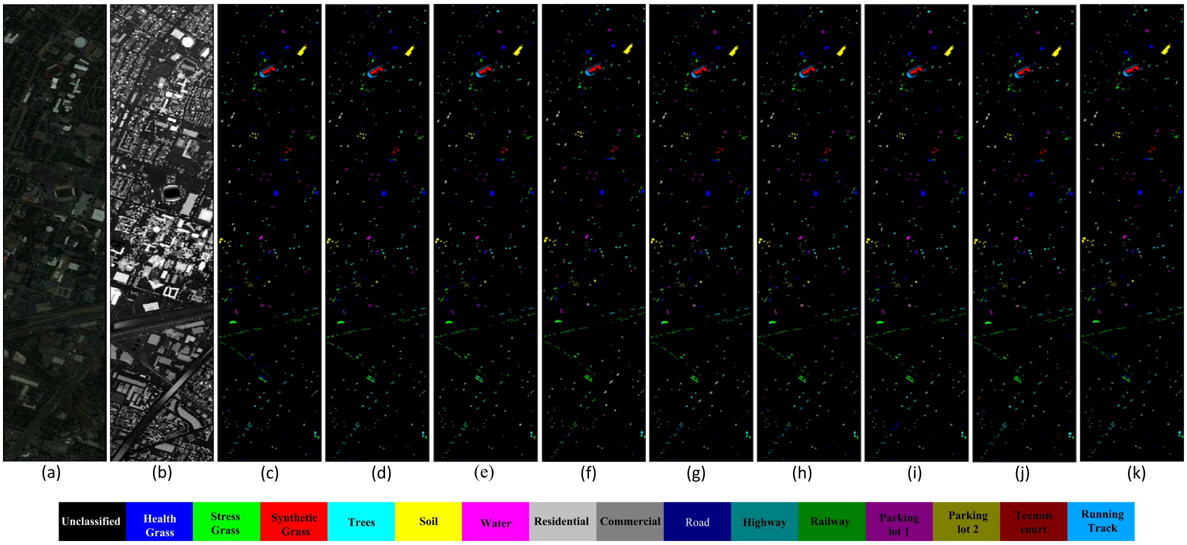


Fig. 10. Houston data visualization and classification maps obtained by different models. (a) False-color image for HSI using bands 60, 40, and 20 as R, G, and B, respectively. (b) Grayscale image for LiDAR data. (c) Ground truth. (d) SVM. (e) ELM. (f) CDCNN. (g) TBCNN. (h) EndNet. (i) FusAtNet. (j) S2Enet. (k) MSLAENet.

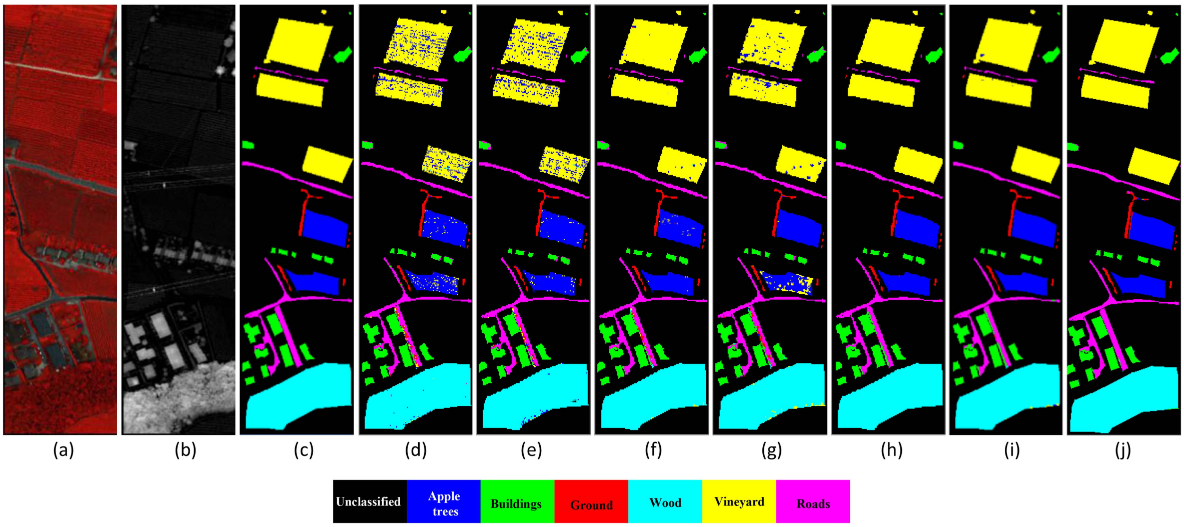


Fig. 11. Trento data visualization and classification maps obtained by different models. (a) False-color image for HSI using bands 40, 20, and 10 as R, G, and B, respectively. (b) Grayscale image for LiDAR data. (c) Ground truth. (d) SVM. (e) ELM. (f) CDCNN. (g) TBCNN. (h) EndNet. (i) FusAtNet. (j) S2Enet. (k) MSLAENet.

3) Computation Time Comparisons: In general, deep learning methods tend to require longer time consumption due to the complex model structure. To quantitatively analyze the computational cost of different methods, we set the training epoch of all methods to 200, and we report their training time and testing time on all datasets in Table VIII. Since the traditional methods have simple models and less time consumption, we ignore them in our report. From this table, it can be seen that more complex datasets tend to require more training time. In addition, the training process of our MSLAENet takes more time, second only to FusAtNet among all compared methods, which is because of the introduction of several attention modules. however, the increase in time is acceptable because our proposed method achieves the best classification accuracy.

D. Ablation Study

In order to further evaluate the performance of each module in MSLAENet, further ablation experiments were carried out. A CNN network with three layers is used as the baseline network, and in the baseline network, we fuse HSI features and LiDAR data features by stacking them. We gradually add CA, MSL, and FF modules to the CNN network, and the impact of each module on the network performance was analyzed by different combinations of modules. Table IX shows the experimental results obtained with different modules and different combinations of modules on different datasets, and the analysis of the experimental results shows that the three modules proposed in this article can improve the classification results to different

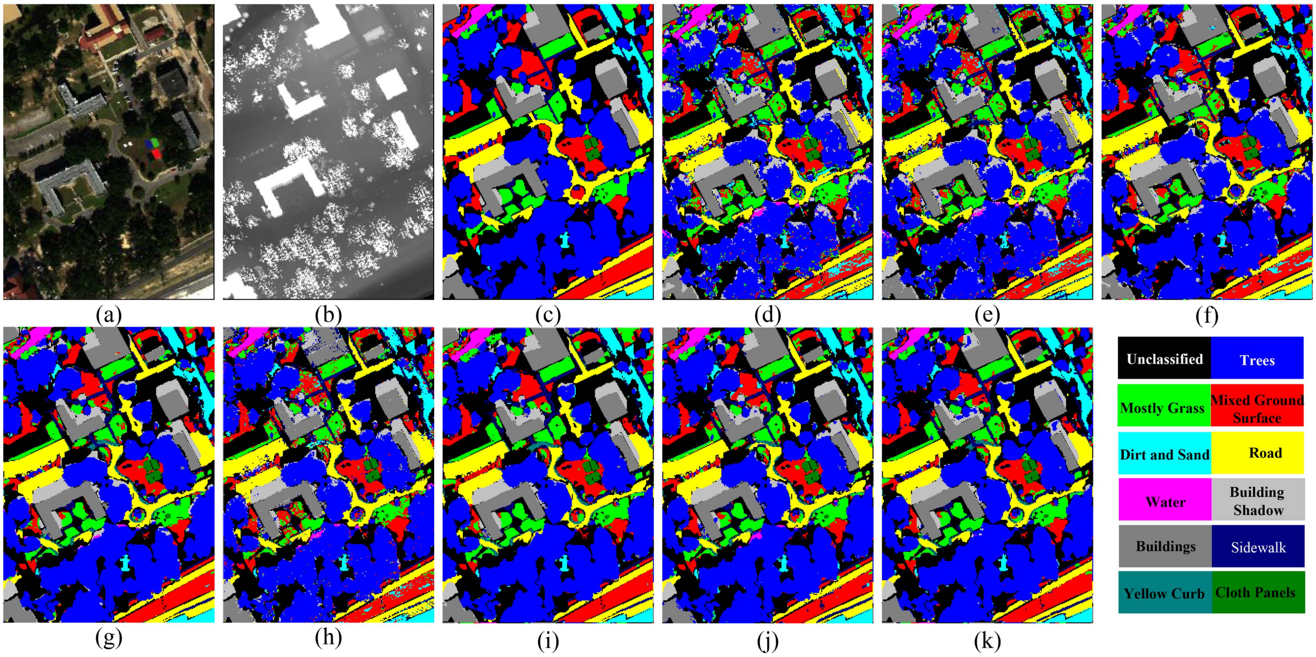


Fig. 12. MUUFL data visualization and classification maps obtained by different models. (a) False-color image for HSI using bands 25, 15, and 5 as R, G, and B, respectively. (b) Grayscale image for LiDAR data. (c) Ground truth. (d) SVM. (e) ELM. (f) CDCNN. (g) TBCNN. (h) EndNet. (i) FusAtNet. (j) S2Enet. (k) MSLAENet.

TABLE VIII
Computation Time With Different Deep Learning Methons on Three Dataset

Dataset	Time(s)	CDCNN	TBCNN	EndNet	FusAtNet	S2ENet	MSLAENet
Houston	train	133.79	230.62	130.33	1205.22	227.52	475.77
	test	3.62	4.93	3.59	13.29	4.05	5.12
Trento	train	61.91	76.79	53.42	325.78	105.54	212.97
	test	9.1	10.98	8.87	26.42	9.21	11.15
MUUFL	train	88.61	96.88	85.34	629.25	175.77	501.81
	test	16.49	18.32	15.92	52.1	16.82	20.24

TABLE IX
Ablation Experiments About Different Module Combination on Different Datasets

CA	MSL	FF	OA(%)		
			houston	Trento	MUUFL
			94.19	98.11	91.05
√			94.64	98.57	91.81
	√		94.80	98.52	92.05
		√	95.51	99.09	92.12
√	√		95.04	98.89	92.13
√		√	95.73	99.13	92.27
	√	√	95.67	99.16	92.18
√	√	√	**96.47**	**99.33**	**92.62**

TABLE X
Ablation Experiments About CA Module Using Different Attention Combination on Different Datasets

SpeAtt_H	SpaAtt_H	SpaAtt_L	OA(%)		
			houston	Trento	MUUFL
			95.67	99.16	92.18
√			95.87	99.19	92.22
	√		95.91	99.22	92.35
		√	95.74	99.23	92.20
√	√		96.17	99.29	92.47
√		√	96.26	99.29	92.28
	√	√	96.35	99.30	92.38
√	√	√	**96.47**	**99.33**	**92.62**

degrees, especially. Our FF module can significantly improve the classification results, and on the Houston, Trento, and MUUFL datasets, we were able to obtain OA improvements of 1.32%, 0.98%, and 1.07% on the baseline network by adding the FF module alone. It can obtain better classification performance than using a single module by further combining these modules, which also shows that MSLAENet benefits from the combination of several modules.

To test the contribution of each branch attention and cross-modal enhancement approach in the CA module, we conducted an ablation study, and the experimental results are shown in Table X. SpeAtt_H denotes spectral attention for HSI, SpaAtt_L denotes spatial attention for LiDAR data, and SpaAtt_H denotes spatial attention for HSI, namely CME. It can be seen that the addition of spatial and spectral attention can enhance the feature

TABLE XI
ABLATION EXPERIMENTS ABOUT THE MSL MODULE USING DIFFERENT CONVOLUTION ON DIFFERENT DATASETS

Name	self-calibrated convolution	vanilla convolution
Houston	96.47	96.29
Trento	99.33	99.30
MUUFL	92.62	92.44

TABLE XII
ABLATION EXPERIMENTS USING DIFFERENT PERCENTAGES OF TRAINING SAMPLES ON DIFFERENT DATASETS

Dataset	30% data	50%data	70%data	100%data
Houston	92.49	94.17	95.26	96.47
Trento	98.57	99.02	99.12	99.33
MUUFL	87.67	90.09	91.35	92.62

representation of HSI and LiDAR data and achieve better classification results, while the CME approach will further enhance the classification effect, which is because the adoption of this approach makes the HSI branch acquire the spatial information of the LiDAR branch and strengthen the feature representation of HSI.

To test the contribution of self-calibrated convolution in the MSL module, we compared the classification results in MSL using self-calibrated convolution and vanilla convolution, as shown in Table XI, using self-calibrated convolution will obtain better classification OA.

Moreover, we conducted additional ablation experiments on all datasets to explore the classification accuracy when using different numbers of training samples. Table XII shows the performance with different percentages of training samples from 30% to 100%, where 100% represents exactly the number of training samples listed in Tables I–III. It can be seen that as the number of training samples increases, the classification OA also increases.

IV. CONCLUSION

In this article, a network for HSI and LiDAR data fusion classification is proposed, which uses self-attention mechanism to adaptively extract spectral and spatial information from HSI and LiDAR data, and cross-attention is used to achieve CME and we use LiDAR data to enhance feature representation of HSI data; self-calibrated convolution and hierarchical residual connection are used to construct MSL module to extract multi-scale information in remote sensing images for classification; in addition, we construct a new attention-based FF module that takes location information into account and fully considers the information complementarity between the two modal data. The effectiveness of the algorithm proposed in this article is demonstrated by conducting experimental validation on three commonly used HSI and LiDAR classification datasets and comparing them with other state-of-the-art methods. However, by experimental analysis of different training samples, we find that our method is highly labeled sample-dependent. In future work, we will consider using weakly supervised or self-supervised techniques to improve this problem.

ACKNOWLEDGMENT

The authors would like to thank Prof. L. Bruzzone, University of Trento, for providing the Trento dataset and the IEEE GRSS Image Analysis and Data Fusion Technical Committee for providing the Houston 2013.

REFERENCES

[1] M. Rast and T. H. Painter, “Earth observation imaging spectroscopy for terrestrial systems: An overview of its history, techniques, and applications of its missions,” *Surv. Geophys.*, vol. 40 no. 3, pp. 303–331, 2019.

[2] 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.

[3] P. Dong, Q. C. Barrett, and C. Batey, *LiDAR Remote Sensing and Applications*. Boca Raton, FL, USA: CRC Press, 2017.

[4] P. Ghamisi et al., “Multisource and multitemporal data fusion in remote sensing: A comprehensive review of the state of the art,” *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 1, pp. 6–39, Mar. 2018.

[5] J. A. Benediktsson, J. A. Palmason, and J. R. Sveinsson, “Classification of hyperspectral data from urban areas based on extended morphological profiles,” *IEEE Trans. Geosci. Remote Sens*, vol. 43, no. 3, pp. 480–491, Mar. 2005.

[6] M. Dalla Mura, J. A. Benediktsson, B. Waske, and L. Bruzzone, “Morphological attribute profiles for the analysis of very high resolution images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 10, pp. 3747–3762, Oct. 2010.

[7] P. Ghamisi, R. Souza, J. A. Beneiktsson, X. X. Zhu, L. Rittner, and R. Lotufo, “Extinction profiles for the classification of remote sensing data,” *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 5631–5645, Oct. 2016.

[8] W. Liao, R. Bellens, A. Pizurica, S. Gautama, and W. Philips, “Combining feature fusion and decision fusion for classification of hyperspectral and LiDAR data,” in *Proc. IEEE Geosci. Remote Sens. Symp.*, 2014, pp. 1241–1244.

[9] P. Ghamisi, J. A. Benediktsson, and S. Phinn, “Land-cover classification using both hyperspectral and lidar data,” *Int. J. Image Data Fusion*, vol. 6, no. 3, pp. 189–215, 2015.

[10] 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 Obs. Remote Sens*, vol. 10, no. 6, pp. 3011–3024, Jun. 2017.

[11] W. Liao, A. Pi zurica, R. Bellens, S. Gautama, and W. Philips, “Generalized graph-based fusion of hyperspectral and lidar data using morphological features,” *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 3, pp. 552–556, Mar. 2015.

[12] L. Yan, M. Cui, and S. Prasad, “Joint euclidean and angular distance-based embeddings for multisource image analysis,” *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 7, pp. 1110–1114, Jul. 2018.

[13] D. Hong, N. Y. okoya, J. Chanussot, and X. Zhu, “CoSpace: Commonsubspace learning from hyperspectral-multispectral correspondences,” *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 4349–4359, Jul. 2019.

[14] W. Liao, R. Bellens, A. Pižurica, S. Gautama, and W. Philips, “Combining feature fusion and decision fusion for classification of hyperspectral and LiDAR data,” in *Proc. IEEE Geosci. Remote Sens. Symp.*, 2014, pp. 1241–1244.

[15] J. Ham, Y. Chen, M. M. Crawford, and J. Ghosh, “Investigation of the random forest framework for classification of hyperspectral data,” *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 492–501, Mar. 2005.

[16] Q. Yuan et al., “Deep learning in environmental remote sensing: Achievements and challenges,” *Remote Sens. Environ.*, vol. 241, 2020, Art. no. 111716.

[17] L. Ma, Y. Liu, X. Zhang, Y. Ye, G. Yin, and B. Johnson, “Deep learning in remote sensing applications: A meta-analysis and review,” *ISPRS J. Photogrammetry Remote Sens.*, vol. 152, pp. 166–177, 2019.

[18] Y. Wan, Y. Qian, X. Zhong, H. Liu, and L. Chen, “Hyperspectral images classification based on double-branch networks with attention feature fusion,” *J. Appl. Remote Sens.*, vol. 15, no. 3, 2021, Art. no. 036517.

[19] 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.

[20] X. He, A. Wang, P. Ghamisi, G. Li, and Y. Chen, "LiDAR data classification using spatial transformation and CNN," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 1, pp. 125–129, Jan. 2019.

[21] L. Sun, S. Cheng, Y. Zheng, Z. Wu, and J. Zhang, "SPANet: Successive pooling attention network for semantic segmentation of remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 4045–4057, May 2022.

[22] L. Gao et al., "STransFuse: Fusing swin transformer and convolutional neural network for remote sensing image semantic segmentation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 10990–11003, Oct. 2021.

[23] L. Chen, H. Liu, M. Yang, Y. Qian, Z. Xiao, and X. Zhong, "Remote sensing image super-resolution via residual aggregation and split attentional fusion network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 9546–9556, Sep. 2021.

[24] Y. Chen, C. Li, P. Ghamisi, X. Jia, and Y. Gu, "Deep fusion of remote sensing data for accurate classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 8, pp. 1253–1257, Aug. 2017.

[25] Q. Feng, D. Zhu, J. Yang, and and B. Li, "Multisource hyperspectral and lidar data fusion for urban land-use mapping based on a modified two-branch convolutional neural network," *ISPRS Int. J. Geo- Inf.*, vol. 8, no. 1, 2019, Art. no. 28.

[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, 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] 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.

[29] J. Wang, J. Zhang, Q. Guo, and T. Li, "Fusion of hyperspectral and lidar data based on dual-branch convolutional neural network," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 3388–3391.

[30] 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. 2022, Art. no. 5500205.

[31] S. Mohla, S. Pande, B. Banerjee, and S. Chaudhuri, "Fusatnet: Dual attention based spectrospatial multimodal fusion network for hyperspectral and lidar classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2020, pp. 92–93.

[32] 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.

[33] M. Guo et al., "Attention mechanisms in computer vision: A Survey," in *Proc. Comput. Vis. Media*, 2022, pp. 1–38.

[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] J. M. Haut, M. E. Paoletti, J. Plaza, A. Plaza, and J. Li, "Visual attention-driven hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 8065–8080, Oct. 2019.

[36] L. Sun, Y. Fang, Y. Chen, W. Huang, Z. Wu, and B. Jeon, "Multi-structure KELM with attention fusion strategy for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–17, Sep. 2022.

[37] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 13713–13722.

[38] 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.

[39] J. J. Liu, Q. Hou, M. M. Cheng, C. Wang, and J. Feng, "Improving convolutional networks with self-calibrated convolutions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10096–10105.

[40] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. Torr, "Res2net: A new multi-scale backbone architecture," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 2, pp. 652–662, Feb. 2021.

[41] Y. Dai, F. Gieseke, S. Oehmcke, Y. Wu, and K. Barnard, "Attentional feature fusion," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2021, pp. 3560–3569.

[42] C. Debes et al., "Hyperspectral and LiDAR data fusion: Outcome of the 2013 GRSS data fusion contest," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2405–2418, Jun. 2014.

[43] G. Mercier and M. Lennon, "Support vector machines for hyperspectral image classification with spectral-based kernels," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2003, vol. 1, pp. 288–290.

[44] W. Li, C. Chen, H. Su, and Q. Du, "Local binary patterns and extreme learning machine for hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 7, pp. 3681–3693, Jul. 2015.

[45] H. Lee and H. Kwon, "Going deeper with contextual CNN for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 26, no. 10, pp. 4843–4855, Oct. 2017.

[46] S. Fang, K. Li, and Z. Li, "S^{2}ENet: Spatial-spectral cross-modal enhancement network for classification of hyperspectral and LiDAR data," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, Oct. 2022.

Yingying Fan received the bachelor's degree in software engineering from Xinjiang University, Urumqi, China, in 2014 and the master's degree in engineering in 2017 from the College of Software, Xinjiang University, where she is currently working toward the Ph.D. degree in computer science and technology.

Her research fields include deep learning and remote sensing image processing.

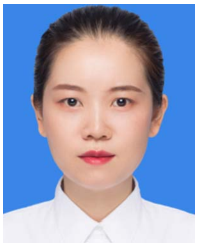
Yurong Qian received the bachelor's and master's degrees in computer science and technology from Xinjiang University, Urumqi, China, in 2002 and 2005, respectively, and a doctorate degree in biology from Nanjing University, Nanjing, China, in 2010.

From 2012 to 2013, she worked as a Postdoctoral Fellow with the Department of Electronics and Computer Engineering, Hanyang University, South Korea. She is currently a Professor with the School of Software, Xinjiang University, China. In 2015, she was trained as a Young Scientific and Technological Innovation Talent by the Science and Technology Department, Xinjiang Province, China. Her research interests include computational intelligence such as Big Data processing, image processing, and artificial neural networks.

Prof. Qian is a Senior Member of the Chinese Computer Federation.

Yugang Qin received the bachelor's degree in computer science from the Wuhan Institute of Technology, Wuhan, China, in 2021. He is currently working toward the master's degree in software engineering from Xinjiang University, Urumqi, China.

His research interests include image processing, pedestrian detection, object detection, and semantic segmentation.



Yaling Wan received the master's degree in engineering from the College of Software, Xinjiang University, Urumqi, China, in 2022.

She is currently a Teacher with the college of education science, Xinjiang Normal University. Her research interests include deep learning and hyperspectral image classification.

Weijun Gong received the bachelor's degree in electronic information engineering from the Wuhan University of Science and Technology, Wuhan, China, in 2009, and the master's degree in computer application technology from the Lanzhou University of Technology, Lanzhou, China, in 2012. He is currently working toward the doctoral degree in computer science and technology with Xinjiang University, Urumqi, China.

His research interests include deep learning, image classification, and emotional expression analysis.

Zhuang Chu received the bachelor's degree in software engineering from Xinjiang University, Urumqi, China, in 2021. He is currently working toward the master's degree in software engineering from Xinjiang University, Urumqi, China.

His research interests include deep learning and remote sensing image classification.

Hui Liu received the bachelor's degree in software engineering from Xinjiang University, Urumqi, China, in 2014, the Master of Engineering from the college of software, Xinjiang University, Urumqi, China, in 2017. She is currently working toward the Ph.D. degree in computer science and technology with Xinjiang University, Urumqi, China.

Her research interests include deep learning and opportunistic networks and the processing of remote sensing image data.