

Spectral-Spatial Latent Reconstruction for Open-Set Hyperspectral Image Classification

Jun Yue, Leyuan Fang, *Senior Member, IEEE*, and Min He

Abstract—Deep learning-based methods have produced significant gains for hyperspectral image (HSI) classification in recent years, leading to high impact academic achievements and industrial applications. Despite the success of deep learning-based methods in HSI classification, they still lack the robustness of handling unknown object in open-set environment (OSE). Open-set classification is to deal with the problem of unknown classes that are not included in the training set, while in closed-set environment (CSE), unknown classes will not appear in the test set. The existing open-set classifiers almost entirely rely on the supervision information given by the known classes in the training set, which leads to the specialization of the learned representations into known classes, and makes it easy to classify unknown classes as known classes. To improve the robustness of HSI classification methods in OSE and meanwhile maintain the classification accuracy of known classes, a spectral-spatial latent reconstruction framework which simultaneously conducts spectral feature reconstruction, spatial feature reconstruction and pixel-wise classification in OSE is proposed. By reconstructing the spectral and spatial features of HSI, the learned feature representation is enhanced, so as to retain the spectral-spatial information useful for rejecting unknown classes and distinguishing known classes. The proposed method uses latent representations for spectral-spatial reconstruction, and achieves robust unknown detection without compromising the accuracy of known classes. Experimental results show that the performance of the proposed method outperforms the existing state-of-the-art methods in OSE.

Index Terms—Deep neural network, hyperspectral image classification, latent reconstruction, open-set classification, spectral feature reconstruction, open-set environment.

I. INTRODUCTION

HYPERSPECTRAL images (HSIs) contain both spatial information and spectral reflectance information of ground objects, in which each pixel corresponds to a spectral curve. The spectral detection range corresponding to hyperspectral technology far exceeds the perception range of the human eye, which can improve people's understanding of nature [1], [2]. Combining both spatial texture feature

This work was supported in part by the National Natural Science Foundation of China under Grant 61922029 and Grant 62101072, in part by the Hunan Provincial Natural Science Foundation of China under Grant 2021JJ30003 and Grant 2021JJ40570, in part by the Science and Technology Plan Project Fund of Hunan Province under Grant 2019RS2016 and in part by the Scientific Research Foundation of Hunan Education Department under Grant 20B022 and Grant 20B157. (*Corresponding author: Leyuan Fang*)

Jun Yue is with the Department of Geomatics Engineering, Changsha University of Science and Technology, Changsha 410114, China (e-mail: jyue@pku.edu.cn).

Leyuan Fang is with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China, and also with the Peng Cheng Laboratory, Shenzhen 518000, China (e-mail: fangleyuan@gmail.com).

Min He is with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China (e-mail: hemin@hnu.edu.cn).

and spectral feature, HSIs have been widely and successfully applied in the fields of environment, agriculture, ecology, ocean, geology, land management and so on. The purpose of HSI classification is to classify each pixel into corresponding ground class (e.g., building, soil, grassland, tree, river and road). As a core step in HSI data processing, HSI classification plays an irreplaceable role in most hyperspectral technology applications [3], [4], [5].

Based on the high spectral dimension of HSI, researchers have proposed many statistical transformation methods to map HSI spectral vectors from high-dimensional space to low-dimensional feature space to extract effective spectral features [6]. Some classic examples of such statistical transformation methods are Linear Discriminant Analysis (LDA) [7], Sparse Preserving Projection (SPP) [8], Local Preserving Projection (LPP) [9] and Local Preserving Discriminant Analysis (LFDA) [10]. Due to the spatial homogeneity and heterogeneity of HSI, it is difficult to make full use of the advantages of HSI only by extracting spectral features. Therefore, researchers have proposed a series of methods to jointly extract spatial-spectral features, such as extended mathematical contour (EMP) [11], directional morphological contour (DMP) [12], and extended attribute profiles (EAP) [13].

With the successful introduction and rapid progress of deep neural network (DNN) [14], it has achieved excellent performance in image classification [15], [16], image segmentation [17], [18], object detection [19], [20], image sharpening [21], [22], etc. Inspired by this, researchers have proposed a series of HSI spectral feature and spatial feature extraction methods based on DNN, so that the classification accuracy of HSI has been gradually improved. In order to fully explore the three-dimensional data characteristics of HSI, some spectral-spatial feature extraction methods combined with DNN have been proposed [23], including stacked auto-encoders [24], spatial pyramid pooling [25], deep fully convolutional network [23], [26], deep deconvolution network [27], superpixel-based discriminative sparse model [28], deep residual network [29], [30], deep transformer [31], and deep recurrent neural network [32].

The performance of HSI classification has been developed by leaps and bounds with the development of deep learning. However, most of the existing HSI classification methods have an ideal assumption, that is, the ground object classes of all samples in the test set appear in the training set. The classification task in a closed-set environment (CSE) is called closed-set classification (CSC) [33], [34]. In fact, this assumption may be invalid in some situations, because in actual application scenarios, there are likely to be ground

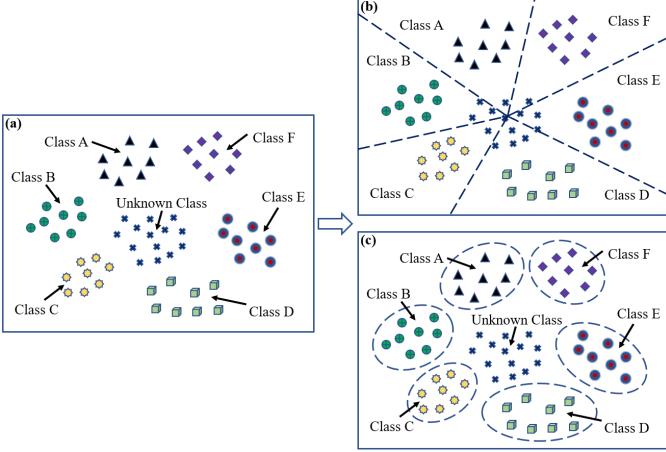


Fig. 1: The illustration of the closed-set classification (CSC) and the open-set classification (OSC). (a) The Illustration of the distribution of the dataset including known classes A, B,..., F and unknown class. (b) The Illustration of the decision boundaries generated by closed-set model and classifier. The closed-set classifier mistakenly divides unknown objects into known classes. (c) The Illustration of the decision boundaries generated by open-set model and classifier.

object classes that do not exist in the training set. Considering that the classes of ground objects on the Earth are very rich, in the face of the vast unknown world, researchers inevitably have to study in depth how to effectively deal with a large number of unknown and dynamically changing open ground object classes in open-set environment (OSE).

Since it is almost impossible for the training set to cover all possible classes in the real word [35], the usability of the model trained in CSE is greatly restricted [36], [37]. In real scenarios, the known classes contained in the training set are often incomplete. In the process of model inference and application, researchers may encounter many new classes that cannot be seen during the training process. In OSE, the object class set in the training set is a subset of the object class set in the test set. The task to be solved in this paper is to accurately classify the known classes and identify the unknown open classes at the same time. This task is also called as the open-set classification (OSC) [38], [39]. Fig. 1 shows the comparison between CSC and OSC. Obviously, CSC will weaken the robustness of the classification method based on the closed-set hypothesis. It can be seen from Fig. 1 that an ideal OSC generated after training can effectively limit the range of known classes in the feature space, so as to reserve enough feature space for the recognition of unknown open classes, and realize the accurate unknown recognition [40], [41].

Generally speaking, researchers will decompose OSC into two sub-tasks (i.e., identifying unknown open classes and classifying known classes) [38]. Identifying the unknown open class can be regarded as a two-class classification problem, that is, to judge whether the test sample belongs to the unknown open classes. This process is also called unknown detection or novelty detection in the literature [42]. Researchers have

proposed a large number of methods to identify unknown open classes, including methods based on distance [43], [44], methods based on statistics [45], [46], [47], methods based on domain [48], [49], methods based on reconstruction [50], [51], [52], methods based on neural network [53] and methods based on information theory [54]. These methods mainly consider how to improve the accuracy of identifying open classes, and to a certain extent ignore how to improve the performance of the sub-task of classifying known classes in OSC [55].

Compared with the closed-set classifier, the open-set classifier [56] trained under OSE can detect samples that do not belong to the classes in the training set. However, usually, the feature extractor corresponding to the open-set classifier will map all samples into the feature space, including samples within known classes and unknown open samples. Therefore, for OSC, it is particularly important to retain features that can help rejecting unknown open classes in the feature extractor. For the feature extractor corresponding to the open classifier, almost all existing deep open classifiers rely on fully supervised learning [33], [57], [58], [59] on the training set, as shown in Fig. 2(a). Most existing OSC methods only use the final prediction of their models to identify unknown open classes and classify known classes. However, the feature extractor trained based on the training set (containing only samples within known classes) tends to retain the features that can help to accurately classify the known classes, and often ignores the features that can be used to identify the unknown open classes. Features that can be used to accurately classify known classes may not necessarily be used to represent unknown classes or separate unknown open classes from known classes [40], [60], [61].

In this paper, we propose a novel open-set HSI classification framework, called Spectral-Spatial Latent Reconstruction (SSLR) learning under the OSE. Our goal is to learn effective spectral-spatial feature representation for each sample, and retain the features that can classify known categories and identify unknown open classes. For unknown open classes that are invisible during training, we can add unsupervised learning as a regularizer, so that the learned spectral-spatial representation can obtain features that are useful for identifying unknown open classes but may not help to classify known classes. Reconstructing the original input from the low dimensional latent representation is a common method of unsupervised learning [14]. Therefore, in the final optimization goal, in addition to the supervised optimization goal in CSE, we also combine the unsupervised learning goal of spectral reconstruction and the unsupervised learning goal of spatial reconstruction. The proposed SSLR method is mainly composed of two modules: spectral-spatial latent reconstruction module and spectral-spatial open-set calibration module. The spectral-spatial open-set calibration module includes unknown detector and closed-set classifier, as shown in Fig. 2(b). Among them, the closed-set classifier uses supervised learning to explore how to improve the accuracy of the classification of known classes, while the unknown detector uses the spectral and spatial latent representation to identify the unknown samples. This mechanism makes it possible for unknown detectors to use spatial-spectral features that are not helpful in classifying

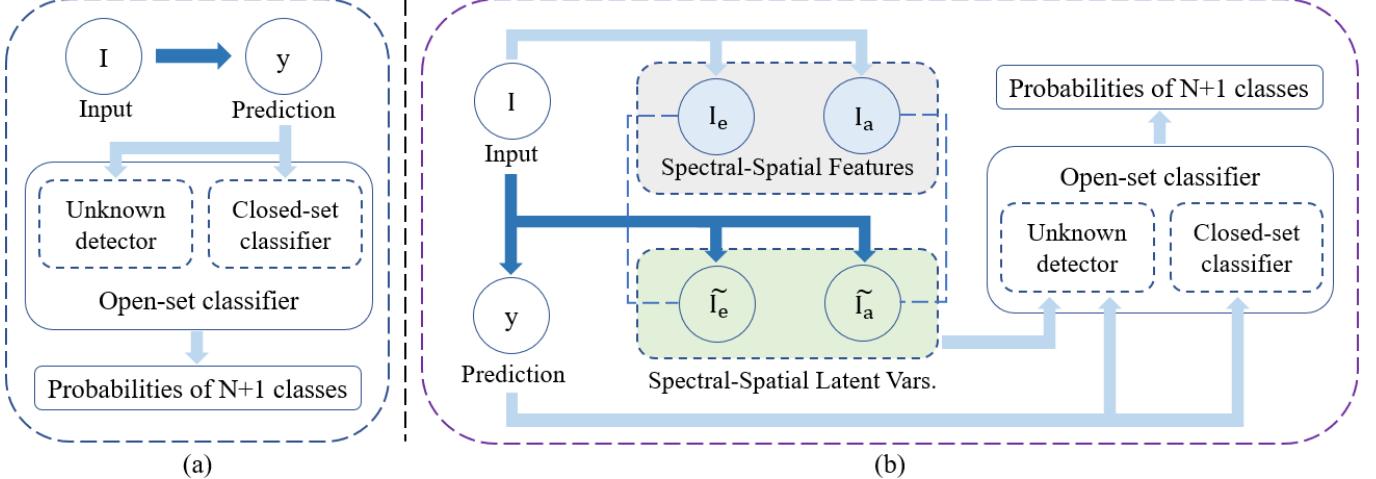


Fig. 2: Overview of existing and our deep open-set classification models. (a) Existing deep open-set classifiers. (b) Spectral-spatial latent reconstruction for open-set HSI classification.

known classes. In addition, the output features of high-level layer of deep neural networks often lose the details of input data [62], [63], [64], [40], which may lead to the failure of unknown detection. The proposed SSLR can use self-supervision to reconstruct spatial and spectral inputs, so as to supplement the missing details to a certain extent.

The rest of this paper is organized as follows. In Section II, the proposed method is described in detail, including the spectral-spatial latent reconstruction module, the spectral-spatial open-set calibration module and the progressive residual convolution module. In Section III, the experimental setup is described in detail, including datasets and training details. In Section VI, the open-set HSI classification results of the proposed method and the comparison methods are shown and discussed, as well as an ablation study to verify the effectiveness of each module. In Section V, the conclusions are summarized, and some future research ideas are given.

II. METHOD

This paper proposes an HSI classification method under open-set setting based on spectral-spatial latent reconstruction, which consists of three modules: spectral-spatial latent reconstruction module, spectral-spatial open-set calibration module and progressive residual convolution module. First, the progressive residual convolution module generates the latent representation of HSIs. Second, the spectral-spatial latent reconstruction module generates the spatial reconstruction matrix and the spectral reconstruction matrix, and calculates the spatial reconstruction error and the spectral reconstruction error. Third, spectral-spatial open-set calibration module models the spectral-spatial reconstruction error, obtains the cumulative distribution function of the Weibull distribution, and judges whether the HSI instance belongs to known classes based on the cumulative distribution function. The framework of the proposed SSLR method is shown in Fig. 3.

A. Spectral-Spatial Latent Reconstruction Module

Assume a 3-D HSI cube $\mathcal{H} \in \mathbb{R}^{N_X \times N_Y \times C}$, where N_X and N_Y are the length and width of the HSI cube in the spatial dimension, respectively. C denotes the number of spectral channels of the HSI cube. In order to reconstruct the spatial texture information, we first use the minimum noise fraction (MNF) method to extract the spatial texture information. The standard principal component analysis (PCA) transform is more sensitive to noise, that is, the principal component with the largest amount of information may not have the highest signal-to-noise ratio. When the variance of the noise contained in a certain principal component is greater than the variance of the signal, the quality of the image formed by the principal component will be poor. In order to solve this problem, the proposed method uses the MNF transformation to extract the texture characteristics of the HSI cube. Through MNF transformation, the first T principal components are selected as the spatial domain matrix, which is represented by $\mathcal{H}_{spa} \in \mathbb{R}^{N_X \times N_Y \times T}$.

For the spectral-spatial encoder network $f_\theta(\cdot)$, it consists of several progressive residual convolution module. We feed the spectral-spatial instance $\mathcal{H}_{ins} \in \mathbb{R}^{K \times K \times C}$ in HSI cube to the network $f_\theta(\cdot)$ to obtain the output of the global average pooling (gap) layer as the latent representation and the output of the fully connected layer as the final hidden output. We use Θ_{lr} and y_{ac} to denote the latent representation and the final hidden output, respectively. For each spectral-spatial instance, the spatial neighborhood is $K \times K$ in HSI cube. The above process can be formulated as

$$[\Theta_{lr}, y_{ac}] = f_\theta(\mathcal{H}_{ins}) \quad (1)$$

For the activation vector y_{ac} , its vector length is N_C , where N_C represents the number of known ground object classes under the closed-set setting. For the i th class, the corresponding activation number is y_{ac}^i . We can use softmax to determine the predicted label y_{pre} of the HSI instance as

$$y_{pre} = \arg \max_{i \in [1, N_C]} \frac{\exp(y_{ac}^i)}{\sum_{j=1}^{N_C} \exp(y_{ac}^j)} \quad (2)$$

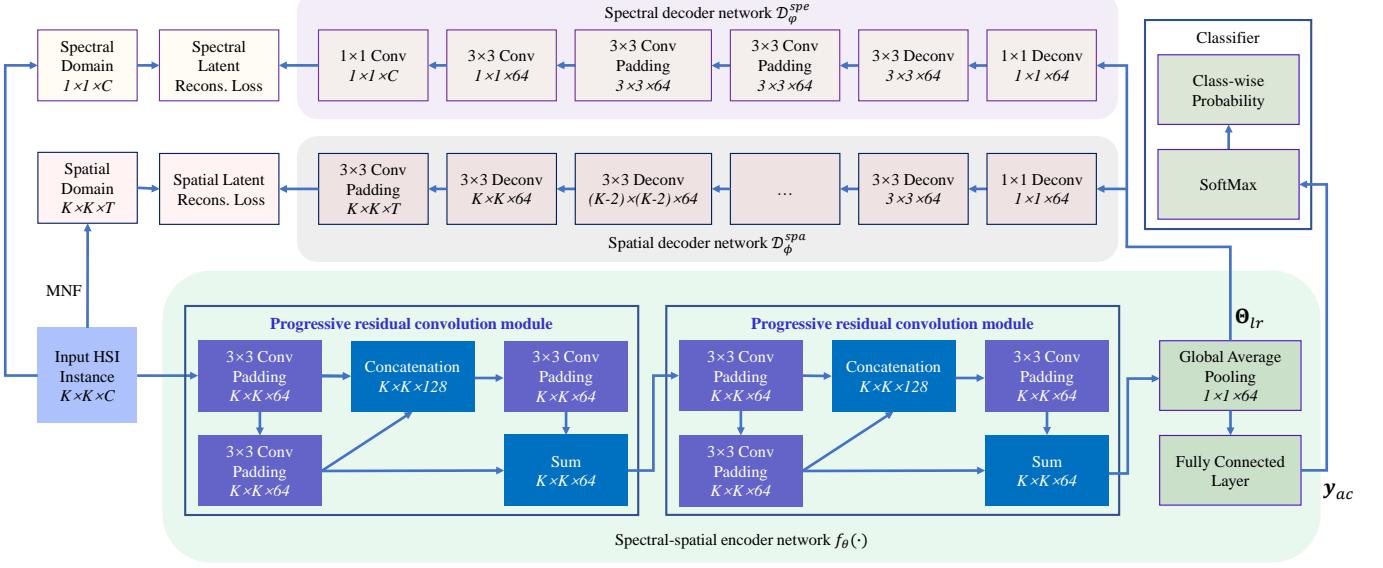


Fig. 3: Spectral-spatial latent reconstruction with progressive residual convolution module.

1) Spatial Latent Reconstruction Loss: To extract spatial texture information during the training process to provide information that can distinguish between known object classes and unknown open object classes, we use the decoder network composed of several deconvolutional layers to reconstruct spatial information from the encoder features. This process can be expressed as

$$\hat{\mathcal{H}}_{spa} = \mathcal{D}_\phi^{spa}(\Theta_{lr}) \quad (3)$$

where $\mathcal{D}_\phi^{spa}(\cdot)$ represents the spatial decoder network used to reconstruct spatial information from the latent representation Θ_{lr} . $\hat{\mathcal{H}}_{spa}$ represents the spatial domain matrix after reconstruction. Corresponding to different HSI instances, we obtain the original spatial domain matrix $\tilde{\mathcal{H}}_{spa} \in \mathbb{R}^{K \times K \times T}$. Then, we calculate the mean absolute error (MAE) \mathcal{E}_{spa} between the original spatial domain matrix $\tilde{\mathcal{H}}_{spa} \in \mathbb{R}^{K \times K \times T}$ and the reconstructed spatial domain matrix $\hat{\mathcal{H}}_{spa} \in \mathbb{R}^{K \times K \times T}$ as the spatial latent reconstruction loss \mathcal{L}_{spa} :

$$\mathcal{L}_{spa} = \mathcal{E}_{spa} = \left\| \tilde{\mathcal{H}}_{spa}, \hat{\mathcal{H}}_{spa} \right\|_1 \quad (4)$$

2) Spectral Latent Reconstruction Loss: One of the characteristics of HSI is that it not only has rich texture features, but also retains the continuous spectral information of the ground objects, which provides a very important indication for distinguishing whether a ground object is open. To fully extract the distinguishable spectral information, we use a spectral decoder network to reconstruct the complete spectral information from the latent representation. First, we obtain the original spectral domain matrix $\tilde{\mathcal{H}}_{spe} \in \mathbb{R}^{1 \times 1 \times C}$ from the spectral-spatial instance $\mathcal{H}_{ins} \in \mathbb{R}^{K \times K \times C}$, and then we use the spectral decoder network to generate the reconstructed spectral domain matrix $\hat{\mathcal{H}}_{spe} \in \mathbb{R}^{1 \times 1 \times C}$ from the latent representation Θ_{lr} :

$$\hat{\mathcal{H}}_{spe} = \mathcal{D}_\varphi^{spe}(\Theta_{lr}) \quad (5)$$

where $\mathcal{D}_\varphi^{spe}(\cdot)$ represents the spectral decoder network. Finally, we calculate the mean absolute error between the original spectral domain matrix $\tilde{\mathcal{H}}_{spe} \in \mathbb{R}^{1 \times 1 \times C}$ and the reconstructed spectral domain matrix $\hat{\mathcal{H}}_{spe} \in \mathbb{R}^{1 \times 1 \times C}$ as the spectral latent reconstruction loss \mathcal{L}_{spe} :

$$\mathcal{L}_{spe} = \mathcal{E}_{spe} = \left\| \tilde{\mathcal{H}}_{spe}, \hat{\mathcal{H}}_{spe} \right\|_1 \quad (6)$$

Finally, we calculate the spectral-spatial latent reconstruction loss $\mathcal{L}_{ss} = \mathcal{L}_{spa} + \mathcal{L}_{spe}$. For a given spectral-spatial instance \mathcal{H}_{ins} , the method of generating spatial and spectral latent reconstruction loss is shown in Algorithm 1. We use the back-propagation method to solve the corresponding parameters:

$$\arg \min_{\theta, \phi, \varphi} (\mathcal{L}_{ce} + \mathcal{L}_{ss}) \quad (7)$$

where \mathcal{L}_{ce} represents the cross entropy loss corresponding to the spectral-spatial encoder network $f_\theta(\cdot)$.

B. Spectral-Spatial Open-Set Calibration Module

Through the previous spectral-spatial latent reconstruction module, the spectral reconstruction MAE and spatial reconstruction MAE are generated. For an HSI instance, if its reconstruction error is large, it means that the instance has not been optimally optimized. Therefore, an HSI instance with a large reconstruction error will be regarded as an unknown open class [65]. In order to find potential open ground objects, we use Extreme Value Theory (EVT) to model the spectral and spatial MAEs.

In the process of extreme value testing, the Weibull distribution is suitable for modeling long-tail data [33], [40], [65]. Through the Weibull distribution, we can find the instance that deviates and treat it as an unknown open ground object. The probability density function (PDF) of the Weibull distribution is as follows:

Algorithm 1 Spectral-Spatial Latent Reconstruction Loss

Input:A given HSI instance $\mathcal{H}_{ins} \in \mathbb{R}^{K \times K \times C}$ Spectral-spatial encoder network $f_\theta(\cdot)$ Spatial decoder network $\mathcal{D}_\phi^{spa}(\cdot)$ Spectral decoder network $\mathcal{D}_\varphi^{spe}(\cdot)$

- 1: Compute the latent representation and activation vector: $[\Theta_{lr}, \mathbf{y}_{ac}] = f_\theta(\mathcal{H}_{ins})$
 - 2: Compute the reconstructed spatial domain matrix $\hat{\mathcal{H}}_{spa} = \mathcal{D}_\phi^{spa}(\Theta_{lr})$ by Eq. (3)
 - 3: Compute the reconstructed spectral domain matrix $\hat{\mathcal{H}}_{spe} = \mathcal{D}_\varphi^{spe}(\Theta_{lr})$ by Eq. (5)
 - 4: Generate the original spatial domain matrix $\tilde{\mathcal{H}}_{spa} \in \mathbb{R}^{K \times K \times T}$ and spectral domain matrix $\tilde{\mathcal{H}}_{spe} \in \mathbb{R}^{1 \times 1 \times C}$ of the HSI instance
 - 5: Calculate the MAE between the original spatial domain matrix and the reconstructed spatial domain matrix \mathcal{L}_{spa} by Eq. (4)
 - 6: Calculate the MAE between the original spectral domain matrix and the reconstructed spectral domain matrix \mathcal{L}_{spe} by Eq. (6)
 - 7: Compute the spectral-spatial latent reconstruction loss: $\mathcal{L}_{ss} = \mathcal{L}_{spa} + \mathcal{L}_{spe}$
 - 8: Return \mathcal{L}_{ss} , \mathcal{L}_{spa} , \mathcal{L}_{spe}
-

$$\mathcal{F}_{\kappa, \lambda}^{pdf}(x) = \begin{cases} \frac{\kappa}{\lambda} \left(\frac{x}{\lambda} \right)^{\kappa-1} e^{-\left(\frac{x}{\lambda} \right)^\kappa} & (x \geq 0) \\ 0 & (x < 0) \end{cases} \quad (8)$$

where κ is the shape parameter, and λ is the scale parameter. The cumulative distribution function (CDF) corresponding to the Weibull distribution is as follows:

$$\mathcal{F}_{\kappa, \lambda}^{cdf}(x) = \begin{cases} 1 - e^{-\left(\frac{x}{\lambda} \right)^\kappa} & (x \geq 0) \\ 0 & (x < 0) \end{cases} \quad (9)$$

We use the spectral-spatial mean absolute error (SSMAE) \mathcal{E}_{ssmae}^i corresponding to the i th known ground object to perform extreme value modeling. The main process of modeling is shown in Algorithm 2. To determine the boundary between unknown and known classes, we need to determine a threshold δ . In previous studies, the threshold is usually set to 50% [65], [40]. For each HSI instance, the method for determining the object classes is shown in Algorithm 3.

C. Progressive Residual Convolution Module

In order to realize the encoder of HSIs, combined with the characteristics of remote sensing images, we design a progressive residual convolution (PRC) network as the spectral-spatial encoder network $f_\theta(\cdot)$. A PRC network is composed of several PRC modules, as shown in Fig. 3. Each PRC module is a convolutional neural network with three layers. In the PRC network, each convolutional layer of the τ th PRC module is denoted by $\Omega_\tau^k(\cdot)$. The PRC module can be formulated as follows:

Algorithm 2 EVT Model Fitting

Input:The number of training samples with known classes N Training samples with known classes $\mathcal{H}_{ins}^i, i \in [1, N]$ The tail number \mathcal{T}

- 1: **for** $i = 1 \dots N$ **do**
 - 2: Generate the spatial reconstructed MAE \mathcal{E}_{spa}^i by Eq. (4)
 - 3: Generate the spectral reconstructed MAE \mathcal{E}_{spe}^i by Eq. (6)
 - 4: Calculate the SSMAE corresponding to the i th sample: $\mathcal{E}_{ssmae}^i = \mathcal{E}_{spa}^i + \mathcal{E}_{spe}^i$
 - 5: **end for**
 - 6: EVT Fit $\mathcal{F}_{\kappa, \lambda}^{cdf}(\cdot) = \text{FitHigh}([\mathcal{E}_{ssmae}^1, \dots, \mathcal{E}_{ssmae}^N], \mathcal{T})$
 - 7: Return model $\mathcal{F}_{\kappa, \lambda}^{cdf}(\cdot)$
-

Algorithm 3 Spectral-Spatial Open-Set Calibration

Input:The total number of testing samples M HSI instances for testing $\mathcal{H}_{ins}^t, t \in [1, M]$ EVT model $\mathcal{F}_{\kappa, \lambda}^{cdf}(\cdot)$ Spectral-spatial encoder network $f_\theta(\cdot)$

- 1: **for** $t = 1 \dots M$ **do**
 - 2: Feed the t th HSI instance \mathcal{H}_{ins}^t into the network $f_\theta(\cdot)$ to generate Θ_{lr}^t and \mathbf{y}_{ac}^t : $[\Theta_{lr}^t, \mathbf{y}_{ac}^t] = f_\theta(\mathcal{H}_{ins}^t)$
 - 3: Generate the predicted ground object class $y_{pre}^t = \arg \max_{i \in [1, N_C]} \frac{\exp(y_{ac}^i)}{\sum_{j=1}^{N_C} \exp(y_{ac}^j)}$ by Eq. (2)
 - 4: Generate the spatial reconstructed MAE \mathcal{E}_{spa}^t and the spectral reconstructed MAE \mathcal{E}_{spe}^t by Eq. (4) and Eq. (6)
 - 5: Calculate the SSMAE corresponding to the t th testing instance: $\mathcal{E}_{ssmae}^t = \mathcal{E}_{spa}^t + \mathcal{E}_{spe}^t$
 - 6: Calculate cumulative probability $\mathcal{F}_{\kappa, \lambda}^{cdf}(\mathcal{E}_{ssmae}^t)$
 - 7: **if** $\mathcal{F}_{\kappa, \lambda}^{cdf}(\mathcal{E}_{ssmae}^t) > \delta$ **then**
 - 8: $y_{pre}^t = \text{unknown}$
 - 9: **end if**
 - 10: **end for**
 - 11: Return $y_{pre}^t, t \in [1, M]$
-

$$\begin{cases} \mathcal{O}_\tau^k = \Omega_\tau^k(\mathcal{I}_\tau^k), k \in \{1, 2, 3\} \\ \mathcal{I}_\tau^{k+1} = \begin{cases} \mathcal{O}_\tau^k & (k = 1) \\ [\mathcal{O}_\tau^{k-1}, \mathcal{O}_\tau^k] & (k = 2) \\ \mathcal{O}_\tau^{k-1} + \mathcal{O}_\tau^k & (k = 3) \end{cases} \end{cases} \quad (10)$$

where \mathcal{I}_τ^k and \mathcal{O}_τ^k denote the input and output of the k th convolutional layer in the τ th PRC module, respectively. Since there are only 3 convolutional layers in each PRC module, the input of the τ th PRC module \mathcal{I}_τ^4 is equal to the input of the next PRC module: $\mathcal{I}_{\tau+1}^1 = \mathcal{I}_\tau^4$. The proposed network combines the advantages of progressive convolutional network [26] and residual network [15], and can make full use of the multi-scale characteristics of HSIs. By performing concatenation and summation of the outputs of multiple levels, collaborative information processing of multi-scale receptive fields and multi-scale feature extraction can be realized.

For the spectral decoder network $\mathcal{D}_\varphi^{spe}(\cdot)$ and the spatial decoder network $\mathcal{D}_\phi^{spa}(\cdot)$, the inputs are the outputs of the gap layer in the spectral-spatial encoder network Θ_{lr} , which is the latent representation of the network $f_\theta(\cdot)$. For spectral decoder network $\mathcal{D}_\varphi^{spe}(\cdot)$, we use two deconvolutional layers and four convolutional layers for implementation, as shown in Fig. 3. For the spatial decoder network $\mathcal{D}_\phi^{spa}(\cdot)$, we use several deconvolutional layers and one convolution layer to achieve, and the number of deconvolutional layers depends on the input neighborhood size of the HSI instance.

III. EXPERIMENTS

A. Datasets and Evaluation

In order to verify the effectiveness of the proposed method, comparative experiments are conducted on three popular HSI datasets in the OSE, including the University of Pavia (UP) dataset, Salinas (SA) dataset and Houston (HU) dataset. For each dataset, the ordinary open-set environments are created. In order to verify the performance of the proposed method in different open-set settings, we set up an additional open-set environment with high openness for the UP dataset. The detailed open-set environments of these three data sets are listed in Table I. In the following experiment, if not specified, the open-set environments are the ordinary environments.

(1) UP dataset: The spatial resolution and ground coverage of this HSI are 1.3m and 0.8 km \times 0.4 km, respectively. The wavelength range corresponding to this HSI is 0.43 μm to 0.86 μm . In the process of data preprocessing, 12 bands that are severely affected by noise and water absorption were removed, and the remaining 103 spectral channels participate in this experiments after processing [66]. The HSI contains a total of 9 ground object classes. In the ordinary open-set experiment, one class (shadows) is selected as the unknown open class, and the remaining 8 classes are selected as the known classes. Fig. 4 shows the pseudo-color composite image, ground truth image and image legend (including background, known classes and unknown open class) of UP dataset in the ordinary environment. In the high openness open-set experiment, 3 classes (i.e., painted metal sheets, bitumen and shadows) are selected as the unknown open class, and the remaining 6 classes are selected as the known classes.

(2) SA dataset: The spatial resolution and ground coverage of this HSI are 3.7m and 1.9 km \times 0.8 km, respectively. The wavelength range corresponding to this HSI is 0.4 μm to 2.5 μm . In the process of data preprocessing, 20 bands that are severely affected by water absorption were removed, and the remaining 204 spectral channels participate in this experiment after processing. The HSI contains a total of 16 ground object classes. In this experiment, two classes are selected as the unknown open class, and the remaining 14 classes are selected as the known classes. Fig. 5 shows the pseudo-color composite image, ground truth image and image legend (including background, known classes and unknown open class) of SA dataset.

(3) HU dataset: The spatial resolution and image size of this HSI are 2.5m and 349 \times 1905, respectively [32]. The wavelength range corresponding to this HSI is 0.38 μm to

TABLE I: The detailed open-set environments of the three HSI datasets (i.e., UP, SA and HU), where N_X and N_Y are the width and height of the HSI data cube, respectively. N_C represents the number of classes in the HSI data. N_K and N_U represent the number of known classes and the number of unknown open classes selected in the experiment, respectively.

Datasets	Openness	N_X	N_Y	N_C	N_K	N_U
UP	Ordinary	610	340	9	8	1
SA	Ordinary	512	217	16	14	2
HU	Ordinary	1905	349	15	11	4
UP	High	610	340	9	6	3

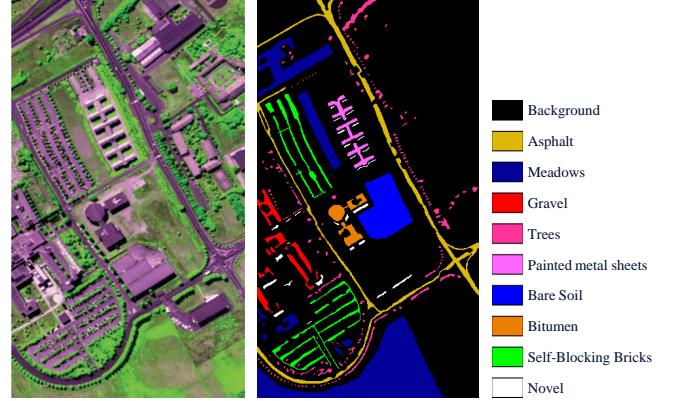


Fig. 4: The false-color composite image of the UP dataset, the corresponding ground truth map and legend (including known classes, unknown open class and background).

1.05 μm . After data preprocessing such as attitude processing and radiation correction, 144 spectral channels are used for experiments. The HSI contains a total of 15 ground object classes. In this experiment, 4 classes are selected as the unknown open class, and the remaining 11 classes are selected as the known classes. Fig. 6 shows the pseudo-color composite image, ground truth image and image legend (including background, known classes and unknown open class) of HU dataset.

On each HSI, we sample L and E from each known class for training and evaluation, and the remaining active pixels are used for testing. In this experiment, three labeled samples ($E = 3$) of each object class are randomly selected for evaluation. In order to avoid extreme values, we perform ten repeated experiments and report the average accuracies and standard deviations of the ten experiments. The accuracy indicators reported in this paper include class-wise accuracy, overall accuracy (OA), average accuracy (AA) and F1 score. Since the OA and AA reported in this experiment include unknown open class, we also refer to OA and AA as open OA and open AA in this paper.

B. Parameter Settings

In order to explore the influence of the number of PRC module N_{prc} on the final accuracy and determine the optimal number of PRC, we set different N_{prc} and test their accuracy

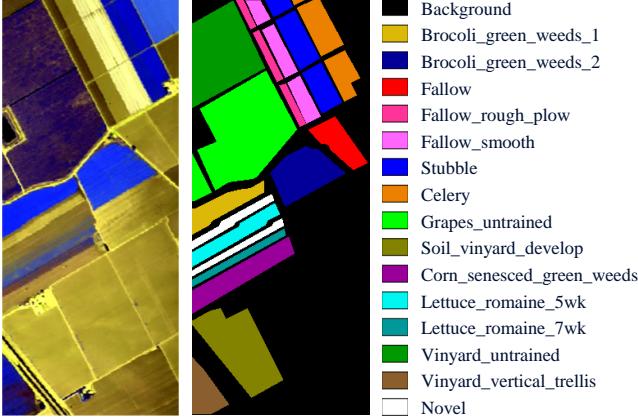


Fig. 5: The false-color composite image of the SA dataset, the corresponding ground truth map and legend (including known classes, unknown open class and background).

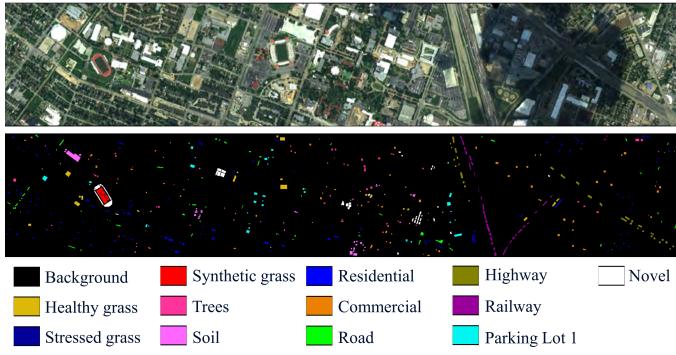


Fig. 6: The false-color composite image of the HU dataset, the corresponding ground truth map and legend (including known classes, unknown open class and background).

on the UP, SA and HU datasets. It is easy to see from the test results shown in Fig. 7 (a)-(c) that the accuracies are the highest when the N_{prc} is set to 2, and so we set the number of PRC module to 2 in the following experiment.

Meanwhile, to explore the influence of the number of principal components (PC) N_{pc} while generating the original spatial domain matrix $\hat{\mathcal{H}}_{spa}$ on the final accuracy and determine the best number of principal components, we set different N_{pc} and test their accuracy on the UP, SA and HU datasets. From the test results shown in Fig. 7 (d)-(f), it is easy to see that the accuracies are the highest when the N_{pc} is set to 3, and so we set the number of principal components to 3 in the following experiment.

We train the SSLR model by optimizing Eq. (7). When training the model, the initial learning rate is set to 0.05. For every 100 epochs, the learning rate is multiplied by 0.8. The computer environment for the experiments is as follows: the processor is “AMD Ryzen 9 5900HX with Radeon Graphics”; the graphics card is “NVIDIA GeForce RTX 3070” with “CUDA version 11.2”; the programming language and the deep learning platform are Python (version 3.7.11) and Tensorflow (version 1.15.4), respectively¹.

IV. RESULTS AND DISCUSSION

A. Accuracies

To verify the accuracy of the HSI classification method under open-set setting, we conducted related experiments on the ordinary openness settings and the high openness setting, and reported the accuracies under different openness conditions. In order to verify the robustness of the proposed method under various conditions to prevent accidental errors, we have conducted ten independent Monte Carlo runs, calculated the average accuracies, and reported the standard deviation of the ten conducted Monte Carlo runs. In addition, we follow the previous studies [65], [67], [68], [23] to verify the robustness of the proposed method under different sample conditions by setting two sample selection conditions. For each experiment, L training samples are randomly selected from each known object class. In this experiment, L is set to 20 and 30 respectively. We compared the proposed method with several state-of-the-art (SOTA) open-set classification methods and SOTA HSI classification methods. Open-set classification methods include MDL4OW (multitask deep learning method for the open world) [65], Classification-Reconstruction learning for Open-Set Recognition (CROSR) [40]. Traditional SOTA HSI classification methods include DFSL (deep few-shot learning for HSI classification) [67], 3D-CNN [69], CDCNN(contextual deep CNN) [70]. These comparison methods include both SOTA open-set classification methods and traditional SOTA HSI classification methods. Therefore, they can be used to comprehensively verify the effectiveness of the proposed SSLR method.

For the UP dataset, we report the class by class accuracy, open OA, open AA and F1 score of the proposed method and the comparison methods in Table II. From this table, we can see that compared with the CDCNN method, when $L = 20$, the open OA, open AA and F1 score of SSLR are increased by 10.6%, 15.1%, and 5.8%, respectively. Compared with the 3D-CNN method, when $L = 20$, the open OA, open AA and F1 score of SSLR are increased by 7.8%, 14.4%, and 4%, respectively. Compared with the DFSL method, when $L = 20$, the open OA, open AA and F1 score of SSLR are increased by 5.5%, 10.2%, and 2.6%, respectively. When $L = 30$, compared with the CDCNN method and the 3D-CNN method, the open OA of SSLR is increased by 10.1% and 8%, and the open AA of SSLR is increased by 14.9% and 14.9%, respectively. For the F1 score, we can see that compared with the CDCNN method, when $L = 30$, the F1 score of SSLR is increased by 5.2%. Compared with the 3D-CNN method, when $L = 30$, the F1 score of SSLR is increased by 4%. When $L = 30$, compared with the DFSL method, the F1 score of the proposed method has been increased from 91.7% to 93.4%. The classification maps of the proposed method and the comparison methods are shown in Fig. 8.

From Table II, we can see that the proposed SSLR can achieve better performance compared with the SOTA open-set recognition methods. Specifically, compared with the CROSR method, the open OA and AA of SSLR are increased by 3.3% and 7.2% when $L = 20$, respectively. Compared with the MDL4OW method, the open OA and AA of SSLR are

¹The source code is available at <https://github.com/JunYue0214>

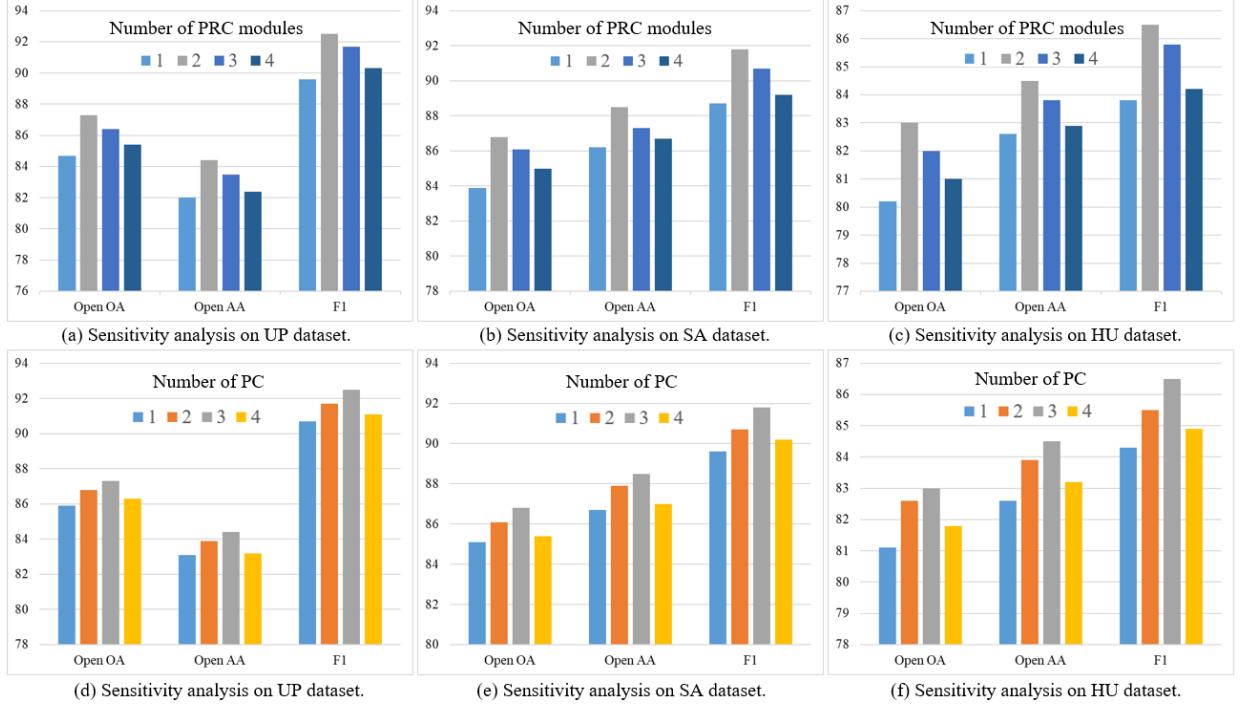


Fig. 7: Sensitivity analysis on the UP, SA and HU datasets. (a)-(c) The accuracies with different number of progressive residual convolution modules. (d)-(f) The accuracies with different number of principal components.

TABLE II: Classification results (%) by using 20 and 30 labeled samples for both known and unknown classes on the UP dataset with the ordinary openness open-set setting.

Class	$L = 20$						$L = 30$					
	CDCNN	3D-CNN	DFSL	CROS R	MDL4OW	Ours	CDCNN	3D-CNN	DFSL	CROS R	MDL4OW	Ours
1	79.8±4.9	71.6±3.2	81.4±3.1	79.0±3.5	88.7±3.1	91.3±2.1	81.3±4.9	87.7±4.0	89.2±3.4	81.4±3.7	91.9±3.3	89.7±2.1
2	75.5±3.3	86.8±4.0	82.6±3.2	89.3±1.3	86.3±2.6	87.9±2.3	77.9±4.8	84.0±3.8	86.4±3.4	90.0±1.3	87.1±2.6	87.8±2.0
3	65.7±1.9	78.4±4.0	76.5±4.6	78.1±2.2	91.3±0.3	91.1±0.5	77.8±2.2	83.5±4.3	82.3±4.9	80.9±2.2	88.4±0.3	92.6±0.5
4	87.0±3.8	74.0±3.3	89.0±2.6	90.6±1.3	89.0±2.6	75.6±2.1	85.6±3.7	76.6±3.5	86.6±2.6	90.6±1.2	92.6±2.7	84.1±2.6
5	57.6±1.7	68.0±2.8	78.2±1.1	78.3±1.1	78.7±0.3	87.1±0.5	75.2±2.1	70.7±3.1	86.3±1.3	75.2±1.0	83.2±0.3	94.8±0.6
6	93.1±2.8	81.6±3.7	92.2±2.7	82.6±2.5	85.1±0.3	90.0±0.8	92.2±2.8	78.7±3.5	85.1±2.4	87.3±2.6	93.0±0.3	98.1±0.6
7	85.4±3.8	86.3±3.9	83.5±5.2	83.9±3.8	95.7±1.4	95.4±0.8	87.1±4.0	91.2±4.0	89.1±5.4	85.9±3.9	92.9±1.3	92.5±2.7
8	75.4±4.7	79.9±3.6	81.9±2.3	81.7±1.2	76.7±4.7	85.6±2.6	76.1±4.7	79.7±3.5	86.8±2.6	81.1±1.2	77.9±4.7	88.1±2.6
Novel	3.8±2.1	3.6±2.2	2.9±2.0	30.8±3.6	42.3±2.8	55.8±1.8	3.1±2.0	3.7±2.5	2.5±1.7	37.6±3.2	43.7±3.0	62.7±1.1
OA	76.7±3.5	79.5±3.7	81.8±3.0	84.0±1.8	85.2±2.2	87.3±1.8	79.0±4.2	81.1±3.7	84.8±3.2	85.4±1.8	87.3±2.3	89.1±1.8
AA	69.3±3.2	70.0±3.4	74.2±3.0	77.2±1.5	81.5±1.4	84.4±1.1	72.9±3.5	72.9±3.6	77.2±3.1	78.9±1.5	83.4±1.4	87.8±1.4
F1	86.7±2.2	88.5±2.2	89.9±1.8	90.9±1.1	91.5±1.3	92.5±1.1	88.2±2.6	89.4±2.2	91.7±1.8	91.6±1.1	92.6±1.3	93.4±1.0

TABLE III: Classification results (%) by using 20 and 30 labeled samples for both known and unknown classes on the UP dataset with the high openness open-set setting.

Class	$L = 20$						$L = 30$					
	CDCNN	3D-CNN	DFSL	CROS R	MDL4OW	Ours	CDCNN	3D-CNN	DFSL	CROS R	MDL4OW	Ours
1	82.8±4.9	76.8±4.4	87.2±3.2	82.8±3.6	87.4±3.0	88.2±1.9	85.5±5.1	91.2±3.9	88.8±3.3	84.8±3.7	84.0±2.8	87.5±1.9
2	75.1±3.1	79.6±3.3	79.4±2.9	79.0±1.1	83.4±2.3	87.9±2.2	78.3±3.3	81.3±3.4	83.7±3.0	80.5±1.1	89.9±2.5	88.6±2.0
3	86.5±2.5	88.6±2.5	77.6±4.5	85.8±2.5	89.0±0.3	89.7±0.5	79.1±2.2	81.1±4.1	85.7±5.0	87.3±2.6	88.6±0.2	94.7±0.6
4	81.1±3.4	85.5±3.7	90.3±2.5	84.3±1.1	87.2±2.4	81.2±2.2	85.3±3.6	73.7±3.2	88.2±2.5	85.5±1.2	84.9±2.3	80.9±2.1
5	82.5±2.2	78.0±2.1	86.2±1.2	86.2±1.1	88.9±0.3	85.8±0.5	83.1±2.3	82.6±3.6	85.4±1.2	85.4±1.1	89.0±0.3	96.6±0.5
6	81.1±2.3	82.8±2.3	87.0±2.5	83.6±2.3	87.4±0.3	89.3±0.7	87.1±2.3	84.2±3.6	77.5±2.2	81.5±2.2	88.1±0.3	88.0±0.5
Novel	2.5±1.6	3.0±1.9	2.7±1.8	45.0±2.9	46.8±0.9	60.2±0.7	3.0±1.9	3.4±2.1	2.9±1.8	47.5±2.7	53.8±0.8	67.0±1.1
OA	72.5±2.8	73.6±2.8	76.3±2.4	78.6±1.3	82.5±1.6	85.1±1.6	74.9±2.9	76.1±3.1	77.7±2.4	79.7±1.3	85.3±1.7	87.3±1.4
AA	70.2±2.4	70.6±2.3	72.9±2.1	78.1±1.3	81.5±1.1	83.2±1.2	71.6±2.4	71.1±2.8	73.2±2.2	78.9±1.3	82.6±1.1	86.2±0.9
F1	83.9±2.0	84.6±2.0	86.4±1.6	85.6±1.0	88.0±1.1	88.9±1.0	85.5±2.0	86.2±2.1	87.3±1.6	86.2±1.0	89.3±1.1	89.8±0.9

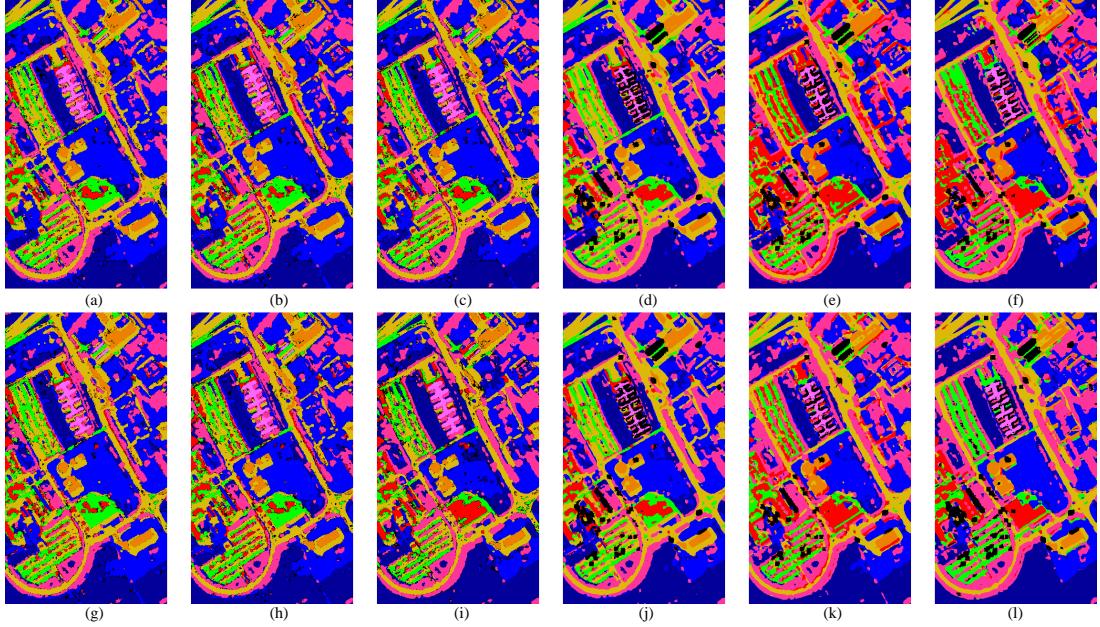


Fig. 8: The classification results of the UP dataset. (a) CDCNN ($L = 20$). (b) 3D-CNN ($L = 20$). (c) DFSL ($L = 20$). (d) CROSER ($L = 20$). (e) MDL4OW ($L = 20$). (f) The proposed SSLR ($L = 20$). (g) CDCNN ($L = 30$). (h) 3D-CNN ($L = 30$). (i) DFSL ($L = 30$). (j) CROSER ($L = 30$). (k) MDL4OW ($L = 30$). (l) The proposed SSLR ($L = 30$).

increased by 1.8% and 4.4% when $L = 30$, respectively. For the F1 score, compared with the CROSER method, the F1 score of SSLR is increased by 1.6% when $L = 20$. Compared with the MDL4OW method, the F1 score of SSLR is increased by 1% when $L = 20$. When $L = 30$, the F1 score of SSLR has been improved by 1.8% and 0.8% compared with the CROSER method and the MDL4OW method, respectively.

For the high openness setting of UP dataset, we report the class by class accuracy, open OA, open AA and F1 score of the proposed method and the comparison methods in Table III. Compared with the MDL4OW method, the open OA and AA of SSLR are increased by 2.6% and 1.7% when $L = 20$, respectively. When $L = 30$, compared with the CROSER method and the DFSL method, the open OA of SSLR is increased by 7.6% and 9.6%, and the open AA of SSLR is increased by 7.3% and 13%, respectively.

In the high openness setting of UP dataset, in addition to comparing the above open-set classification methods and traditional HSI classification methods, we also add the attention-based methods and the graph based method, including Vision Transformer (ViT) [71], SpectralFormer [31] and Graph Convolutional Network (GCN) [72] for comparison. The results (open OA, open AA and F1 score) of the proposed method and the comparison methods are reported in Table IV. Compared with the graph-based method (i.e., GCN), the open OA, open AA and F1 of the proposed method are increased by 10.9%, 11.9% and 8.3% when $L = 20$, respectively. Compared with the attention-based methods (i.e., ViT and SpectralFormer), the open OA of the proposed method is increased by 12.8% and 8.6% when $L = 20$, respectively. When $L = 30$, compared with SpectralFormer, the open OA and open AA of SSLR are increased by 9.5% and 12.4%, respectively. It can be seen from Table IV that the proposed SSLR can effectively improve the

performance of HSI classification under open-set environment.

TABLE IV: Comparison between the proposed method and the attention and graph based methods on the UP dataset with the high openness open-set setting.

Methods		$L = 20$	$L = 30$
ViT	OA	72.3 ± 3.1	73.6 ± 2.9
	AA	71.9 ± 3.2	72.1 ± 3.1
	F1	78.5 ± 2.7	80.2 ± 2.5
GCN	OA	74.2 ± 3.0	75.5 ± 2.8
	AA	71.3 ± 2.5	72.1 ± 2.3
	F1	80.6 ± 2.1	81.3 ± 2.0
SpectralFormer	OA	76.5 ± 2.7	77.8 ± 2.4
	AA	71.9 ± 2.5	73.8 ± 2.3
	F1	82.6 ± 2.1	83.5 ± 2.1
Ours	OA	85.1 ± 1.6	87.3 ± 1.4
	AA	83.2 ± 1.2	86.2 ± 0.9
	F1	88.9 ± 1.0	89.8 ± 0.9

For the SA dataset, we report the class by class accuracy, open OA, open AA and F1 score of the proposed method and the comparison methods in Table V. From this table, we can see that compared with the CDCNN method, when $L = 20$, the open OA and open AA of SSLR are increased by 7.6% and 9.4%, respectively. Compared with the 3D-CNN method, when $L = 20$, the OA and AA of SSLR are increased by 6.2% and 8.3%, respectively. Compared with the DFSL method, when $L = 20$, the OA and AA of SSLR are increased by 6.1% and 8.2%, respectively. When $L = 30$, compared with the CDCNN method and the 3D-CNN method, the open OA of SSLR is increased by 6.9% and 5.8%, and the open AA of SSLR is increased by 7.8% and 6.9%, respectively. For the F1

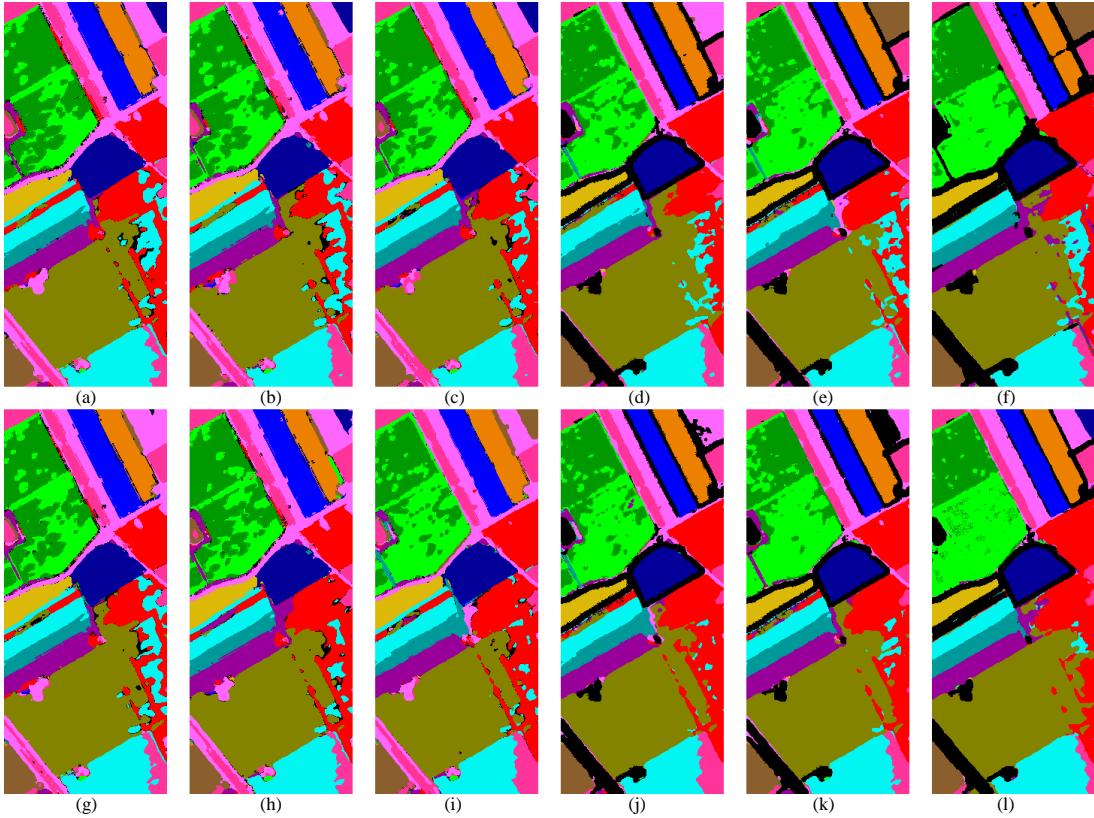


Fig. 9: The classification results of the SA dataset. (a) CDCNN ($L = 20$). (b) 3D-CNN ($L = 20$). (c) DFSL ($L = 20$). (d) CROSER ($L = 20$). (e) MDL4OW ($L = 20$). (f) The proposed SSLR ($L = 20$). (g) CDCNN ($L = 30$). (h) 3D-CNN ($L = 30$). (i) DFSL ($L = 30$). (j) CROSER ($L = 30$). (k) MDL4OW ($L = 30$). (l) The proposed SSLR ($L = 30$).

score, we can see that compared with the CDCNN method, when $L = 20$, the F1 score of SSLR is increased by 3.5%. Compared with the 3D-CNN method, when $L = 20$, the F1 score of SSLR is increased by 2.6%. Compared with the DFSL method, when $L = 20$, the F1 score of SSLR is increased by 2.6%. When $L = 30$, compared with the DFSL method, the F1 score of the proposed method has been increased from 91.0% to 92.6%. The classification maps of the proposed method and the comparison methods are shown in Fig. 9.

From Table V, we can see that the proposed SSLR can achieve better performance compared with the SOTA open-set recognition methods. Specifically, compared with the CROSER method, the open OA and AA of SSLR are increased by 3.8% and 5.1% when $L = 20$, respectively. Compared with the MDL4OW method, the open OA and AA of SSLR are increased by 2.1% and 1.7% when $L = 30$, respectively. For the F1 score, compared with the CROSER method, the F1 score of SSLR is increased by 2.2% when $L = 20$. Compared with the MDL4OW method, the F1 score of SSLR is increased by 0.8% when $L = 20$. When $L = 30$, the F1 scores of SSLR has been improved by 1.7% and 1% compared with the CROSER method and the MDL4OW method, respectively.

For the HU dataset, we report the class by class accuracy, open OA, open AA and F1 score of the proposed method and the comparison methods in Table VI. From this table, we can see that compared with the CDCNN method, when $L = 20$, the open OA and open AA of SSLR are increased by 7.2% and

5.1%, respectively. Compared with the 3D-CNN method, when $L = 20$, the OA and AA of SSLR are increased by 6.4% and 4.4%, respectively. Compared with the DFSL method, when $L = 20$, the OA and AA of SSLR are increased by 6.3% and 4.1%, respectively. When $L = 30$, compared with the CDCNN method and the 3D-CNN method, the open OA of SSLR is increased by 6.9% and 6%, and the open AA of SSLR is increased by 4.7% and 3.8%, respectively. For the F1 score, we can see that compared with the CDCNN method, when $L = 20$, the F1 score of SSLR is increased by 2.9%. When $L = 30$, compared with the DFSL method, the F1 score of the proposed method has been increased from 85.3% to 87.2%. The classification maps of the proposed method and the comparison methods are shown in Fig. 10.

From Table VI, we can see that the proposed SSLR can achieve better performance compared with the SOTA open-set recognition methods. Specifically, compared with the CROSER method, the open OA and AA of SSLR are increased by 3.2% and 2.3% when $L = 20$, respectively. Compared with the MDL4OW method, the open OA and AA of SSLR are increased by 3% and 2.3% when $L = 30$, respectively. For the F1 score, compared with the CROSER method, the F1 score of SSLR is increased by 0.6% when $L = 20$. Compared with the MDL4OW method, the F1 score of SSLR is increased by 0.4% when $L = 20$. When $L = 30$, the F1 score of SSLR has been improved by 1.2% and 0.7% compared with the CROSER method and the MDL4OW method, respectively.

TABLE V: Classification results (%) by using 20 and 30 labeled samples for both known and unknown classes on the SA dataset.

Class	$L = 20$						$L = 30$					
	CDCNN	3D-CNN	DFSL	CROSRR	MDL4OW	Ours	CDCNN	3D-CNN	DFSL	CROSRR	MDL4OW	Ours
1	83.8 \pm 2.5	84.6 \pm 2.7	84.6 \pm 2.6	84.7 \pm 2.7	91.2 \pm 0.6	85.5 \pm 2.7	85.5 \pm 2.7	86.5 \pm 2.8	86.3 \pm 3.0	86.4 \pm 2.8	92.3 \pm 0.5	81.5 \pm 2.0
2	85.4 \pm 1.6	86.5 \pm 1.5	86.5 \pm 1.4	86.4 \pm 1.6	95.0 \pm 0.2	88.2 \pm 2.8	87.3 \pm 1.6	88.2 \pm 1.5	88.3 \pm 1.6	88.3 \pm 1.6	95.7 \pm 0.2	86.3 \pm 2.0
3	85.8 \pm 1.3	87.8 \pm 0.6	87.2 \pm 1.1	88.5 \pm 0.2	84.7 \pm 8.7	94.4 \pm 3.0	89.0 \pm 0.5	89.6 \pm 0.7	90.4 \pm 0.2	90.4 \pm 0.2	84.4 \pm 8.8	96.7 \pm 1.5
4	87.7 \pm 0.3	88.5 \pm 0.3	88.7 \pm 0.2	88.6 \pm 0.3	84.6 \pm 8.8	94.0 \pm 3.0	89.5 \pm 0.3	90.3 \pm 0.3	90.3 \pm 0.3	90.3 \pm 0.2	84.7 \pm 8.8	98.8 \pm 0.2
5	81.6 \pm 4.0	82.9 \pm 3.9	82.7 \pm 3.9	82.6 \pm 3.9	87.0 \pm 0.5	95.5 \pm 1.5	83.7 \pm 3.9	84.7 \pm 3.8	84.5 \pm 3.8	84.8 \pm 3.8	91.0 \pm 0.3	95.6 \pm 0.5
6	84.6 \pm 2.1	85.7 \pm 2.1	85.6 \pm 2.1	85.4 \pm 2.2	86.9 \pm 2.7	90.0 \pm 2.8	86.4 \pm 2.2	87.4 \pm 2.1	87.3 \pm 2.1	87.2 \pm 2.3	88.6 \pm 2.7	87.4 \pm 2.8
7	85.7 \pm 1.5	86.3 \pm 1.6	86.4 \pm 1.6	85.8 \pm 1.8	88.4 \pm 2.7	88.6 \pm 2.8	87.2 \pm 1.6	88.0 \pm 1.8	87.7 \pm 1.9	87.8 \pm 1.8	90.2 \pm 2.8	80.2 \pm 2.5
8	70.3 \pm 11.3	73.9 \pm 9.5	73.8 \pm 9.6	77.9 \pm 6.9	77.6 \pm 2.4	78.3 \pm 2.4	74.1 \pm 10.0	74.7 \pm 10.1	82.1 \pm 5.4	81.4 \pm 6.0	82.4 \pm 2.6	84.7 \pm 2.6
9	87.7 \pm 0.1	88.7 \pm 0.1	88.7 \pm 0.1	88.7 \pm 0.1	94.1 \pm 3.0	94.2 \pm 2.9	89.6 \pm 0.2	90.5 \pm 0.2	90.5 \pm 0.2	90.5 \pm 0.1	94.3 \pm 2.9	95.5 \pm 1.5
10	84.8 \pm 1.9	85.6 \pm 2.1	86.1 \pm 1.8	87.1 \pm 1.1	91.0 \pm 1.4	94.7 \pm 1.4	86.8 \pm 2.0	88.7 \pm 1.3	88.2 \pm 1.6	88.2 \pm 1.6	93.0 \pm 1.4	95.3 \pm 1.5
11	87.9 \pm 0.2	88.7 \pm 0.2	88.8 \pm 0.1	88.8 \pm 0.1	95.9 \pm 1.5	95.9 \pm 1.5	89.8 \pm 0.1	90.6 \pm 0.2	90.7 \pm 0.2	90.8 \pm 0.2	95.9 \pm 1.4	96.8 \pm 1.5
12	87.8 \pm 0.3	88.7 \pm 0.4	88.7 \pm 0.2	88.7 \pm 0.3	95.9 \pm 1.5	96.0 \pm 1.4	89.6 \pm 0.2	90.3 \pm 0.3	90.7 \pm 0.3	90.7 \pm 0.2	95.8 \pm 1.5	96.4 \pm 1.5
13	84.3 \pm 2.4	84.0 \pm 3.1	84.5 \pm 2.7	82.8 \pm 3.8	79.6 \pm 2.4	84.1 \pm 2.6	85.1 \pm 2.9	86.9 \pm 2.4	84.5 \pm 4.0	84.4 \pm 4.0	75.3 \pm 2.3	88.9 \pm 2.8
14	87.3 \pm 0.3	88.2 \pm 0.4	88.2 \pm 0.4	88.3 \pm 0.4	98.6 \pm 0.1	95.0 \pm 1.4	89.2 \pm 0.3	90.1 \pm 0.4	90.0 \pm 0.5	90.0 \pm 0.5	98.6 \pm 0.1	93.8 \pm 1.5
Novel	1.8 \pm 1.0	2.5 \pm 1.3	3.5 \pm 1.8	47.0 \pm 3.7	44.0 \pm 1.9	53.3 \pm 1.6	1.7 \pm 0.9	2.4 \pm 1.3	3.5 \pm 1.8	47.0 \pm 3.7	44.0 \pm 1.8	54.6 \pm 1.5
OA	79.2 \pm 3.5	80.6 \pm 3.3	80.7 \pm 3.2	83.0 \pm 2.9	85.0 \pm 2.1	86.8\pm2.3	81.3 \pm 3.4	82.4 \pm 3.3	83.6 \pm 2.5	85.0 \pm 2.7	86.1 \pm 2.2	88.2\pm1.8
AA	79.1 \pm 2.0	80.2 \pm 1.9	80.3 \pm 1.9	83.4 \pm 1.9	86.3 \pm 2.1	88.5\pm2.0	81.0 \pm 1.9	81.9 \pm 1.9	82.3 \pm 1.7	85.2 \pm 1.9	87.1 \pm 2.2	88.8\pm1.4
F1	88.3 \pm 2.2	89.2 \pm 2.0	89.2 \pm 2.0	89.6 \pm 1.7	91.0 \pm 1.3	91.8\pm1.4	89.6 \pm 2.0	90.2 \pm 2.0	91.0 \pm 1.5	90.9 \pm 1.5	91.6 \pm 1.3	92.6\pm1.1

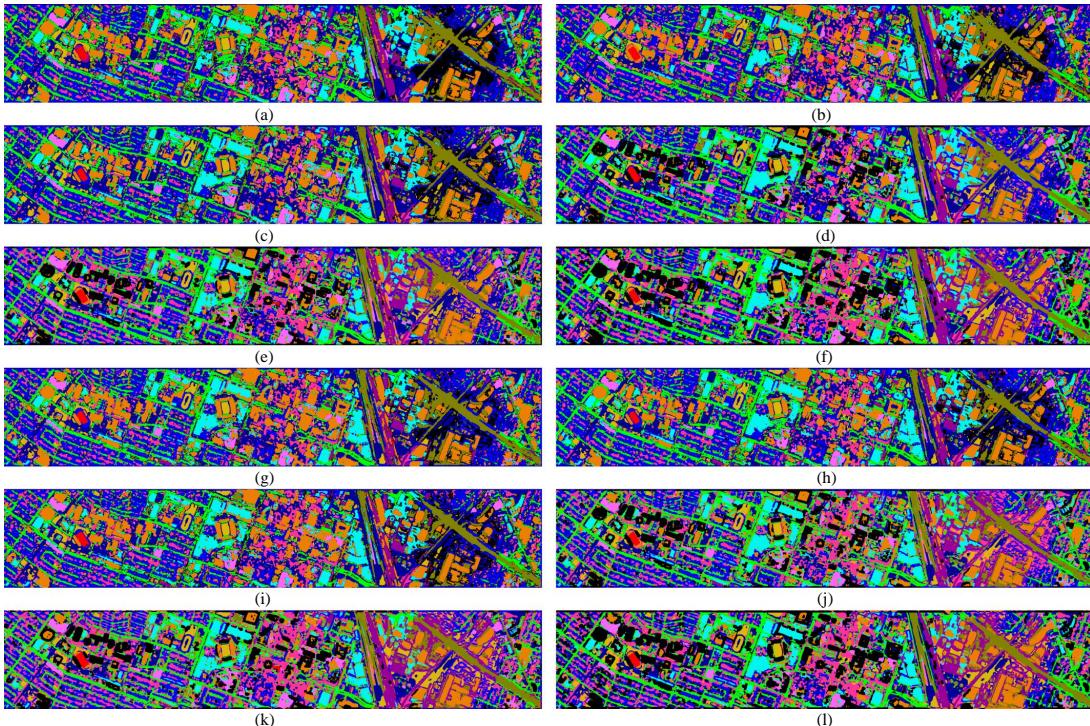


Fig. 10: The classification results of the HU dataset. (a) CDCNN ($L = 20$). (b) 3D-CNN ($L = 20$). (c) DFSL ($L = 20$). (d) CROSRR ($L = 20$). (e) MDL4OW ($L = 20$). (f) The proposed SSLR ($L = 20$). (g) CDCNN ($L = 30$). (h) 3D-CNN ($L = 30$). (i) DFSL ($L = 30$). (j) CROSRR ($L = 30$). (k) MDL4OW ($L = 30$). (l) The proposed SSLR ($L = 30$).

In order to verify the effectiveness of the proposed method in complex scene with open-set setting, comparative experiments are conducted on Houston 2018 dataset [73]. The spatial resolution and image size of the Houston 2018 dataset are 1m and 601 \times 2384. For the HSI, the wavelength range is 0.38 μ m to 1.05 μ m. The number of spectral channels corresponding to the HSI is 48. This dataset contains 20 object classes, such as

roads, sidewalks and railways. We follow the previous study [74] to select the training and testing samples. In the open-set setting of Houston 2018 dataset, three classes (i.e., stadium seats, paved parking lots and unpaved parking lots) are selected as the unknown open class, and the remaining 17 classes are selected as the known classes.

The open OA, open AA and F1 score of the proposed

TABLE VI: Classification results (%) by using 20 and 30 labeled samples for both known and unknown classes on the HU dataset.

Class	$L = 20$						$L = 30$					
	CDCNN	3D-CNN	DFSL	CROSR	MDL4OW	Ours	CDCNN	3D-CNN	DFSL	CROSR	MDL4OW	Ours
1	88.9±4.2	90.8±3.8	91.0±3.6	88.6±4.2	87.2±1.2	87.9±0.9	83.7±3.3	90.0±3.1	93.0±3.1	90.7±3.5	88.9±0.5	89.6±0.4
2	92.6±4.4	93.0±3.7	94.1±3.7	93.0±4.4	88.8±0.2	89.0±0.3	93.3±4.1	94.4±3.6	94.8±3.2	92.0±3.7	89.1±0.3	89.8±0.4
3	93.2±4.5	94.0±3.8	94.4±3.7	93.2±4.4	89.3±0.4	89.2±0.3	93.6±4.0	94.3±3.6	95.1±3.2	92.8±3.6	89.3±0.2	90.1±0.1
4	91.9±4.2	93.1±3.8	88.9±3.4	87.5±4.1	86.6±1.4	86.6±1.8	88.0±3.7	88.8±3.1	94.0±3.1	90.9±3.8	87.4±1.3	87.9±1.3
5	92.3±4.3	92.5±4.0	93.2±3.6	93.0±4.4	89.1±0.3	88.9±0.2	92.5±4.0	93.6±3.6	94.7±3.2	92.8±3.7	89.5±0.2	90.0±0.3
6	86.2±3.9	85.5±3.4	87.4±3.2	83.3±3.9	86.5±1.6	87.2±1.4	86.9±3.5	89.1±3.5	87.9±2.9	87.2±3.5	87.6±1.2	88.2±1.1
7	65.5±2.3	68.1±2.7	72.0±2.5	68.7±3.3	79.9±5.7	81.7±4.8	84.6±3.4	75.7±2.5	75.1±2.4	72.4±2.9	79.4±6.5	83.2±4.4
8	80.7±3.7	82.5±3.3	74.0±2.6	81.0±3.4	85.4±2.0	85.4±2.1	77.9±2.9	85.0±3.1	76.0±2.1	85.2±3.4	83.9±3.8	88.1±1.3
9	90.6±4.2	88.6±3.6	92.6±3.5	92.2±4.2	88.9±0.3	88.8±0.3	90.5±3.9	92.4±3.6	92.6±2.9	91.4±3.6	88.8±0.6	89.8±0.3
10	78.8±3.2	76.8±3.0	78.3±3.0	79.3±3.6	82.4±4.4	86.6±1.5	79.6±3.3	80.9±2.8	80.4±2.3	81.7±3.0	87.2±1.4	86.5±2.1
11	84.6±3.7	88.2±3.6	92.9±3.4	90.9±4.3	87.6±0.9	88.6±0.3	91.0±4.1	90.8±3.3	94.0±3.0	88.4±3.6	87.9±0.9	89.2±0.6
Novel	6.9±3.8	8.2±4.4	5.7±2.1	35.5±2.0	39.5±2.1	54.2±1.5	8.6±3.6	6.9±3.8	5.6±2.1	34.2±2.1	40.2±2.0	55.2±1.5
OA	75.8±2.9	76.6±2.5	76.7±2.6	79.8±3.3	80.5±1.7	83.0±1.2	77.3±2.7	78.2±2.3	78.2±2.2	80.9±2.8	81.2±1.6	84.2±1.1
AA	79.4±3.2	80.1±2.8	80.4±2.8	82.2±3.5	82.6±1.6	84.5±1.2	80.9±3.0	81.8±2.6	81.9±2.4	83.3±3.0	83.3±1.5	85.6±1.1
F1	83.6±2.2	84.0±2.0	85.3±1.8	85.9±2.3	86.1±0.9	86.5±0.7	84.5±2.0	85.2±1.8	85.3±1.6	86.0±1.9	86.5±0.9	87.2±0.6

method and the comparison methods are reported in Table VII. Compared with the MDL4OW method, the open OA and AA of the proposed method are increased by 2.1% and 2.3%, respectively. Compared with the SpectralFormer method and the GCN method, the open OA of SSLR is increased by 7% and 11.4%, and the open AA of SSLR is increased by 7.1% and 11.5%, respectively. For the F1 score, compared with the SpectralFormer method and the GCN method, the F1 of SSLR is increased by 6.4% and 9.4%, respectively. From the above results, it can be seen that the proposed method achieves the best performance on the Houston 2018 dataset compared with the open-set methods and the traditional HSI classification methods.

TABLE VII: Classification results (%) for both known and unknown classes on the Houston 2018 dataset.

Methods	OA	AA	F1
CDCNN	65.7±4.0	64.7±4.2	68.5±3.8
3D-CNN	67.3±3.9	66.5±4.1	70.4±3.7
GCN	69.5±3.4	68.7±3.6	73.2±3.1
DFSL	72.1±3.2	70.9±3.5	73.5±2.9
SpectralFormer	73.9±2.9	73.1±3.1	76.2±2.7
CROSR	77.2±2.8	75.9±3.3	78.7±2.6
MDL4OW	78.8±2.7	77.9±3.1	80.3±2.5
Ours	80.9±2.3	80.2±2.5	82.6±2.1

B. Ablation Study

To verify the effectiveness of each module of the proposed SSLR method under open-set setting, we conducted an ablation study on three datasets (i.e. UP dataset, SA dataset and HU dataset). In the experimental setup of this ablation study, SSLR without the spectral latent reconstruction, SSLR without the spatial latent reconstruction and SSLR without the progressive residual convolution module were compared with our proposed method. The relevant accuracy indicators and

TABLE VIII: Results of ten runs for the UP, SA and HU datasets in the ablation study.

L	UP		SA		HU	
	$L = 20$	$L = 30$	$L = 20$	$L = 30$	$L = 20$	$L = 30$
The proposed SSLR						
OA	87.3±1.8	89.1±1.8	86.8±2.3	88.2±1.8	83.0±1.2	84.2±1.1
AA	84.4±1.1	87.8±1.4	88.5±2.0	88.8±1.4	84.5±1.2	85.6±1.1
F1	92.5±1.1	93.4±1.0	91.8±1.4	92.6±1.1	86.5±0.7	87.2±0.6
SSLR without spectral latent reconstruction						
OA	85.8±2.2	87.9±2.0	85.2±2.4	86.4±2.3	80.3±2.0	81.5±2.1
AA	83.1±2.1	83.8±1.8	86.7±2.4	87.3±2.4	81.6±1.9	81.6±2.0
F1	89.7±1.9	91.5±1.7	90.2±1.7	91.0±1.6	83.9±1.7	84.4±1.9
SSLR without spatial latent reconstruction						
OA	86.3±2.3	88.2±2.1	85.5±2.7	86.8±2.2	80.5±1.9	81.7±2.0
AA	83.9±2.1	84.2±2.0	86.8±2.6	87.5±2.5	81.8±2.1	81.9±1.8
F1	89.0±1.9	91.8±1.9	90.7±1.9	91.4±1.8	84.2±1.6	84.8±1.7
SSLR without progressive residual convolution module						
OA	86.9±2.1	88.4±1.9	86.1±2.9	87.0±2.2	80.7±1.8	81.8±1.7
AA	83.8±1.6	84.7±1.7	87.2±2.5	87.9±2.6	82.1±1.7	82.3±1.6
F1	91.2±1.5	92.1±1.4	91.1±2.0	91.8±1.7	84.7±1.5	85.2±1.8

results of the ablation study are shown in Table VIII. It can be seen from the table that each module proposed in this paper plays a very important role on improving the accuracy.

For the UP dataset, when the spectral latent reconstruction is removed, in the case of $L = 20$, the open OA of SSLR is reduced by 1.5%. When the spatial latent reconstruction is removed, the open OA of SSLR is reduced by 1% ($L = 20$). After removing the progressive residual convolution module, the open OA of SSLR decreases by 0.4% ($L = 20$). In the case of $L = 30$, when the spectral latent reconstruction is removed, the open OA, open AA and F1 score of SSLR are reduced by 1.2%, 4% and 1.9%.

For the SA dataset, when the spatial latent reconstruction is removed, the open OA, open AA and F1 score of SSLR

are reduced by 1.4%, 1.3% and 1.2% ($L = 30$), respectively. After removing the progressive residual convolution module, the open OA, open AA and F1 score of are reduced by 1.2%, 0.9% and 0.8% ($L = 30$), respectively. When the spectral latent reconstruction is removed, in the case of $L = 20$, the open OA of SSLR is reduced by 1.6%. When the spatial latent reconstruction is removed, the open OA of SSLR is reduced by 1.3% ($L = 20$). After removing the progressive residual convolution module, the open OA of SSLR decreases by 0.7% ($L = 20$).

For the HU dataset, when the spectral latent reconstruction is removed, in the case of $L = 20$, the open OA of SSLR is reduced by 2.7%. When the spatial latent reconstruction is removed, the open OA of SSLR is reduced by 2.5% ($L = 20$). After removing the progressive residual convolution module, the open OA of SSLR decreases by 2.3% ($L = 20$). In the case of $L = 30$, when the spectral latent reconstruction is removed, the open OA, open AA and F1 score of SSLR are reduced by 2.7%, 4% and 2.8%. When the spatial latent reconstruction is removed, the open OA, open AA and F1 score of SSLR are reduced by 2.5%, 3.7% and 2.4% ($L = 30$), respectively.

C. Computational Cost

In this paper, we report the training times and inference times of the proposed method and the comparison methods in Table IX on the UP dataset. In this table, we can see that the training time and inference time of this method are less than those of the comparison methods. Specifically, when $L = 30$, compared with the CDCNN method, the training time is reduced from 113.76s to 65.94s. When $L = 20$, compared with the MDL4OW method, the inference time is reduced from 32.28s to 30.87s.

TABLE IX: Training and inference time (s) on the UP dataset.

Methods	Training time		Inference time	
	$L = 20$	$L = 30$	$L = 20$	$L = 30$
CDCNN	96.87	113.76	32.70	32.89
3D-CNN	101.29	123.47	33.54	34.45
DFSL	96.94	115.45	44.59	44.28
CROSR	67.93	78.84	43.23	42.91
MDL4OW	60.98	89.87	32.28	31.98
Ours	58.76	65.94	30.87	30.14

V. CONCLUSION

Despite the method based on deep learning has achieved success in HSI classification, it still lacks robustness in dealing with unknown classes (those other than known classes) in OSE. The existing open-set classifiers rely on the network trained using the known samples in the training set, which may cause the learned features to retain more information that can help the classification of known classes, and it is likely to ignore the information that can be used to distinguish between known and unknown classes. In order to improve the robustness of HSI classification method in OSE and maintain

its classification accuracy in CSE, a spectral spatial latent reconstruction framework for spectral feature reconstruction, spatial feature reconstruction and pixel by pixel classification in OSE is proposed. By reconstructing the spectral and spatial features of HSI, the learned spectral-spatial features are enhanced, thereby retaining more effective spectral-spatial information, which can be used to separate unknown classes from known classes and classify known classes. Comprehensive comparison experiments show that the performance of the proposed method is better than existing deep open-set classifiers in multiple HSI datasets. Next, we will continue to explore other hyperspectral data processing tasks in the OSE (such as object detection and instance segmentation), and explore how to achieve end-to-end open-set HSI classification.

REFERENCES

- [1] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. A. Benediktsson, “Deep learning for hyperspectral image classification: An overview,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 9, pp. 6690–6709, 2019.
- [2] L. Zhang, L. Zhang, and B. Du, “Deep learning for remote sensing data: A technical tutorial on the state of the art,” *IEEE Geoscience and Remote Sensing Magazine*, vol. 4, no. 2, pp. 22–40, 2016.
- [3] X. X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, F. Xu, and F. Fraundorfer, “Deep learning in remote sensing: A comprehensive review and list of resources,” *IEEE Geoscience and Remote Sensing Magazine*, vol. 5, no. 4, pp. 8–36, 2017.
- [4] J. A. Benediktsson and P. Ghamisi, *Spectral-spatial classification of hyperspectral remote sensing images*. Artech House, 2015.
- [5] P. Ghamisi, J. Plaza, Y. Chen, J. Li, and A. J. Plaza, “Advanced spectral classifiers for hyperspectral images: A review,” *IEEE Geoscience and Remote Sensing Magazine*, vol. 5, no. 1, pp. 8–32, 2017.
- [6] J. Song, S. Xia, J. Wang, M. Patel, and D. Chen, “Uncertainty quantification of hyperspectral image denoising frameworks based on sliding-window low-rank matrix approximation,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–12, 2022.
- [7] Q. Du and H. Ren, “Real-time constrained linear discriminant analysis to target detection and classification in hyperspectral imagery,” *Pattern Recognition*, vol. 36, no. 1, pp. 1–12, 2003.
- [8] L. Xie, M. Yin, X. Yin, Y. Liu, and G. Yin, “Low-rank sparse preserving projections for dimensionality reduction,” *IEEE Transactions on Image Processing*, vol. 27, no. 11, pp. 5261–5274, 2018.
- [9] Y. W. Chen and X. H. Han, *Classification of High-Resolution Satellite Images Using Supervised Locality Preserving Projections*. Springer Berlin Heidelberg, 2008.
- [10] W. Li, S. Prasad, J. E. Fowler, and L. M. Bruce, “Locality-preserving discriminant analysis in kernel-induced feature spaces for hyperspectral image classification,” *IEEE Geoscience & Remote Sensing Letters*, vol. 8, no. 5, pp. 894–898, 2011.
- [11] A. Plaza, P. Martinez, J. Plaza, and R. Perez, “Dimensionality reduction and classification of hyperspectral image data using sequences of extended morphological transformations,” *IEEE Transactions on Geoscience & Remote Sensing*, vol. 43, no. 3, pp. 466–479, 2005.
- [12] W. Liao, R. Bellens, A. Pizurica, W. Philips, and Y. Pi, “Classification of hyperspectral data over urban areas using directional morphological profiles and semi-supervised feature extraction,” *IEEE Journal of Selected Topics in Applied Earth Observations & Remote Sensing*, vol. 5, no. 4, pp. 1177–1190, 2012.
- [13] P. R. Marpu, M. Pedernana, M. D. Mura, J. A. Benediktsson, and L. Bruzzone, “Automatic generation of standard deviation attribute profiles for spectral-spatial classification of remote sensing data,” *IEEE Geoscience & Remote Sensing Letters*, vol. 10, no. 2, pp. 293–297, 2013.
- [14] G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *science*, vol. 313, no. 5786, pp. 504–507, 2006.
- [15] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.
- [16] S. Wang, J. Yue, J. Liu, Q. Tian, and M. Wang, “Large-scale few-shot learning via multi-modal knowledge discovery,” in *European Conference on Computer Vision*. Springer, 2020, pp. 718–734.

- [17] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2018.
- [18] Z. Tian, H. Zhao, M. Shu, Z. Yang, R. Li, and J. Jia, “Prior guided feature enrichment network for few-shot segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2020.
- [19] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Region-based convolutional networks for accurate object detection and segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 1, pp. 142–158, 2016.
- [20] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.
- [21] R. Dian, S. Li, A. Guo, and L. Fang, “Deep hyperspectral image sharpening,” *IEEE transactions on neural networks and learning systems*, vol. 29, no. 11, pp. 5345–5355, 2018.
- [22] J. Ma, W. Yu, C. Chen, P. Liang, X. Guo, and J. Jiang, “Pan-gan: An unsupervised pan-sharpening method for remote sensing image fusion,” *Information Fusion*, vol. 62, pp. 110–120, 2020.
- [23] J. Yue, D. Zhu, L. Fang, P. Ghamisi, and Y. Wang, “Adaptive spatial pyramid constraint for hyperspectral image classification with limited training samples,” *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–14, 2021.
- [24] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, “Deep learning-based classification of hyperspectral data,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 6, pp. 2094–2107, 2014.
- [25] J. Yue, S. Mao, and M. Li, “A deep learning framework for hyperspectral image classification using spatial pyramid pooling,” *Remote Sensing Letters*, vol. 7, no. 9, pp. 875–884, 2016.
- [26] J. Yue, L. Fang, H. Rahmani, and P. Ghamisi, “Self-supervised learning with adaptive distillation for hyperspectral image classification,” *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–13, 2021.
- [27] X. Ma, A. Fu, J. Wang, H. Wang, and B. Yin, “Hyperspectral image classification based on deep deconvolution network with skip architecture,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 8, pp. 4781–4791, 2018.
- [28] L. Fang, S. Li, X. Kang, and J. A. Benediktsson, “Spectral–spatial classification of hyperspectral images with a superpixel-based discriminative sparse model,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 8, pp. 4186–4201, 2015.
- [29] Z. Zhong, J. Li, Z. Luo, and M. Chapman, “Spectral–spatial residual network for hyperspectral image classification: A 3-d deep learning framework,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 2, pp. 847–858, 2018.
- [30] Y. Xu, Z. Li, W. Li, Q. Du, C. Liu, Z. Fang, and L. Zhai, “Dual-channel residual network for hyperspectral image classification with noisy labels,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–11, 2022.
- [31] D. Hong, Z. Han, J. Yao, L. Gao, B. Zhang, A. Plaza, and J. Chanussot, “Spectralformer: Rethinking hyperspectral image classification with transformers,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.
- [32] L. Mou, P. Ghamisi, and X. X. Zhu, “Deep recurrent neural networks for hyperspectral image classification,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 3639–3655, 2017.
- [33] A. Bendale and T. E. Boult, “Towards open set deep networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1563–1572.
- [34] R. P. Duin and E. Pekalska, “Open issues in pattern recognition,” in *Computer Recognition Systems*. Springer, 2005, pp. 27–42.
- [35] J. McCarthy and P. J. Hayes, “Some philosophical problems from the standpoint of artificial intelligence,” in *Readings in artificial intelligence*. Elsevier, 1981, pp. 431–450.
- [36] M. J. Wilber, W. J. Scheirer, P. Leitner, B. Heflin, J. Zott, D. Reinke, D. K. Delaney, and T. E. Boult, “Animal recognition in the mojave desert: Vision tools for field biologists,” in *2013 IEEE Workshop on Applications of Computer Vision (WACV)*. IEEE, 2013, pp. 206–213.
- [37] N. Sünderhauf, O. Brock, W. Scheirer, R. Hadsell, D. Fox, J. Leitner, B. Upcroft, P. Abbeel, W. Burgard, M. Milford *et al.*, “The limits and potentials of deep learning for robotics,” *The International Journal of Robotics Research*, vol. 37, no. 4–5, pp. 405–420, 2018.
- [38] C. Geng, S.-j. Huang, and S. Chen, “Recent advances in open set recognition: A survey,” *IEEE transactions on pattern analysis and machine intelligence*, 2020.
- [39] P. Oza and V. M. Patel, “C2ae: Class conditioned auto-encoder for open-set recognition,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2307–2316.
- [40] R. Yoshihashi, W. Shao, R. Kawakami, S. You, M. Iida, and T. Naemura, “Classification-reconstruction learning for open-set recognition,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4016–4025.
- [41] Y. Liu, Y. Tang, L. Zhang, L. Liu, M. Song, K. Gong, Y. Peng, J. Hou, and T. Jiang, “Hyperspectral open set classification with unknown classes rejection towards deep networks,” *International Journal of Remote Sensing*, vol. 41, no. 16, pp. 6355–6383, 2020.
- [42] M. A. Pimentel, D. A. Clifton, L. Clifton, and L. Tarassenko, “A review of novelty detection,” *Signal Processing*, vol. 99, pp. 215–249, 2014.
- [43] A. Ghosh, S. Parthasarathy, and M. E. Otey, “Fast mining of distance-based outliers in high-dimensional datasets,” *Data Mining and Knowledge Discovery*, vol. 16, no. 3, pp. 349–364, 2008.
- [44] S.-P. Yong, J. D. Deng, and M. K. Purvis, “Novelty detection in wildlife scenes through semantic context modelling,” *Pattern Recognition*, vol. 45, no. 9, pp. 3439–3450, 2012.
- [45] D. A. Clifton, S. Hugueny, and L. Tarassenko, “Novelty detection with multivariate extreme value statistics,” *Journal of signal processing systems*, vol. 65, no. 3, pp. 371–389, 2011.
- [46] K. Yamanishi, J.-I. Takeuchi, G. Williams, and P. Milne, “On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms,” *Data Mining and Knowledge Discovery*, vol. 8, no. 3, pp. 275–300, 2004.
- [47] S. Ntalampiras, I. Potamitis, and N. Fakotakis, “Probabilistic novelty detection for acoustic surveillance under real-world conditions,” *IEEE Transactions on Multimedia*, vol. 13, no. 4, pp. 713–719, 2011.
- [48] M. Wu and J. Ye, “A small sphere and large margin approach for novelty detection using training data with outliers,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 11, pp. 2088–2092, 2009.
- [49] A. B. Gardner, A. M. Krieger, G. Vachtsevanos, B. Litt, and L. P. Kaelbing, “One-class novelty detection for seizure analysis from intracranial eeg,” *Journal of Machine Learning Research*, vol. 7, no. 6, 2006.
- [50] M. K. Albertini and R. F. de Mello, “A self-organizing neural network for detecting novelties,” in *Proceedings of the 2007 ACM symposium on Applied computing*, 2007, pp. 462–466.
- [51] H. Hoffmann, “Kernel pca for novelty detection,” *Pattern recognition*, vol. 40, no. 3, pp. 863–874, 2007.
- [52] R. Kassab and F. Alexandre, “Incremental data-driven learning of a novelty detection model for one-class classification with application to high-dimensional noisy data,” *Machine learning*, vol. 74, no. 2, pp. 191–234, 2009.
- [53] Y. Li, M. J. Pont, and N. B. Jones, “Improving the performance of radial basis function classifiers in condition monitoring and fault diagnosis applications where ‘unknown’ faults may occur,” *Pattern Recognition Letters*, vol. 23, no. 5, pp. 569–577, 2002.
- [54] M. Filippone and G. Sanguinetti, “Information theoretic novelty detection,” *Pattern Recognition*, vol. 43, no. 3, pp. 805–814, 2010.
- [55] H.-M. Yang, X.-Y. Zhang, F. Yin, Q. Yang, and C.-L. Liu, “Convolutional prototype network for open set recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.
- [56] W. J. Scheirer, A. de Rezende Rocha, A. Sapkota, and T. E. Boult, “Toward open set recognition,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 7, pp. 1757–1772, 2012.
- [57] L. Shu, H. Xu, and B. Liu, “Doc: Deep open classification of text documents,” *arXiv preprint arXiv:1709.08716*, 2017.
- [58] Z. Ge, S. Demyanov, Z. Chen, and R. Garnavi, “Generative openmax for multi-class open set classification,” *arXiv preprint arXiv:1707.07418*, 2017.
- [59] E. M. Rudd, L. P. Jain, W. J. Scheirer, and T. E. Boult, “The extreme value machine,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 3, pp. 762–768, 2018.
- [60] W. J. Scheirer, L. P. Jain, and T. E. Boult, “Probability models for open set recognition,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 11, pp. 2317–2324, 2014.
- [61] L. Neal, M. Olson, X. Fern, W.-K. Wong, and F. Li, “Open set learning with counterfactual images,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 613–628.
- [62] Y. Zhang, K. Lee, and H. Lee, “Augmenting supervised neural networks with unsupervised objectives for large-scale image classification,” in *International conference on machine learning*. PMLR, 2016, pp. 612–621.

- [63] A. Dosovitskiy and T. Brox, "Inverting visual representations with convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4829–4837.
- [64] J. Yue, L. Fang, P. Ghamisi, W. Xie, J. Li, J. Chanussot, and A. J. Plaza, "Optical remote sensing image understanding with weak supervision: Concepts, methods, and perspectives," *IEEE Geoscience and Remote Sensing Magazine*, pp. 2–21, 2022.
- [65] S. Liu, Q. Shi, and L. Zhang, "Few-shot hyperspectral image classification with unknown classes using multitask deep learning," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–18, 2020.
- [66] J. Yue, W. Zhao, S. Mao, and H. Liu, "Spectral-spatial classification of hyperspectral images using deep convolutional neural networks," *Remote Sensing Letters*, vol. 6, no. 6, pp. 468–477, 2015.
- [67] B. Liu, X. Yu, A. Yu, P. Zhang, G. Wan, and R. Wang, "Deep few-shot learning for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 4, pp. 2290–2304, 2019.
- [68] C. Zhang, J. Yue, and Q. Qin, "Global prototypical network for few-shot hyperspectral image classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, pp. 1–1, 2020.
- [69] A. B. Hamida, A. Benoit, P. Lambert, and C. B. Amar, "3-d deep learning approach for remote sensing image classification," *IEEE Transactions on geoscience and remote sensing*, vol. 56, no. 8, pp. 4420–4434, 2018.
- [70] H. Lee and H. Kwon, "Going deeper with contextual cnn for hyperspectral image classification," *IEEE Transactions on Image Processing*, vol. 26, no. 10, pp. 4843–4855, 2017.
- [71] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021.
- [72] D. Hong, L. Gao, J. Yao, B. Zhang, A. Plaza, and J. Chanussot, "Graph convolutional networks for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 7, pp. 5966–5978, 2021.
- [73] Y. Xu, B. Du, L. Zhang, D. Cerra, M. Pato, E. Carmona, S. Prasad, N. Yokoya, R. Hänsch, and B. Le Saux, "Advanced multi-sensor optical remote sensing for urban land use and land cover classification: Outcome of the 2018 ieee grss data fusion contest," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 6, pp. 1709–1724, 2019.
- [74] R. Hang, Z. Li, Q. Liu, P. Ghamisi, and S. S. Bhattacharyya, "Hyperspectral image classification with attention-aided cnns," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 3, pp. 2281–2293, 2021.



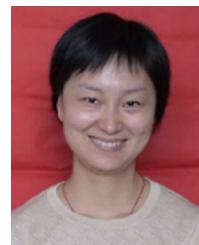
Jun Yue received the B.Eng. degree in geodesy from Wuhan University, Wuhan, China, in 2013 and the Ph.D. degree in GIS from Peking University, Beijing, China, in 2018.

He is currently an Assistant Professor with the Department of Geomatics Engineering, Changsha University of Science and Technology. His research interests include satellite image understanding, pattern recognition, and few-shot learning. Dr. Yue serves as a reviewer for IEEE Transactions on Image Processing, IEEE Transactions on Neural Networks and Learning Systems, IEEE Transactions on Geoscience and Remote Sensing, ISPRS Journal of Photogrammetry and Remote Sensing, IEEE Geoscience and Remote Sensing Letters, IEEE Transactions on Biomedical Engineering, Information Fusion, Information Sciences, etc.



Leyuan Fang (S'10-M'14-SM'17) received the Ph.D. degree from the College of Electrical and Information Engineering, Hunan University, Changsha, China, in 2015.

From August 2016 to September 2017, he was a Postdoc Researcher with the Department of Biomedical Engineering, Duke University, Durham, NC, USA. He is currently a Professor with the College of Electrical and Information Engineering, Hunan University. His research interests include sparse representation and multi-resolution analysis in remote sensing and medical image processing. He is the associate editors of IEEE Transactions on Image Processing, IEEE Transactions on Geoscience and Remote Sensing, IEEE Transactions on Neural Networks and Learning Systems, and Neurocomputing. He was a recipient of one 2nd-Grade National Award at the Nature and Science Progress of China in 2019.



Min He received the B.Eng. degree from Xiangtan University, Xiangtan, China, in 1999. She received the M.Sc. degree in circuits and systems and the Ph.D. degree in automatic control science and engineering from Hunan University, Changsha, China, in 2003 and 2011, respectively.

Since 2003, she has worked with the College of Electrical and Information Engineering, Hunan University. Her research interests include AI algorithms, especially in image analysis and pattern recognition.