Dual-Branch Spectral–Spatial Adversarial Representation Learning for Hyperspectral Image Classification With Few Labeled Samples

Caihao Sun, Xiaohua Zhang, *Member, IEEE*, Hongyun Meng, Xianghai Cao, *Member, IEEE*, Jinhua Zhang, and Licheng Jiao, *Fellow, IEEE*

***Abstract*—Recently, deep learning methods, particularly the convolutional neural networks, have been extensively employed for extracting spectral–spatial features in hyperspectral image (HSI) classification tasks, yielding promising results. Conventional methods often use small image patches as input and combine spectral and spatial features with fixed strategies. However, the equal treatment of all pixels within heterogeneous patches can negatively impact feature extraction performance. In this article, we propose a semisupervised dual-branch spectral–spatial adversarial representation learning (SSARL) method for HSI classification. SSARL adaptively assigns attention weights to different pixels and adds a spectral constraint to spatial features. Our approach consists of three main components: 1) a dual-branch framework designed to independently extract spectral and spatial information from pixel and patch samples; 2) a class consistency loss that adaptively combines spectral and spatial classification results, mitigating the adverse effects of heterogeneous patches and enabling appropriate feature selection for various situations; and 3) the deep learning model on the labeled sample size by adding the adversarial representation module and conditional entropy to two branches, reducing the deep learning model's reliance on labeled sample size. Experimental results demonstrate that SSARL outperforms competitive methods on small-sized (0.3%–5%) labeled samples and exhibits superior performance for boundary test pixels.**

***Index Terms*—Adversarial network, class consistency loss, dual branch, generative adversarial network (GAN), hyperspectral image (HSI) classification, semisupervised, spectral–spatial feature.**

I. Introduction

HYPERSPECTRAL imaging is a type of remote sensing technology that captures abundant spectral and spatial information. Unlike conventional RGB images, hyperspectral images (HSIs) are 3-D form of images, which enable a wide range of applications [1], [2], including modern agriculture [3], aviation industry [4], security [5], and biomedicine [6]. HSI classification, an essential process in remote sensing, discriminates ground objects with unique spectral characteristics. Although HSIs contain a large number of spectral bands that provide rich information, they also introduce redundancy and noise [7], [8]. Consequently, researchers have focused on effective feature extraction methods. Traditional supervised methods [9] typically transform high-dimensional data into low-dimensional features and design manual features based on prior knowledge [10]. However, features obtained through traditional methods rely heavily on expert experience, which often results in low classification accuracy for practical applications.

Recently, deep-learning-based methods have demonstrated improved performance in HSI classification due to their powerful feature extraction capabilities. They automatically extract deep and discriminative features, overcoming the limitations of traditional methods. Examples include stacked autoencoders (SAEs) [11], deep belief networks (DBNs) [12], [13], [14], convolutional neural networks (CNNs) [15], and generative adversarial networks (GANs) [16], [17], [18], [19]. The aforementioned methods primarily extract spectral features from individual hyperspectral pixels. In addition, numerous studies have shown that incorporating spatial information into classifiers can effectively enhance performance [20]. Spatial features address two key challenges: 1) high-dimensional spectral features not only contain abundant information but also introduce redundancy and noise—by operating on all the pixels within an image patch and extracting features, noise and errors can be effectively reduced; and 2) the same land cover types often exhibit distinct spatial structures, while within-class spectral differences can lead to variations in spectral–spatial features exploiting the correlation between neighboring pixels within a patch, thus mitigating the impact of spectral changes [21]. The core concept of spatial features involves fusing features from all the pixels within an image patch and treating them as central pixel features. Utilizing patch samples and designing spatially structured models are approaches to obtain spatial features at present. Yang et al. [22] designed a two-channel CNN structure, with one channel for spectral feature extraction and another for spatial feature extraction. Two types of features are concatenated and sent to a fully connected layer. Li et al. [23] used the

Manuscript received 1 April 2023; revised 20 May 2023 and 16 June 2023; accepted 21 June 2023. Date of publication 29 June 2023; date of current version 12 July 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 61877066, in part by the Aero-Science Fund under Grant 20175181013, and in part by the Science and Technology Plan Project of Xi'an under Grant 21RGZN0010. *(Corresponding author: Xiaohua Zhang.)*

Caihao Sun, Xiaohua Zhang, Xianghai Cao, Jinhua Zhang, and Licheng Jiao are with the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, School of Artificial Intelligence, Xidian University, Xi'an 710071, China (e-mail: caihaosun@stu.xidian.edu.cn; xh_zhang@mail.xidian.edu.cn; caoxh@xidian.edu.cn; jhzhangcs@stu.xidian.edu.cn; lchjiao@mail.xidian.edu.cn).

Hongyun Meng is with the School of Mathematics and Statistics, Xidian University, Xi'an 710071, China (e-mail: menghy@xidian.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2023.3290678

3-D CNN to directly extract spectral–spatial features from HSI patch samples. Similarly, the discriminator of the 3-D GAN [24] can classify samples and determine their authenticity based on extracted spectral–spatial features. Fang et al. [25] proposed a new multiclass GAN that combines spectral and spatial features. To reduce model complexity and enhance spatial feature abstraction, HybridSN proposed by Roy et al. [26] consists of 3-D CNN and 2-D CNN layers. An image patch sample passes through three 3-D CNN layers and one 2-D CNN layer successively to obtain a spectral–spatial joint feature. Moreover, models based on attention mechanisms [27], [28], [29] can extract global features of images. SSFTT [30] first inputs 3-D image patches to a CNN, and the output feature maps are divided into semantic patches. These patches are then input to a transformer-based encoder. Deep learning HSI classifiers that utilize spectral–spatial features and patch samples have achieved impressive results. However, deep-learning-based methods typically require a large number of labeled training samples to optimize the abundant parameters of deep models and avoid overfitting. In addition, spatial features also have inherent shortcomings, which will be analyzed in detail in the following.

For HSI spectral–spatial classification with a small number of labeled samples, semisupervised learning is considered a promising approach. Semisupervised learning aims to extract information from a large number of unlabeled samples. Sun et al. [31] proposed a semisupervised algorithm that combines clustering and manifold techniques. Seydgar et al. [32] designed a semisupervised framework capable of generating reliable fake labels, which are effective for various deep learning models. The semisupervised method based on the folded spectrum GAN [33] folds the original spectral vector into a 2-D square as the input of the GAN. Similarly, HSGAN [34] extracts spectral features using a custom 1-D GAN and employs a novel CNN framework for classification. A specialized voting strategy is utilized to enhance performance. DAE-GCN [35] introduces a spectral–spatial graph to train a graph convolutional network using a semisupervised strategy. Tang et al. [36] proposed a method for extracting multiscale spatial–spectral features based on a ladder structure. The complexity of hyperspectral data distribution, however, still limits the performance of semisupervised models.

In addition to the issues mentioned above, the use of image patch samples and spatial–spectral features presents challenges related to vague boundaries and misclassification. First, some methods assume that all the pixels within an image patch contribute equally to the classification of the central pixel. However, realistic image patches can be divided into homogeneous and heterogeneous patches: those consisting of the same class of pixels and those containing multiple classes of pixels. Spatial features extracted from homogeneous patches can enhance the classification performance by introducing spatial relationship and suppressing noise. In contrast, spatial features extracted from heterogeneous patches can be viewed as the fusion of pixels from different classes. Consequently, the extracted spatial features from heterogeneous patches may not accurately represent central pixels, limiting their classification performance [37]. The influence of spatial features can be mitigated by properly emphasizing spectral features, which focus on the spectral vector itself. Second, the existing research on spectral–spatial features primarily relies on fixed strategies for fusing the two types of features [21], such as concatenating feature vectors [38], [39], [40]. Considering the characteristics of heterogeneous patches, these fixed strategies may result in reduced performance; especially, patches at boundaries are typically heterogeneous. Adapting the combination of two features could alleviate this issue, such as adding spectral constraints to guide the assignment of attention weight. Therefore, effectively utilizing both types of samples and features remains a critical challenge.

In order to extract deep adaptive spectral–spatial features from various image patches and address sample scarcity and imbalance, we proposed a semisupervised spectral–spatial-dependent learning framework that combines the GAN and the global joint attention mechanism, named dual-branch spectral–spatial adversarial representation learning (SSARL). An adversarial representation module is incorporated to handle limited labeled samples, while the dual-branch structure and class consistency loss offer a novel strategy for adaptively combining spectral and spatial features. The characteristics of pixel and patch samples are also considered. The contributions of this article are summarized as follows.

1) We propose a learnable dual-branch framework that extracts all the useful spectral and spatial features by processing pixel and patch samples independently in parallel.
2) We introduce a loss function called class consistency loss, which replaces the existing feature fusion strategies. This function adds spectral constraints and adjusts the attention weights of spectral and spatial branches adaptively, allowing the learned framework to perform well on heterogeneous patches.
3) We apply an adversarial representation module for spectral and spatial feature extraction. Through the adversarial process, robust features are learned from limited labeled samples.

The rest of this article is organized as follows. Section II presents the details of the proposed SSARL. Section III showcases the results and analysis of our experiments. Finally, Section IV concludes this article.

II. Methodology

In this section, first, we briefly introduce the proposed SSARL. Second, the adversarial representation module is illustrated. Then, the proposed class consistency loss is given. Finally, the details of complete spectral and spatial branches are introduced.

A. Overview of the Proposed Model

An HSI dataset can be represented as $\mathcal{H} \in R^{h\times w\times b}$, where h, w, and b represent the height of spatial size, the width of spatial size, and the number of spectral bands, respectively. The dataset contains N labeled pixel samples. $x_{\text{spe}} \in \mathbb{R}^{1\times 1\times d}$ represents the spectral sample; $y_{\text{spe}} \in \mathbb{R}^{1\times 1\times c}$ represents the corresponding one-hot label, where c denotes the number of classes. x_{spa} represents the patch sample with a size of $m \times m \times d$, where

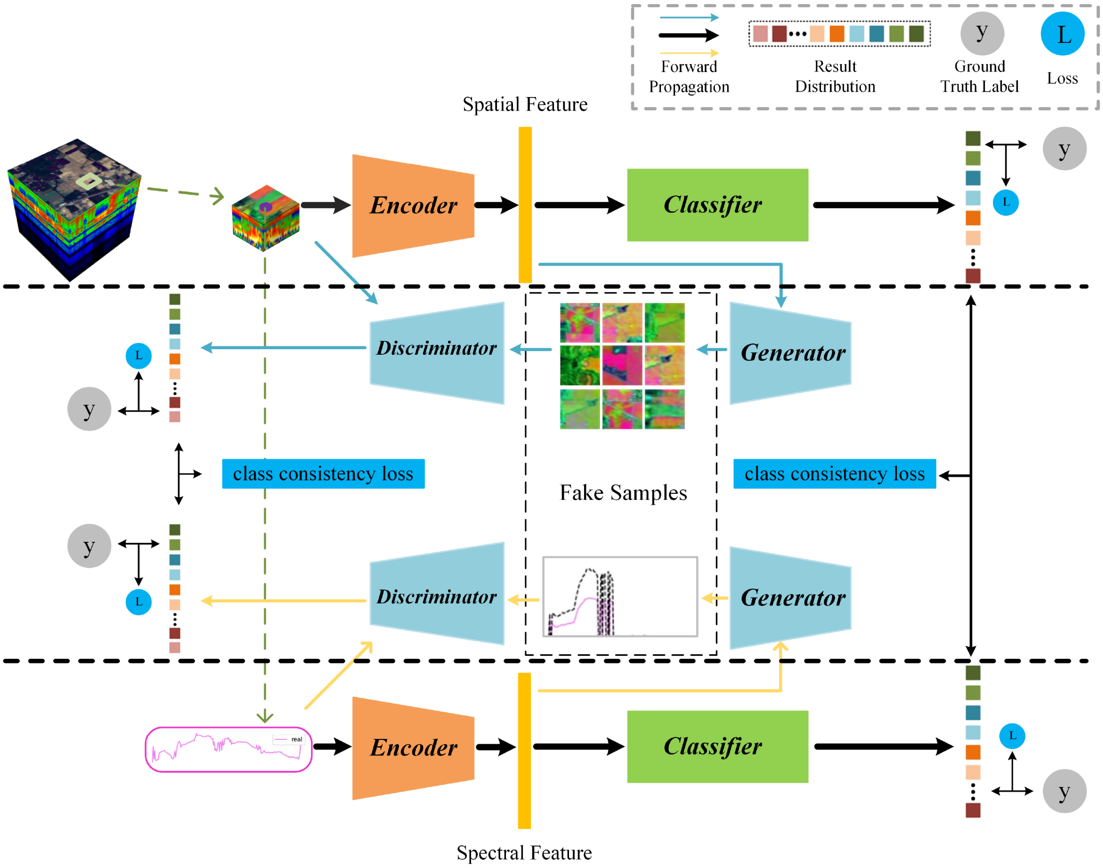


Fig. 1. Structure of SSARL.

m represents the height and width. y_{spa} indicates the one-hot label of corresponding central pixel in the patch. The collection of the labeled pixel samples and labels is represented as L_{spe}, and the unlabeled collection is U_{spe}. Sample pairs $(x_{\text{spe}}, x_{\text{spa}}, y)$ and $(x_{\text{spe}}, x_{\text{spa}})$ are inputs to the model. The SSARL framework is illustrated in Fig. 1. The proposed model contains a spectral branch (the upper half of Fig. 1) and a spatial branch (the lower half of Fig. 1). Each branch consists of an encoder (E) based on the CNN and a classifier (C), including a fully connected layer and Softmax. Instead of learning hierarchical spatial–spectral features or concatenating spectral and spatial features [21], the spectral branch extracts the spectral feature from labeled and unlabeled pixel samples, and the spatial branch extracts the spatial feature from labeled and unlabeled patch samples. To extract robust features from limited labeled samples, we apply adversarial representation modules (the middle part of Fig. 1) based on the GAN to two branches. The proposed class consistency loss unifies the results obtained by two classifiers and discriminators.

The dual-branch structure makes full use of spectral and spatial features from pixel and patch samples. Inspired by the GAN, we insert an adversarial representation module. This module contains a generator (G) and a discriminator (D). Through the adversarial process, the module uses limited labeled samples to enlarge the sample space and increases sample diversity, thus preventing overfitting. Finally, the proposed class consistency loss adds additional constraints to two discriminators and classifiers. The proposed loss function can make use of features adaptively rather than adopting fixed spectral–spatial combination strategies. Meanwhile, dual-branch structure and class consistency loss can reduce the negative impact of spatial features extracted from heterogeneous patches (e.g., a boundary patch is composed of multiclass pixels, and the representation ability of spatial feature is weakened, so more attention should be paid to spectral feature from central pixel). The aforementioned parts will be illustrated in the following sections.

B. Adversarial Representation Module

GANs have been widely used for data augmentation for natural image processing in computer vision [41]. They can maintain an identical distribution as original samples and increase the diversity [42], [43], [44], [45]. The adversarial representation module utilizes the adversarial process to enhance the extracted semantic features. Instead of reconstructing samples at pixel level through the root-mean-square error (e.g., SAE), the adversarial representation module can be seen as a sample construction through variable constraint based on the CNN and the GAN. The proposed adversarial representation module can be applied to pixel-spectral feature and patch-spatial feature extraction. During the adversarial process, multiple mappings from correct features to potential samples are learned. The encoder is also

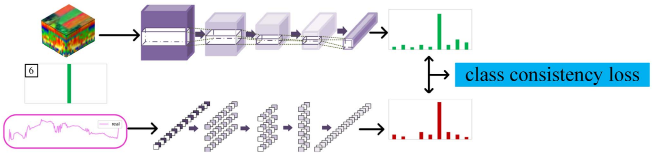


Fig. 2. Class consistency loss. Classification results of two branches are the same, but there are still differences in the possibility of other classes. Class consistency loss calculates the distance between two results.

guided to map sample space to semantic feature space at the class level.

To satisfy Lipschitz continuity, we adopt spectral normalization (SN) [46] for the discriminator. The Lipschitz continuity is defined as the gradient's rate being less than K, formulated as follows:

$$\frac{\|D(x_1) - D(x_2)\|_2}{\|x_1 - x_2\|_2} \leq K \quad \forall x_1, x_2 \tag{1}$$

where D is the function of discriminator, and x_1 and x_2 are two variables close enough. $\|\|_2$ represents the Euclidean norm. SN can be formulated as follows:

$$\|A\|_2 = \sup_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2} = \sigma(A) \tag{2}$$

where sup represents the upper bound. When x is considered within minimum neighborhood, the nonlinear discriminator function D can be regarded as a linear function $D(x) = W_\theta x + b_\theta$, where θ is the parameter of D. Under this condition, Zhang et al. [47] proved that applying SN on multilayer can meet Lipschitz continuity. SN normalizes the parameter matrix by dividing it by the maximum singular value of the parameter matrix on every layer. For a fully connected network layer, SN directly calculates the maximum singular value of the second-order matrix. For a convolution layer, the parameter matrix $[K_H, K_W, C_{\text{in}}, C_{\text{out}}]$ is reshaped into a second-order matrix $[K_H \times K_W \times C_{\text{in}}, C_{\text{out}}]$, and the maximum singular value is calculated by the iterative method.

C. Class Consistency Loss

CycleGAN [48] proposed a consistency loss to guide the mapping between the source domain and the target domain. For the HSI, the input pair samples $(x_{\text{spe}}, x_{\text{spa}})$ belong to the same class, though they have different forms. Therefore, the outputs of two branches should ideally be coincident. However, spectral and spatial features have their own pros and cons, which may lead to different classification results. In order to combine the advantages of both the features in different situations, we proposed a result-driven loss function to assign different attention weights to two features, named class consistency loss, as shown in Fig. 2. The class consistency loss is the distance between two results from two branches. The root-mean-square error is used to measure this distance. The class consistency loss is defined as follows:

$$L_{ss} = E_{x_{\text{spe}}, x_{\text{spa}}}[||F_{\text{spe}}(x_{\text{spe}}) - F_{\text{spa}}(x_{\text{spa}})||^2] \tag{3}$$

where F_{spe} and F_{spa} represent the models of the spectral branch and the spatial branch, respectively. The class consistency loss is added as a constraint when training. The calculated loss value uses stochastic gradient descent to update parameters. If the two prediction results are the same (whether it is right or wrong), the class consistency loss is close to 0 and fine-tunes the network. If the prediction results are different, only one result can be selected randomly as the final result; thus, the final prediction results could be worse. In this case, the consistency loss guides to adjust the parameters. In the classification stage, the role of class consistency loss is to use networks to achieve adaptive voting on the two prediction results obtained from spectral and spatial classifiers. As for the discriminator, which is equivalent to a multiclass classifier, the class consistency loss plays the same role. In the generation stage, the class consistency loss also restricts the samples generated by the two generators to be the same class because the spectral and spatial feature generators used come from the same class. For feature extraction, spectral–spatial features with bias and different contributions of pixels in one patch are learned. We aim to learn the spectral–spatial attention weights and distinguish contributions from different pixels, thereby improving the classification accuracy of boundary samples.

D. Spectral Pixel and Spatial Patch Branches

Based on the adversarial representation module and class consistency loss, the spectral and spatial branches are designed to exploit spectral and spatial information. The input of the spectral branch is $L_{\text{spe}}, U_{\text{spe}} \in \mathbb{R}$, and the input of the spatial branch is L_{spa} and U_{spa}. The encoder maps samples $x_{\text{spe}}(x_{\text{spa}})$ to features $f_{\text{spe}}(f_{\text{spa}})$. Instead of random noise, the features extracted by the spectral (spatial) encoder are used as the input of the spectral (spatial) generator. The fake samples are denoted as $\hat{x_{\text{spe}}} \sim G_{\text{spe}}(f_{\text{spe}})$ and $\hat{x_{\text{spa}}} \sim G_{\text{spa}}(f_{\text{spa}})$. Then, the following three parts are input to the spectral discriminator: $(x_{\text{spe}}, y_{\text{spe}}) \sim L_{\text{spe}}$, $x_{\text{spe}} \sim U_{\text{spe}}$, and $\hat{x_{\text{spe}}} \sim G_{\text{spe}}(f_{\text{spe}})$. Similarly, the following parts are input to the spatial discriminator: $(x_{\text{spa}}, y_{\text{spa}}) \sim L_{\text{spa}}$, $x_{\text{spa}} \sim U_{\text{spa}}$, and $\hat{x_{\text{spa}}} \sim G_{\text{spa}}(f_{\text{spa}})$. After training, the test samples flow through the trained encoders and classifiers. The configuration of the spectral branch is shown in Fig. 3. 1-D convolution and 1-D transposed convolution are widely used. The size of convolutional kernel is k and stride is s. p and Op stand for padding. O represents the number of kernel. SN represents the spectral regularization. The spatial branch modifies the 1-D modules into 2-D modules.

The proposed model employs a semisupervised method. To utilize the unlabeled samples, we add conditional entropy to the objective function. The specific label of a real unlabeled sample is unknown, but it should belong to a certain class; therefore, conditional entropy is added as a prior condition to enhance the performance of the classifier. The equation for conditional entropy is shown as

$$\text{Loss}_{CE} = \lambda E_{x \sim U} \sum_{c=1}^{\overline{C}} P_c(y_c|x) \log p_c(y_c|x) \tag{4}$$

where λ represents the hyperparameter.

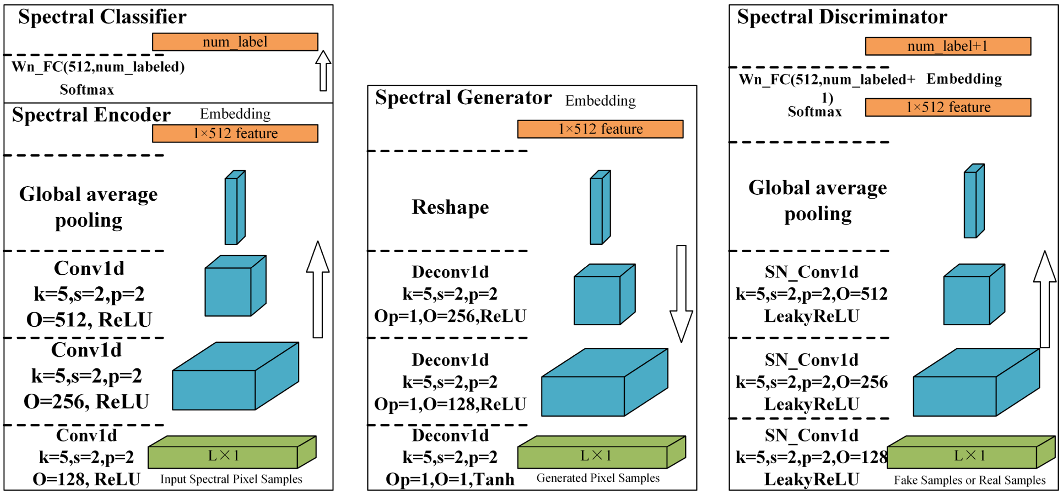

Spectral Classifier
num_label
Wn_FC(512,num_labeled)
Softmax
Spectral Encoder
Embedding
1×512 feature
Global average pooling
Conv1d k=5,s=2,p=2 O=512, ReLU
Conv1d k=5,s=2,p=2 O=256, ReLU
Conv1d k=5,s=2,p=2 O=128, ReLU
L×1
Input Spectral Pixel Samples
Spectral Generator
Embedding
1×512 feature
Reshape
Deconv1d k=5,s=2,p=2 Op=1,O=256,ReLU
Deconv1d k=5,s=2,p=2 Op=1,O=128,ReLU
Deconv1d k=5,s=2,p=2 Op=1,O=1,Tanh
L×1
Generated Pixel Samples
Spectral Discriminator
num_label+1
Wn_FC(512,num_labeled+1)
Softmax
Embedding
1×512 feature
Global average pooling
SN_Conv1d k=5,s=2,p=2,O=512 LeakyReLU
SN_Conv1d k=5,s=2,p=2,O=256 LeakyReLU
SN_Conv1d k=5,s=2,p=2,O=128 LeakyReLU
L×1
Fake Samples or Real Samples

Fig. 3. Configuration of the generator, encoder, discriminator, and classifier in the spectral branch. The arrows indicate the direction of sample processing.

Given real labeled sample pairs, the purpose of the spectral (spatial) discriminator is to classify them correctly. For real unlabeled samples, the purpose of the discriminator is to assign the proper classes. It will classify the real samples into $\overline{C}$ classes. The posterior probability is represented as follows:

$$p(c|x_{\text{spe}};\theta)=\frac{e^{f_c}}{\sum_{c'=1}^{\overline{C}} e^{f_{c'}}}\qquad \forall c\in\{1,\ldots,\overline{C}\} \tag{5}$$

where θ represents the parameters of discriminator, and $f=D_\theta(x_{\text{spe}})$ represents the output feature of the discriminator. Essentially, the discriminator is a modified classifier.

The role of the discriminator also includes discriminating the authenticity of sample. The formula to calculate the samples belongs to the real set is

$$p(\text{real}|x_{\text{spe}};D_\theta)=\frac{\sum_{c'=1}^{\overline{C}} e^{f_c}}{1+\sum_{c'=1}^{\overline{C}} e^{f_{c'}}}. \tag{6}$$

We can assume the generated samples as class $\overline{C}+1$, and it can be represented as 1 in denominator. Then, the samples that belong to fake set can be shown as

$$p(\text{fake}|x_{\text{spe}};D_\theta)=1-p(\text{real}|x_{\text{spe}};D_\theta). \tag{7}$$

The objective function for the discriminator and the classifier in the branch consists of the class cross entropy of labeled samples and the conditional entropy of unlabeled samples. It can be predicted that the performance of the network may be worse when initially updating the network parameters with unlabeled samples. Formula 6 can be translated to the following equation when optimizing:

$$\min E_{x\sim U_{\text{spe}}}[-\log p(\text{real}|x;D_\theta)] \tag{8}$$

where x_{spe} represents the random variable conforming the distribution of U_{spe}, and E represents the expectation. Then, we calculate the negative gradient using the following equation:

$$\begin{aligned}-\frac{\partial}{\partial f_c}[-\log p(\text{real}|x_{\text{spe}};D_\theta)] &= \frac{1}{1+\sum_{c'=1}^{\overline{C}} e^{f_{c'}}}\,\frac{e^{f_c}}{\sum_{c'=1}^{\overline{C}} e^{f_{c'}}}\\ &= p(\text{fake}|x_{\text{spe}};D_\theta)p(c|x_{\text{spe}};\theta).\end{aligned} \tag{9}$$

During the updating of network parameters, the predicted result $p(c|x_{\text{spe}};\theta)$ is strengthened. The neurons related to class $\hat{c}=\arg\max[p(c|x_{\text{spe}};\theta)]$ are stimulated to update parameters. The objective functions for the classifier, generator, and discriminator in the spectral pixel branch are expressed as follows:

$$\begin{aligned}\max L_{C_{\text{spe}}} &= E_{x,y\sim L_{\text{spe}}}\log p_{C_{\text{spe}}}(y_c|x,y)\\ &\quad+\lambda E_{x\sim U_{\text{spe}}}\sum_{c=1}^{\overline{C}} p_{C_{\text{spe}}}(y_c|x)\log p_{C_{\text{spe}}}(y_c|x)\\ &\quad-E_{x\sim\mathbb{R}}[\|C_{\text{spe}}(x_{\text{spe}})-C_{\text{spa}}(x_{\text{spa}})\|^2] \quad (10)\\ \max L_{G_{\text{spe}}} &= \beta E_{x\sim G_{\text{spe}}(f_{\text{spe}})}[-\log(p(\text{real}|x;D_{\text{spe}}))]\\ &\quad+E_{x\sim G(f)}[\|D_{\text{spe}}(x_{\text{spe}})-D_{\text{spa}}(x_{\text{spa}})\|^2] \quad (11)\\ \max L_{D_{\text{spe}}} &= E_{x,y\sim L_{\text{spe}}}\log p_{D_{\text{spe}}}(y_c|x,y)\\ &\quad+\lambda E_{x\sim U_{\text{spe}}}\sum_{c=1}^{\overline{C}} p_{D_{\text{spe}}}(y_c|x)\log p_{D_{\text{spe}}}(y_c|x)\\ &\quad-\beta(E_{x\sim U_{\text{spe}}}[-\log(p(\text{real}|x;D_{\text{spe}}))]\\ &\quad+E_{x\sim G_{\text{spe}}(f_{\text{spe}})}[-\log(p(\text{fake}|x;D_{\text{spe}}))])\\ &\quad-E_{x\sim\mathbb{R}}[\|D_{\text{spe}}(x_{\text{spe}})-D_{\text{spa}}(x_{\text{spa}})\|^2]. \quad (12)\end{aligned}$$

In formula (10), the first two terms represent the cross entropy for classifying labeled samples and the conditional entropy for unlabeled samples. In formula (11), the objective function of the generator is responsible for making the discriminator misjudgment. For the objective function of the discriminator in formula (12), the first term aims to classify the labeled

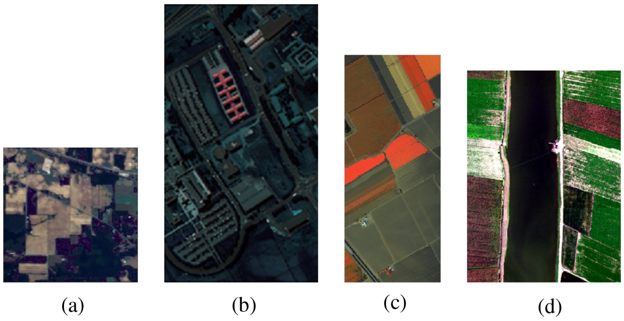


Fig. 4. (a) IP, (b) PU, (c) SA, and (d) WHU datasets.

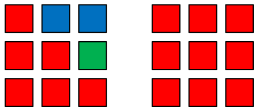


Fig. 5. Illustration of boundary samples and patches, boundary samples (left) and within-class samples (right).

samples correctly. The second term assigns the dominant class of unlabeled samples using conditional entropy. The third and fourth terms judge whether the samples are real or fake. The final terms of the three formulas represent the class consistency loss between spatial and spectral branches. Here, β represents the adversarial weight.

If the basic classification performance of the classifier is good and predicts correctly, the training process will develop in the right direction. However, the number of labeled training samples is limited, and the model may not be sufficiently learned. Therefore, it is possible that the initial performance of the classifier is poor, leading to many incorrect predictions, and the error is magnified through the update process of Formula (9) and conditional entropy. To address this issue, initial λ is set to a small value and gradually increases as the training progresses.

The spatial and spectral branches propagate forward at the same time and backpropagate after calculating their respective loss values with the objective function. Within each branch, E, G, D, and C are optimized alternately. Considering E and G as a whole for updating parameters, we optimize the network parameters of D when the network parameters of E, G, and C are fixed. Conversely, when D is fixed, the rest parameters are updated. We use the Adam optimizer. Through this end-to-end network structure and alternating optimization, classifier C is finally used to classify test samples.

III. Experimental Results and Analysis

In this section, the datasets, configuration, and hyperparameters are introduced first. Second, we analyze the influence of the proposed dual-branch structure, adversarial learning, and class consistency loss and analyze the sensitivity of the model to the number of labeled samples. Third, we compare the performance of the proposed model with that of competitive methods. Finally, we discuss the comparison results.

A. Dataset Description

Four public datasets are employed in the experiments: Indian Pines (IP), Pavia University (PU), Salinas (SA), and WHU-Hi-LongKou (WHU) [49]. Fig. 4 shows the false-color images.

1) *Indian Pines:* The IP dataset was gathered by the AVIRIS sensor in the northeast of Indiana. The IP scene consists of two-thirds agriculture and one-third forest or other natural perennial vegetation. Some of the crops present, such as corn and soybeans, were in early stages of growth with less than 5% coverage. The scene is 145 × 145 in size and has a spatial resolution of 20 m/pixel. Twenty spectral bands are removed due to water absorption, leaving 200 bands with a spectral range of 400–2500 nm. The IP dataset contains 16 classes with 10 249 labeled pixels and background information.
2) *Pavia University:* The PU dataset was gathered in the University of Pavia in northern Italy by the Reflective Optics System Imaging Spectrometer in 2002. The number of spectral bands is 115, including 12 noisy bands. It has a spatial size of 610 × 340 and a resolution of 1.3 m/pixel. After removing 12 noisy bands, 103 bands range from 430 to 860 nm. PU consists of nine classes with 42 776 labeled pixels for classification.
3) *Salinas:* The SA dataset was gathered by the AVIRIS sensor over the Salinas Valley, CA, USA, and was characterized by high spatial resolution (3.7 m/pixel). This image is available only in the form of at-sensor radiance data. The ground area of SA includes vegetables, bare soils, and vineyard fields. SA consists of 224 spectral bands; 20 water absorption bands are removed. The dataset has a size of 512 × 217 × 224 and a spatial resolution of 3.7 m/pixel with 16 land-cover classes.
4) *WHU-Hi-LongKou:* The WHU dataset was acquired from 13:49 to 14:37 on July 17, 2018, in Longkou Town, Hubei province, China, with an 8-mm focal length Headwall Nano-Hyperspec imaging sensor equipped on a DJI Matrice 600 Pro (DJI M600 Pro) UAV platform. The study area contains nine classes, including six crop species and three ground objects. The imagery has a size of 550 × 400 pixels. The spectral dimension comprises 270 bands ranging from 400 to 1000 nm, and the spatial resolution of the UAV-borne hyperspectral imagery is approximately 0.463 m.

Table I shows the number of each class of four datasets. For IP, we randomly select 5% of the labeled samples as labeled training set. For PU, SA, and WHU, the proportions of selected labeled samples are 3%, 1%, and 0.3% of whole samples. The random selection follows the class balance. All input data are normalized between -1 and 1 in advance.

Particularly, we define boundary samples as pixels that differ from any of the eight surrounding pixels, as shown Fig. 5.

B. Experimental Setting

The whole experiments are conducted on a computer equipped with an NVIDIA GeForce GTX 1080Ti with 12-GB RAM. The software environment is Ubuntu 14.04 ultimate

TABLE I
LAND-COVER CLASS INFORMATION AND THE NUMBER OF ANNOTATED SAMPLES OF IP, PU, SA, AND WHU

	IP		PU		SA		WHU	
No.	Class	Number	Class	Number	Class	number	Class	number
1	Alfalfa	46	Asphalt	6631	Brocoli-green-weeds-1	1977	Corn	34511
2	Corn-notill	1428	Meadows	18649	Brocoli-green-weeds-2	3726	Cotton	8374
3	Corn-mintill	830	Gravel	2099	Fallow	1976	Seasame	3031
4	Corn	237	Trees	3064	Fallow-rough-pow	1394	Broad-leaf soybean	63212
5	Grass-pasture	483	Paint metal sheets	1345	Fallow-smooth	2678	Narrow-leaf soybean	4151
6	Grass-trees	730	Bare Soil	5029	Stubble	3959	Rice	11854
7	Grass-pasture-mowed	28	Bitumen	1330	Celery	3579	Water	67056
8	Hay-windrowed	478	Self-Blocking Bricks	3682	Grapes-untrained	11213	Roads and houses	7124
9	Oats	20	Shadows	947	Soil-vinyard-develop	6197	Mixed weed	5229
10	Soybean-notill	972	–	–	Cone-senesced-G-weeds	3249	-	-
11	Soybean-mintill	2455	–	–	Lettuce-romaine-4wk	1058	-	-
12	Soybean-clean	593	–	–	Lettuce-romaine-5wk	1908	-	-
13	Wheat	205	–	–	Lettuce-romaine-6wk	909	-	-
14	Woods	1265	–	–	Lettuce-romaine-7wk	1061	-	-
15	Buildings-Grass-Trees	386	–	–	Vinyard-untrained	7164	-	-
16	Stone-Steel-Towers	93	–	–	Vinyard-vertical-trellis	1737	-	-
Total		10249		42776		53785		204542

TABLE II
FLOATING POINT OPERATIONS AND PARAMETERS OF SSARL

	Spatial D	Spatial EGC	Spectral D	Spectral EGC
FLOPs (M)	16.48	107.04	12.82	12.89
Params (M)	1.00	4.13	0.82	0.82

64 bit. The deep learning frameworks used are TensorFlow and Pytorch.

The samples are processed by a Gaussian smoothing kernel before being input to the model. In the spatial branch, the spectral dimension of HSI patches is reduced to 10 by PCA, and the size of patches is set to 8 × 8 for IP, PU, and SA, and 9 × 9 for WHU.

During training, we use the batch size of 32. An annealing algorithm is considered for setting the learning rate, with a range of [0.0, 0.002]. The conditional entropy weight λ determines the effect of unlabeled samples. We consider the λ values in the range [0.5,1]. For every 100 training steps, λ increases by 0.05. The adversarial weight β with the value range of [0.5,1] increases by 0.05 every 100 training steps. The number of training steps is 1000. The network parameters are presented in Section II. The above parameters are adjusted using a standard random grid search cross-validation framework. F1-score, overall accuracy (OA), average accuracy (AA), and kappa coefficient (Kappa) are used to quantitatively evaluate models. The results are obtained after ten independent runs, with the training and test sets randomly divided each time. The FLOPs and parameters of SSARL (Input 8 × 8 × 10) are shown in Table II. D represents the discriminator and EGC represents the thread process.

C. Ablation Study

To demonstrate the effectiveness of the semisupervised strategy, spectral adversarial learning branch, and class consistency loss, we compare CNN, CNN-CE, CNN-CE-SS, and proposed method. CNN represents the method with the same structure and configuration as the encoder and the classifier in the proposed spatial branch but uses a standard cross-entropy loss function. Therefore, CNN does not utilize unlabeled samples. CNN-CE represents a semisupervised model that adds conditional entropy for training. CNN-CE-SS represents a model that adds spectral and spatial branches based on CNN-CE without class consistency loss, and the structure and configuration are the same as those of the proposed method.

The results of OA, AA, and Kappa are shown in Table III. The overall classification results presented in the table (from left to right) increase with the increase of innovations. Compared with CNN, the OA has improved by 0.5%, 1.0%, 0.8%, and 0.6% after utilizing unlabeled samples, which proves that the information is mined from a large number of unlabeled samples. Compared with CNN-CE, CNN-CE-SS shows a significant improvement of OA in IP and WHU, a slight improvement in PU, but a decline in SA due to the introduction of spectral features. It can be inferred that adversarial representation learning modules mitigated the sample imbalance by generating fake samples in IP. The decline demonstrates the possible shortcomings of spatial features without applying combination methods. Compared with CNN-CE-SS, the proposed method has improved OA by 0.34%, 0.01%, 0.21%, and 0.50%. The OA of SA and PU does not vary significantly, but proposed method performs better on AA and Kappa. It proves that the class consistency loss inhibits the influence of heterogeneous patches and extracts valuable joint features by adding spectral constraint adaptively. On the premise that SSARL achieves better performance, Table III presents two outliers. First, CNN has achieved better performance on AA in IP. The second issue is that CNN-CE-SS performed worse than CNN-CE in SA. IP and SA are obtained from the same series of hyperspectral sensors. It can be inferred that caution is needed in the use of spatial–spectral features in SA and the imbalanced unlabeled samples in IP.

D. Sensitive Analysis of the Number of Labeled Samples

The number of labeled training samples greatly affects the classification performance of deep learning methods. Therefore, we analyze the performance of the proposed method and other methods using different numbers of labeled samples. Figs. 6–8 show the OA of seven methods RBF-SVM, SAE, DBN, PPF-CNN [22], 3-D CNN, MSGAN, and proposed method using different sizes of labeled training samples in IP, PU, and SA, respectively. Figures show that the accuracy of all the methods

TABLE III
EFFECTIVE ANALYSIS OF ADVERSARIAL LEARNING AND CLASS CONSISTENCY LOSS

Dataset	Evaluating indices	CNN	CNN-CE	CNN-CE-SS	SSARL
IP (Train: 5%)	OA	97.0 ± 0.4	97.5 ± 0.4	98.3 ± 0.5	$\mathbf{98.6 \pm 0.4}$
	AA	$\mathbf{96.2 \pm 0.9}$	93.6 ± 0.5	93.1 ± 3.9	95.7 ± 1.8
	Kappa	96.6 ± 0.5	97.1 ± 0.5	98.0 ± 0.5	$\mathbf{98.4 \pm 0.4}$
PU (Train: 3%)	OA	98.9 ± 0.1	99.9 ± 0.1	99.9 ± 0.0	$\mathbf{99.9 \pm 0.0}$
	AA	98.2 ± 0.2	99.6 ± 0.1	99.7 ± 0.1	$\mathbf{99.7 \pm 0.1}$
	Kappa	98.5 ± 0.1	99.7 ± 0.1	99.8 ± 0.0	$\mathbf{99.8 \pm 0.0}$
SA (Train: 1%)	OA	99.1 ± 0.1	99.9 ± 0.0	99.8 ± 0.1	$\mathbf{99.9 \pm 0.0}$
	AA	98.3 ± 0.2	$\mathbf{100.0 \pm 0.0}$	99.8 ± 0.1	$\mathbf{100.0 \pm 0.0}$
	Kappa	99.0 ± 0.1	100.0 ± 0.0	99.8 ± 0.1	$\mathbf{100.0 \pm 0.0}$
WHU(Train: 0.3%)	OA	94.6 ± 0.7	95.2 ± 0.3	98.0 ± 0.0	$\mathbf{98.5 \pm 0.1}$
	AA	84.2 ± 0.9	89.8 ± 0.6	96.9 ± 0.2	$\mathbf{97.1 \pm 0.1}$
	Kappa	92.8 ± 0.8	94.2 ± 0.5	97.4 ± 0.3	$\mathbf{98.3 \pm 0.0}$

The bold values represent the best results.

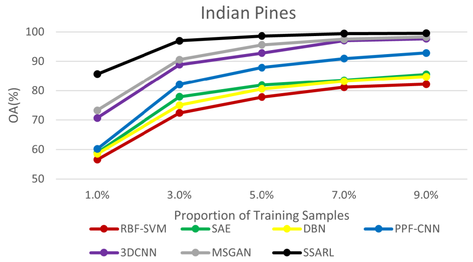


Fig. 6. OA of the seven methods on IP under different sizes of labeled training samples.

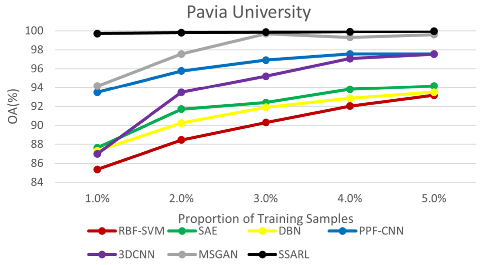


Fig. 7. OA of the seven methods on PU under different sizes of labeled training samples.

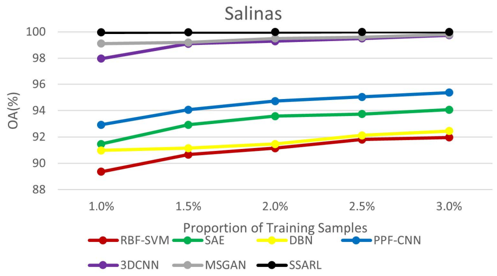


Fig. 8. OA of the seven methods on SA under different sizes of labeled training samples.

on three datasets decreases with the decrease in the number of labeled samples. First, the curve of SSARL is above other curves, which proves that our method has the highest accuracy. Second, the curve of SSARL is stablest, which demonstrates that it is the least sensitive to the sizes of labeled samples and can perform better. In SA and PU, the proposed method is almost unaffected by the sizes of labeled samples. Therefore, our method is a better choice when the number of labeled samples for training is limited. It is worth noting that MSGAN performs well, which may be attributed to the ability to expand the sample space effectively with a small number of labeled samples from GAN-based methods.

E. Comparison of Classification Results

In this section, we directly compare the classification performance between SSARL and other methods, including traditional method RBF-SVM [10], and seven deep learning methods 1-D CNN [15], RDACN [50], 3-D CNN [51], HybridSN [26], SSFTT [30] introducing self-attention, semisupervised RSEN [52], and dual-branch model DBR [53]. We also compare the proposed method with four GAN-based methods: MSGAN [25], 3-D GAN [24], HSGAN [34], and ARL-GAN [47].

1) Indian Pines: The classification results of the IP dataset are shown in Table IV. This table records the average classification accuracy and standard deviation in ten independent runs. The last 16 rows record the classification F1-score of the corresponding class. Compared with RBF-SVM, 1-D CNN have extracted deep spectral feature from pixel samples. Considering the spatial feature and image patch samples, HybridSN and 3-D CNN utilize the spectral–spatial feature. The OA of 3-D CNN shows a 29% improvement compared to 1-D CNN. Furthermore, compared with 3-D CNN, which uses 3-D convolutional layers to extract spatial–spectral features from patch samples, RSEN and DBR utilize pixel, patch samples, and unlabeled samples to obtain information, resulting in a 3.4% improvement in the OA of RSEN compared to HybridSN. SSFTT extracts global features, which results in an improvement of 5.7% OA compared to RSEN. Classifiers usually perform worse in IP when the number of labeled samples is limited. Through the adversarial process, the encoder extracts robust spatial and spectral features. The dual-branch structure and class consistency loss ensure the performance on heterogeneous samples. Compared with SSFTT, SSARL has improved OA, AA, and Kappa by 1.2%, 0.2%, and 1.5%, respectively. The proposed method also achieves the optimal classification results in 14 classes, especially in classes 1, 9, 13, and 16, which have a small sample size.

Fig. 9 shows the classification maps of different methods in the IP dataset. First, maps based on deep learning using

TABLE IV
QUANTITATIVE CLASSIFICATION RESULTS OF DIFFERENT METHODS IN THE IP DATASET

	RBF-SVM	HybridSN	RSEN	3DCNN	1DCNN	RDACN	DBR	SSFTT	SSARL
OA (%)	77.8 ± 0.8	88.3 ± 0.5	91.7 ± 1.7	92.8 ± 0.8	63.9 ± 4.8	97.0 ± 0.1	94.5 ± 0.8	97.4 ± 0.7	**98.6 ± 0.4**
AA (%)	61.3 ± 1.4	78.1 ± 0.1	79.4 ± 4.2	89.4 ± 1.4	54.3 ± 3.6	**96.7 ± 0.3**	91.1 ± 0.7	95.5 ± 1.1	95.7 ± 1.8
Kappa	74.5 ± 1.0	86.7 ± 0.8	90.4 ± 5.6	91.9 ± 0.9	58.8 ± 8.9	96.6 ± 0.4	93.7 ± 0.4	96.9 ± 0.9	**98.4 ± 0.4**
1	6.1 ± 11.2	60.0 ± 0.0	62.5 ± 5.0	50.4 ± 8.4	22.8 ± 7.8	**100.0 ± 0.0**	50.0 ± 4.8	90.2 ± 2.4	96.5 ± 5.1
2	72.9 ± 3.6	87.1 ± 0.6	89.3 ± 0.3	92.7 ± 3.5	53.8 ± 3.5	95.2 ± 0.8	93.3 ± 0.6	96.2 ± 0.7	**98.9 ± 0.4**
3	58.0 ± 3.6	80.4 ± 0.0	91.2 ± 2.8	87.2 ± 10.4	44.8 ± 7.8	95.8 ± 0.0	95.6 ± 0.9	95.7 ± 1.4	**96.9 ± 2.9**
4	39.0 ± 15.0	75.6 ± 5.9	73.0 ± 8.3	83.4 ± 8.3	23.7 ± 2.5	99.1 ± 0.1	76.5 ± 1.2	96.5 ± 0.5	**99.8 ± 0.3**
5	87.0 ± 4.5	87.7 ± 0.7	91.5 ± 4.5	84.0 ± 5.7	74.5 ± 4.5	95.7 ± 0.0	92.2 ± 0.6	**98.4 ± 0.8**	97.3 ± 1.3
6	92.4 ± 2.0	92.6 ± 2.1	99.4 ± 0.6	93.4 ± 2.5	87.1 ± 6.9	96.9 ± 0.3	99.0 ± 0.5	98.3 ± 1.7	**99.7 ± 0.2**
7	0.0 ± 0.0	65.0 ± 0.0	41.2 ± 27.8	97.2 ± 4.8	21.1 ± 2.5	81.3 ± 0.5	89.9 ± 0.5	**100.0 ± 0.0**	**100.0 ± 0.0**
8	98.1 ± 1.4	94.5 ± 1.4	97.4 ± 0.3	97.4 ± 2.8	85.6 ± 4.5	**100.0 ± 0.0**	96.5 ± 0.3	99.3 ± 0.4	**100.0 ± 0.0**
9	0.0 ± 0.0	53.7 ± 2.6	64.3 ± 1.9	77.0 ± 11.1	33.3 ± 1.9	88.9 ± 0.6	**100.0 ± 0.0**	91.0 ± 0.3	87.0 ± 23.6
10	65.8 ± 3.7	86.0 ± 0.2	87.4 ± 2.4	93.3 ± 5.0	56.3 ± 9.2	95.9 ± 0.1	91.2 ± 2.2	95.7 ± 0.4	**96.4 ± 3.1**
11	85.3 ± 2.9	92.6 ± 5.5	95.4 ± 2.3	94.9 ± 2.7	67.8 ± 2.3	98.2 ± 0.4	96.0 ± 0.0	98.2 ± 1.5	**99.6 ± 0.3**
12	69.6 ± 6.5	80.3 ± 2.8	74.4 ± 4.2	89.8 ± 4.3	39.8 ± 5.6	94.3 ± 0.5	89.2 ± 0.4	92.9 ± 6.4	**97.7 ± 1.8**
13	92.3 ± 4.1	87.3 ± 7.4	99.2 ± 0.3	92.8 ± 5.9	69.6 ± 1.9	96.2 ± 0.0	**100.0 ± 0.0**	99.7 ± 0.0	**100.0 ± 0.0**
14	96.6 ± 1.0	96.3 ± 0.1	95.4 ± 1.2	98.3 ± 1.3	86.4 ± 1.2	99.3 ± 0.1	98.3 ± 0.1	99.6 ± 0.2	**100.0 ± 0.0**
15	41.7 ± 7.0	74.8 ± 7.5	90.6 ± 3.7	77.8 ± 13.4	32.7 ± 3.6	96.2 ± 0.7	95.6 ± 0.0	99.0 ± 0.5	**99.9 ± 0.1**
16	75.2 ± 9.0	71.3 ± 2.5	88.9 ± 2.9	88.4 ± 5.3	82.7 ± 8.9	**96.7 ± 0.0**	95.5 ± 0.9	94.8 ± 2.2	95.3 ± 4.9

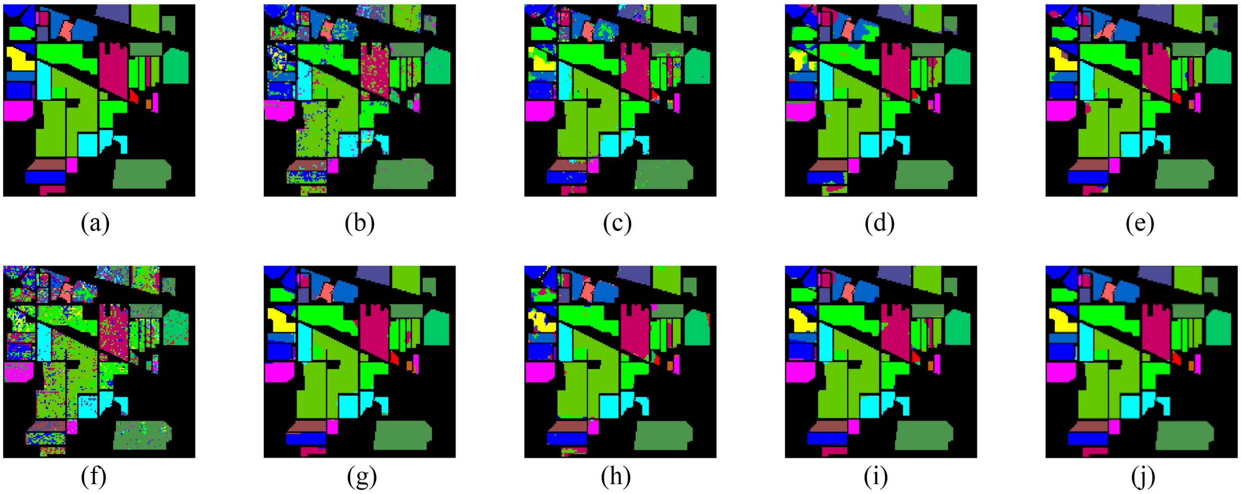

(a) (b) (c) (d) (e)
(f) (g) (h) (i) (j)

Fig. 9. Ground truth and classification maps in IP. (a) Ground truth. (b) RBF-SVM. (c) HybridSN. (d) RSEN. (e) 3-D CNN. (f) 1-D CNN. (g) RDACN. (h) DBR. (i) SSFTT. (j) SSARL.

spectral–spatial features have fewer dot noises. However, due to complex spatial information at the boundary, a large number of misclassification points are presented. The map produced by SSARL is closest to the ground truth map. It demonstrates that proposed method can combine spectral and spatial features adaptively, thereby classifying test samples more accurately.

2) Pavia University: The classification results of the PU dataset are shown in Table V. The distribution of samples in PU is more scattered than IP, making it easier to classify. First, SSARL performs better than the other seven models on OA, AA, and Kappa. The performance of RBF-SVM, 1-D CNN, 3-D CNN, HybridSN, RSEN, and DBR improves progressively due to spectral–spatial features and unlabeled samples. The 1.9% OA improvement proves that using unlabeled samples for semisupervised training can effectively improve the classification performance. Compared with 3-D CNN, the proposed method has improved OA, AA, and Kappa by 4.7%, 8.5%, and 6.0%, respectively. Compared with SSFTT, SSARL has improved OA, AA, and Kappa by 1.1%, 1.5%, and 0.3%, respectively. The proposed method has eight classes (a total of nine classes) achieving the best classification results, with six classes achieving entirely correct classification results. Performance in the eighth and fifth classes has improved.

Fig. 10 shows the classification maps of different methods in the PU dataset. It is consistent with the conclusions in the IP dataset. SSARL has achieved better regional consistency and boundary performance.

3) Salinas: The classification results of the SA dataset are presented in Table VI. SA is a relatively easier dataset to classify than IP. Therefore, all the methods achieved higher classification results than those in the IP dataset. First, under the three evaluation criteria OA, AA, and Kappa, the classification performance of SSARL is better than that of the other seven methods. However, we found that RSEN did not perform well, which may be due to a large number of unlabeled samples negatively impacting classification. Therefore, our method gradually increases the loss weight of unlabeled samples during training. Compared with the

TABLE V
QUANTITATIVE CLASSIFICATION RESULTS OF DIFFERENT METHODS IN THE PU DATASET

	RBF-SVM	HybridSN	RSEN	3DCNN	1DCNN	RDACN	DBR	SSFTT	SSARL
OA (%)	90.3 ± 0.6	98.3 ± 0.9	98.8 ± 0.3	95.2 ± 0.7	91.3 ± 0.4	97.5 ± 0.3	99.5 ± 0.0	98.8 ± 0.9	$\mathbf{99.9 \pm 0.0}$
AA (%)	86.6 ± 0.9	95.9 ± 0.6	97.8 ± 0.4	91.2 ± 1.1	88.1 ± 0.7	96.8 ± 0.9	99.4 ± 0.0	98.2 ± 0.6	$\mathbf{99.7 \pm 0.1}$
Kappa	87.1 ± 0.8	97.7 ± 0.4	98.4 ± 0.9	93.8 ± 0.9	88.4 ± 0.9	97.0 ± 0.2	99.4 ± 0.1	98.5 ± 0.3	$\mathbf{99.8 \pm 0.0}$
1	90.7 ± 1.1	98.0 ± 0.2	98.7 ± 0.7	95.5 ± 1.2	92.3 ± 0.6	98.2 ± 0.5	99.6 ± 0.1	99.1 ± 0.2	$\mathbf{100.0 \pm 0.0}$
2	96.8 ± 0.7	99.7 ± 0.1	99.6 ± 0.4	99.4 ± 0.3	95.9 ± 0.2	99.4 ± 0.6	99.8 ± 0.0	99.7 ± 0.0	$\mathbf{100.0 \pm 0.0}$
3	60.2 ± 5.4	95.1 ± 6.0	95.4 ± 0.9	92.6 ± 5.4	72.1 ± 0.7	94.4 ± 0.6	$\mathbf{98.7 \pm 0.0}$	96.9 ± 0.7	97.4 ± 0.1
4	90.8 ± 2.0	96.9 ± 1.7	99.6 ± 0.0	75.2 ± 4.9	91.8 ± 0.7	97.6 ± 0.1	99.1 ± 0.0	96.6 ± 0.6	$\mathbf{100.0 \pm 0.0}$
5	98.8 ± 0.4	97.7 ± 0.3	99.8 ± 0.2	95.4 ± 4.3	99.5 ± 0.1	98.6 ± 0.6	99.8 ± 0.1	99.6 ± 0.2	$\mathbf{100.0 \pm 0.0}$
6	79.5 ± 4.9	99.7 ± 1.8	98.5 ± 0.7	99.4 ± 0.6	87.3 ± 0.8	89.8 ± 0.9	99.4 ± 0.0	99.4 ± 0.3	$\mathbf{100.0 \pm 0.0}$
7	74.3 ± 5.1	97.6 ± 5.7	94.6 ± 0.2	91.5 ± 3.4	82.9 ± 1.2	97.6 ± 0.1	99.6 ± 0.0	$\mathbf{99.9 \pm 0.0}$	99.8 ± 0.1
8	88.8 ± 2.2	95.7 ± 0.8	97.5 ± 0.7	94.8 ± 1.4	80.0 ± 0.5	97.3 ± 0.4	99.0 ± 0.4	96.3 ± 0.9	$\mathbf{100.0 \pm 0.0}$
9	99.8 ± 0.1	87.9 ± 0.1	99.5 ± 0.3	77.4 ± 2.8	96.5 ± 1.8	96.3 ± 0.5	99.5 ± 0.1	96.2 ± 0.5	$\mathbf{99.9 \pm 0.1}$

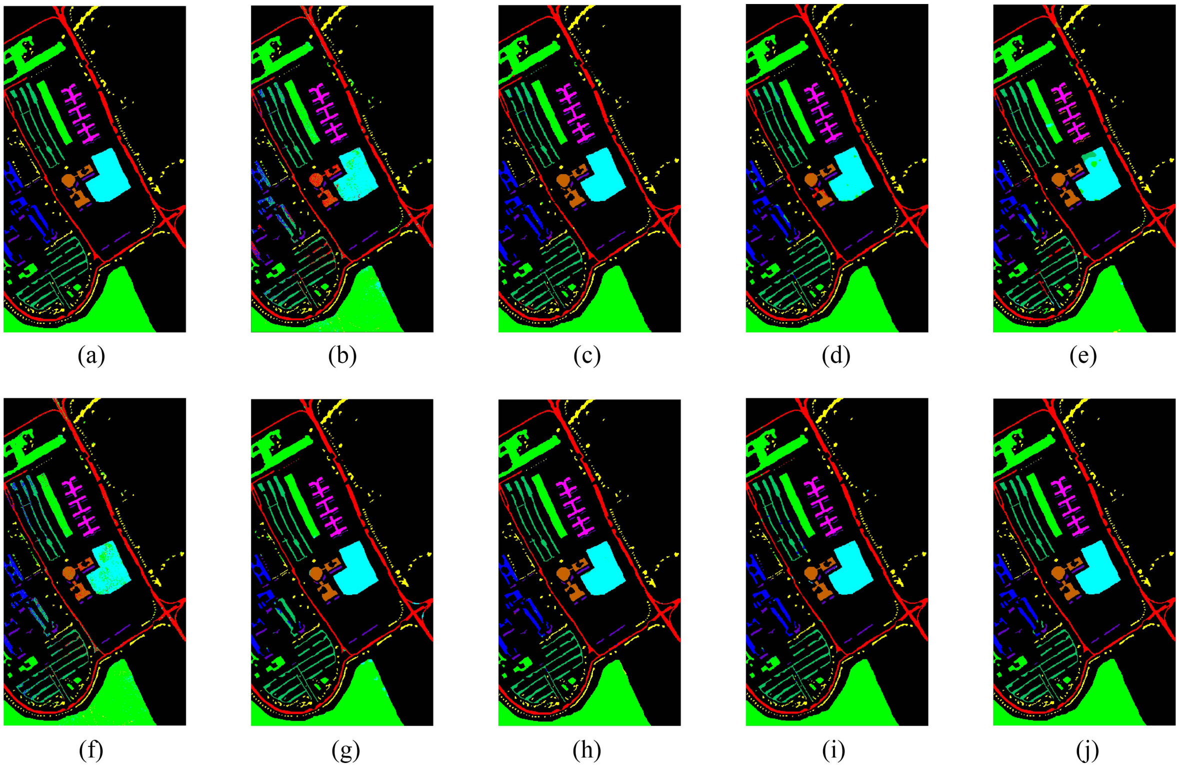

(a) (b) (c) (d) (e)
(f) (g) (h) (i) (j)

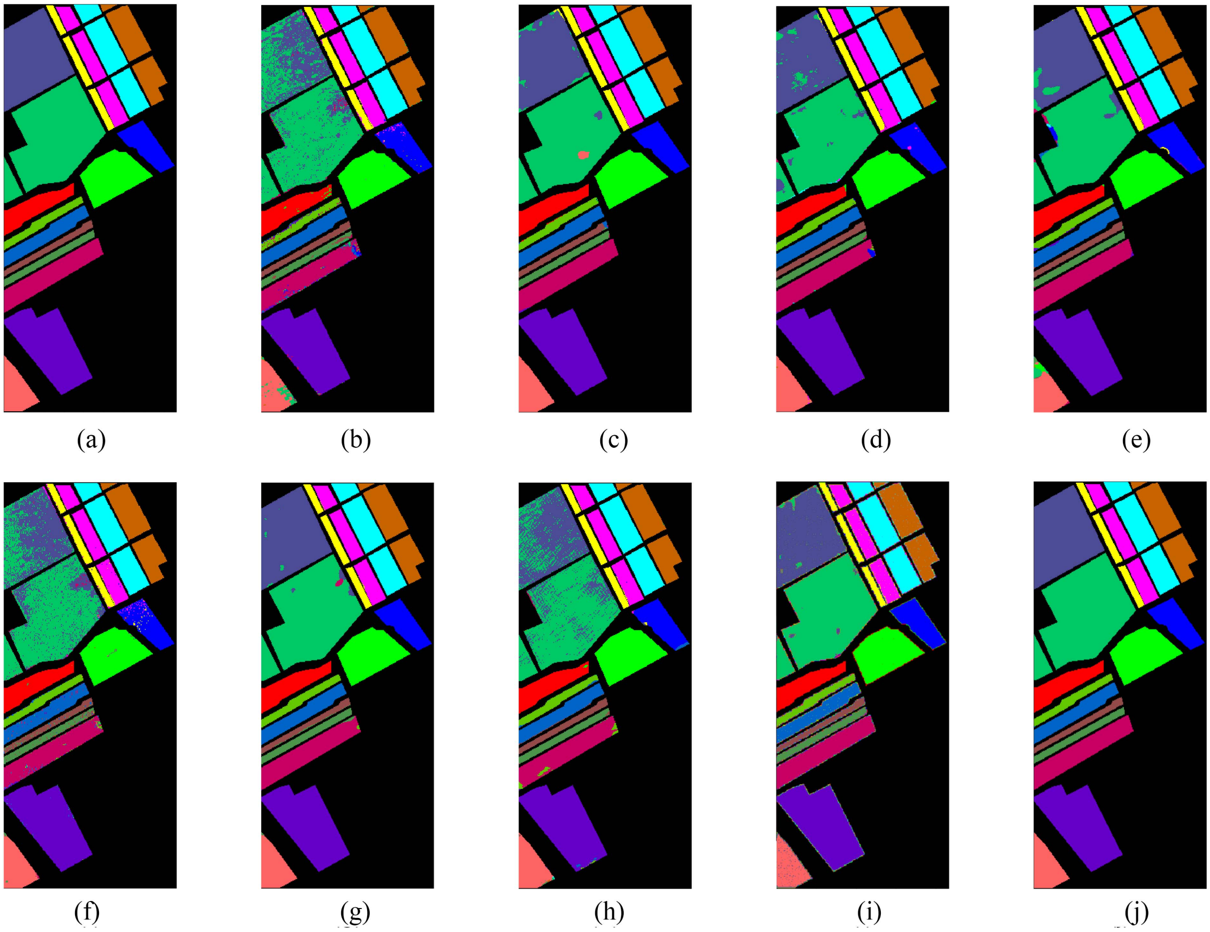
Fig. 10. Ground truth and classification maps in PU. (a) Ground truth. (b) RBF-SVM. (c) HybridSN. (d) RSEN. (e) 3-D CNN. (f) 1-D CNN. (g) RDACN. (h) DBR. (i) SSFTT. (j) SSARL.

competitor method SSFTT, SSARL has improved OA, AA, and Kappa by 0.8%, 0.78%, and 0.99%, respectively. SSARL has achieved the best classification accuracy on all the classes. It has achieved 100% accuracy in 13 of them. For class 16, the performance has been significantly improved.

Fig. 11 shows the classification maps of different methods on the SA dataset. SSARL can distinguish samples from the 8th and 15th classes more effectively. It shows that SSARL can better classify the boundary samples and reduce the classification error points within the class.

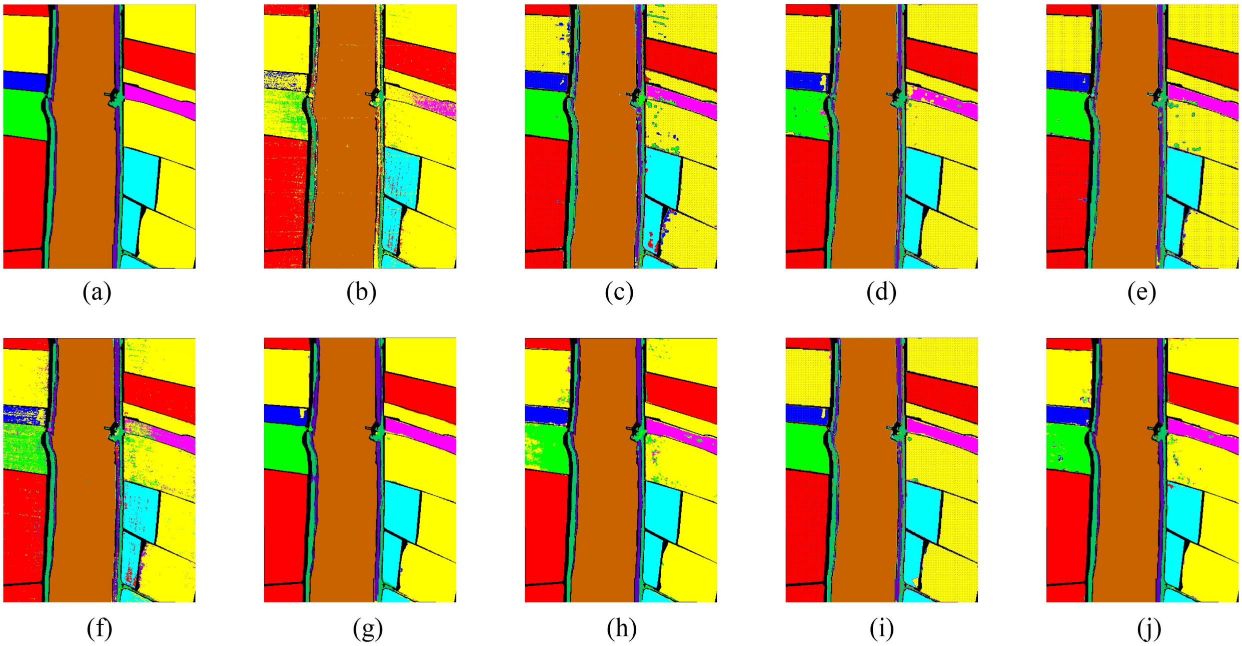
Furthermore, we compared the performance of several latest models based on GAN. The comparison results are presented in Table VII. HSGAN uses spectral samples, while 3-D GAN and ARL-GAN use a spatial image patch trained model for classification. MSGAN is a spectral–spatial method. Compared with HSGAN, SSARL has increased OA by 24.48%, 14.69%, and 11.70% on three datasets. Compared with 3-D GAN, SSARL increases OA by 3.13%, 1.64%, and 1.11% on three datasets. Compared with MSGAN, SSARL has increased OA by 2.65%, 0.69%, and 0.88% on three datasets. These results demonstrate that adversarial representation model, class consistency loss, and the dual-branch structure contribute to better classification accuracy.

4) WHU-Hi-LongKou: The classification results of the WHU dataset are presented in Table VIII. The size and data amount of WHU is larger than those of the above datasets. As shown in the table, SSARL outperforms other competent methods in terms of OA, AA, and Kappa. DBR, which uses a


(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Fig. 11. Ground truth and classification maps in PU. (a) Ground truth. (b) RBF-SVM. (c) HybridSN. (d) RSEN. (e) 3-D CNN. (f) 1-D CNN. (g) RDACN. (h) DBR. (i) SSFTT. (j) SSARL.

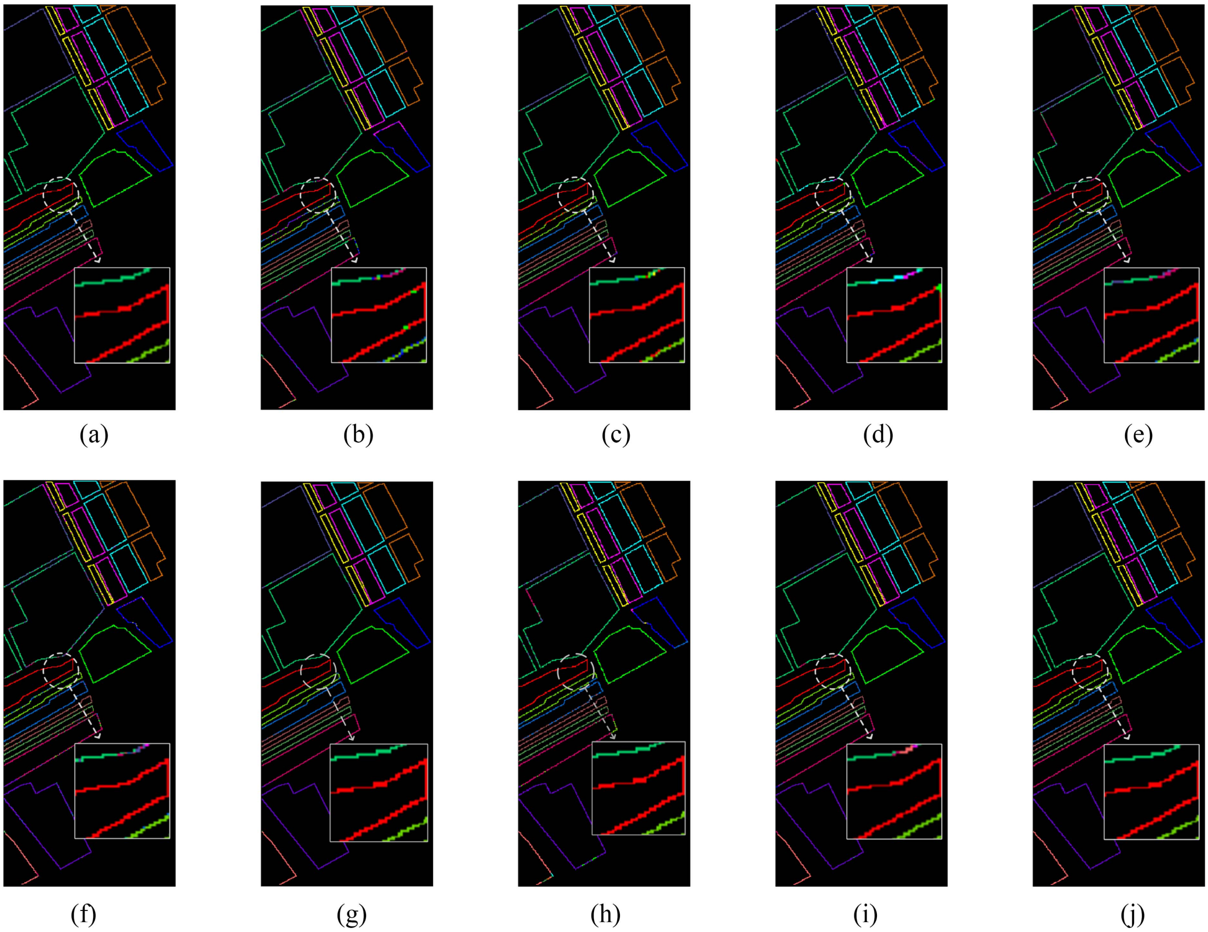

(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Fig. 12. Ground truth and classification maps in WHU. (a) Ground truth. (b) RBF-SVM. (c) HybridSN. (d) RSEN. (e) 3-D CNN. (f) 1-D CNN. (g) RDACN. (h) DBR. (i) SSFTT. (j) SSARL.

TABLE VI
QUANTITATIVE CLASSIFICATION RESULTS OF DIFFERENT METHODS IN THE SA DATASET

	RBF-SVM	HybridSN	RSEN	3DCNN	1DCNN	RDACN	DBR	SSFTT	SSARL
OA (%)	89.3 ± 0.7	98.6 ± 0.9	97.7 ± 0.6	97.9 ± 0.4	89.8 ± 0.6	94.8 ± 0.6	93.3 ± 0.8	99.1 ± 0.1	**99.9 ± 0.0**
AA (%)	92.5 ± 0.6	98.2 ± 0.2	98.5 ± 0.2	98.0 ± 0.6	93.3 ± 0.2	96.4 ± 0.2	96.1 ± 0.4	99.2 ± 0.4	**100.0 ± 0.0**
Kappa	88.0 ± 0.8	98.4 ± 0.4	97.4 ± 0.9	97.9 ± 0.5	88.6 ± 0.3	94.1 ± 0.5	92.5 ± 0.5	99.0 ± 0.1	**100.0 ± 0.0**
1	97.4 ± 1.5	97.3 ± 0.3	99.6 ± 0.1	99.4 ± 0.6	99.8 ± 0.0	**100.0 ± 0.0**	94.5 ± 0.2	**100.0 ± 0.0**	**100.0 ± 0.0**
2	99.7 ± 0.2	99.5 ± 0.2	99.5 ± 0.1	99.4 ± 0.6	99.3 ± 0.2	**100.0 ± 0.0**	99.9 ± 0.0	**100.0 ± 0.0**	**100.0 ± 0.0**
3	93.7 ± 1.5	99.4 ± 0.4	96.7 ± 0.1	98.8 ± 1.8	95.9 ± 1.3	**100.0 ± 0.0**	98.5 ± 0.1	**100.0 ± 0.0**	**100.0 ± 0.0**
4	97.8 ± 1.3	99.6 ± 0.4	98.3 ± 0.0	98.8 ± 1.9	95.9 ± 0.6	95.2 ± 0.4	**99.9 ± 0.0**	98.7 ± 0.6	**99.9 ± 0.0**
5	97.5 ± 1.1	99.5 ± 0.3	98.4 ± 0.2	99.0 ± 0.7	95.8 ± 2.2	97.4 ± 0.6	99.3 ± 0.3	99.0 ± 0.0	**99.9 ± 0.1**
6	99.5 ± 0.3	99.9 ± 0.1	99.0 ± 0.3	99.8 ± 0.3	99.8 ± 0.2	99.8 ± 0.1	**100.0 ± 0.0**	99.8 ± 0.1	**100.0 ± 0.0**
7	99.3 ± 0.2	99.9 ± 0.1	99.8 ± 0.1	97.9 ± 2.5	99.6 ± 0.3	**100.0 ± 0.0**	99.9 ± 0.0	99.8 ± 0.0	**100.0 ± 0.0**
8	88.9 ± 2.9	98.1 ± 1.2	95.7 ± 0.6	96.7 ± 2.2	80.2 ± 4.9	90.4 ± 0.1	87.0 ± 0.7	98.4 ± 0.1	**100.0 ± 0.0**
9	99.2 ± 0.3	99.3 ± 0.3	99.7 ± 0.0	99.8 ± 0.3	99.1 ± 0.4	99.8 ± 0.0	**100.0 ± 0.0**	99.8 ± 0.0	**100.0 ± 0.0**
10	88.8 ± 1.9	98.8 ± 0.8	98.0 ± 0.1	98.2 ± 2.1	94.0 ± 2.7	97.1 ± 0.5	95.0 ± 0.4	99.5 ± 0.1	**100.0 ± 0.0**
11	87.5 ± 4.6	94.4 ± 0.1	98.7 ± 0.9	97.3 ± 3.8	95.7 ± 0.8	99.6 ± 0.3	90.1 ± 0.7	99.8 ± 0.1	**99.9 ± 0.1**
12	98.0 ± 2.3	98.4 ± 0.4	**100.0 ± 0.0**	98.5 ± 2.1	94.8 ± 0.8	99.8 ± 0.0	**100.0 ± 0.0**	99.2 ± 0.0	**100.0 ± 0.0**
13	98.0 ± 0.8	95.8 ± 0.1	99.9 ± 0.0	97.4 ± 5.4	87.1 ± 0.7	93.7 ± 0.4	98.6 ± 0.6	98.3 ± 0.2	**100.0 ± 0.0**
14	89.6 ± 2.6	96.7 ± 2.5	98.9 ± 1.0	98.9 ± 1.0	85.0 ± 1.8	85.5 ± 0.5	98.4 ± 0.9	98.8 ± 0.7	**100.0 ± 0.0**
15	53.9 ± 7.6	97.7 ± 0.4	93.7 ± 2.4	96.5 ± 1.5	67.6 ± 4.1	81.4 ± 0.7	76.9 ± 0.1	97.5 ± 0.6	**100.0 ± 0.1**
16	90.8 ± 5.0	97.4 ± 0.8	99.6 ± 0.4	94.7 ± 5.1	98.7 ± 1.0	98.8 ± 0.2	99.0 ± 0.4	99.8 ± 0.0	**100.0 ± 0.0**


(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Fig. 13. Ground truth and classification maps of boundary test samples in SA. (a) Ground truth. (b) RBF-SVM. (c) HybridSN. (d) RSEN. (e) 3-D CNN. (f) 1-D CNN. (g) RDACN. (h) DBR. (i) SSFTT. (j) SSARL.

1-D and 2-D pretrained network, achieves the best performance in four classes. Fig. 12 displays the classification maps of different methods in the WHU dataset. It can be observed that SSFTT and RSEN produce regular misclassification points within class regions. The complex boundary even leads to the misclassification of background pixels.

F. Classification of Boundary Samples

SSARL focuses on improving the accuracy of boundary samples, which is a disadvantage of spatial feature from heterogeneous patches, and utilizes a different strategy of feature utilization. Therefore, in this section, we analyze the OA of

TABLE VII
Classification Performance of Several Methods Based on GAN

Dataset	Evaluating	HSGAN	3-D GAN	ARL-GAN	MSGAN	SSARL
IP (Train:5%)	OA (%)	74.1 ± 1.2	95.1 ± 0.6	98.3 ± 0.4	95.6 ± 0.5	**98.6 ± 0.4**
	AA (%)	65.6 ± 1.6	84.8 ± 2.6	95.0 ± 2.1	91.4 ± 1.5	**95.7 ± 1.8**
	Kappa	70.4 ± 1.5	94.9 ± 0.7	98.0 ± 0.4	95.0 ± 0.5	**98.4 ± 0.4**
PU (Train:3%)	OA (%)	85.7 ± 0.5	98.2 ± 0.5	99.8 ± 0.1	99.2 ± 0.2	**99.9 ± 0.0**
	AA (%)	81.4 ± 0.5	87.2 ± 0.9	**99.7 ± 0.2**	98.4 ± 0.3	**99.7 ± 0.1**
	Kappa	84.3 ± 4.0	94.9 ± 0.7	**99.8 ± 0.1**	98.9 ± 0.3	**99.8 ± 0.0**
SA (Train:1%)	OA (%)	88.3 ± 0.5	98.9 ± 0.3	**100.0 ± 0.0**	99.1 ± 0.1	**100.0 ± 0.0**
	AA (%)	92.3 ± 1.0	92.6 ± 0.8	**100.0 ± 0.0**	99.2 ± 0.3	**100.0 ± 0.0**
	Kappa	87.0 ± 0.6	97.9 ± 0.9	**100.0 ± 0.0**	99.0 ± 0.1	**100.0 ± 0.0**

TABLE VIII
Quantitative Classification Results of Different Methods in the WHU Dataset

	RBF-SVM	HybridSN	RSEN	3DCNN	1DCNN	RDACN	DBR	SSFTT	SSARL
OA (%)	90.2 ± 0.8	97.3 ± 0.7	98.3 ± 0.6	98.4 ± 0.5	94.6 ± 0.7	98.5 ± 0.4	97.8 ± 0.8	**98.7 ± 0.1**	98.5 ± 0.1
AA (%)	63.3 ± 1.5	94.6 ± 0.2	93.1 ± 0.5	96.4 ± 0.6	84.2 ± 0.9	95.8 ± 0.2	94.4 ± 0.1	96.5 ± 0.2	**97.1 ± 0.1**
Kappa	86.7 ± 1.4	96.4 ± 0.7	97.8 ± 0.3	97.9 ± 0.7	92.8 ± 0.8	98.1 ± 0.5	97.2 ± 0.1	**98.3 ± 0.1**	**98.3 ± 0.0**
1	95.0 ± 0.6	98.5 ± 0.5	99.7 ± 0.0	99.2 ± 0.2	97.5 ± 0.1	**99.7 ± 0.2**	**99.7 ± 0.1**	**99.7 ± 0.0**	**99.7 ± 0.0**
2	38.3 ± 3.7	90.4 ± 0.8	94.9 ± 0.6	95.1 ± 0.3	70.4 ± 0.7	98.7 ± 0.0	83.4 ± 0.1	98.8 ± 0.2	**99.0 ± 0.1**
3	44.7 ± 1.0	87.1 ± 0.5	88.5 ± 0.5	97.3 ± 0.4	75.7 ± 0.4	96.7 ± 0.3	92.1 ± 0.2	96.6 ± 0.0	**97.4 ± 0.0**
4	88.0 ± 1.9	97.7 ± 0.8	98.4 ± 0.1	95.5 ± 0.8	94.4 ± 0.2	**99.2 ± 0.1**	97.8 ± 0.4	99.0 ± 0.2	98.9 ± 0.1
5	56.4 ± 3.3	91.4 ± 0.6	87.3 ± 0.9	95.0 ± 0.3	71.8 ± 0.6	**96.64 ± 0.4**	86.9 ± 0.3	94.6 ± 0.2	95.0 ± 0.0
6	94.9 ± 1.2	96.8 ± 0.1	98.0 ± 0.3	98.7 ± 0.1	94.1 ± 0.2	98.7 ± 0.2	**99.5 ± 0.1**	98.6 ± 0.2	98.9 ± 0.0
7	99.8 ± 0.0	99.5 ± 0.1	**99.9 ± 0.1**	**99.9 ± 0.0**	99.7 ± 0.0	99.6 ± 0.0	**99.9 ± 0.0**	99.5 ± 0.6	99.4 ± 0.0
8	78.4 ± 2.6	89.5 ± 0.0	95.3 ± 0.4	92.1 ± 1.6	87.8 ± 0.5	88.6 ± 0.2	94.6 ± 0.1	93.5 ± 0.2	**95.8 ± 0.1**
9	34.9 ± 0.2	88.6 ± 0.4	92.3 ± 0.2	89.1 ± 0.4	84.5 ± 0.8	86.4 ± 0.5	**95.8 ± 0.3**	89.1 ± 0.1	93.1 ± 0.0

TABLE IX
Classification Performance on Boundary Test Samples

Dataset	Evaluating indices	RBF-SVM	HybridSN	RSEN	3DCNN	1DCNN	RDACN	DBR	SSFTT	SSARL
IP (5%)	OA (%)	62.4	82.4	87.2	58.8	53.7	89.7	92.3	90.1	**94.1**
PU (3%)	OA (%)	83.8	94.5	96.5	95.5	87.8	97.4	97.4	96.7	**97.5**
SA (1%)	OA (%)	87.6	96.3	96.3	94.7	86.7	93.6	91.2	95.1	**98.0**
WHU (0.3%)	OA (%)	69.8	83.9	84.1	82.2	79.9	79.7	**85.6**	82.6	84.9

boundary samples individually. The test sets only consist of boundary samples. Visualization results in SA are displayed in Fig. 13, where gray circles point to their magnified detail. And the quantitative classification performances are shown in Table IX.

The class boundaries of IP and SA are flatter, while those of PU and WHU are more irregular. The classification results in IP and WHU are worse than SA due to their complicated and unbalanced samples. Although the class boundaries of PU are complex, the classification result is fine. Compared with Section III-E, the OA of boundary test samples is lower than that of the entire test sets, 7.9%, 3.8%, 2.3%, and 13.4% lower on four datasets from HybridSN, and 7.3%, 2.1%, 4.1%, and 16.1% lower on four datasets from SSFTT. Therefore, it is proved that the spatial information of heterogeneous samples at the boundary is susceptible to be influenced by neighbor classes, which reduces the effectiveness of spatial features. Compared with other methods, SSARL extracts robust spectral and spatial features from the two branches, which are utilized by adding class consistency loss instead of concatenating them fixedly. Therefore, SSARL can adapt to both within-class and class boundary situations. The percentage of correct classification for SSARL is higher than that of other best methods by 1.8%, 0.1%, and 1.7%. On WHU, DBR achieves the best performance.

IV. Conclusion

In this article, we proposed a dual-branch SSARL for HSI classification based on a generative adversarial network. This method mainly focuses on training with limited labeled samples and utilization of spectral–spatial feature. Especially, we considered the relationship between pixel samples and complex heterogeneous image patch samples. We improved the ability of extracting feature from labeled and unlabeled samples by adding adversarial process. Two branches were, respectively, responsible for generating pixels and image patches and extracting their features. The class consistency loss was proposed to combine two branches. The experiments comprehensively proved the effectiveness of two-branch structure and class consistency loss. Compared with competent methods, SSARL performed better on four datasets. Moreover, the proposed SSARL aimed at improving the classification performance of boundary samples, which is often overlooked but has a negative impact on the overall classification results. We believe that there are two limitations to the proposed method. First, the training process is unstable. Although the proposed method can perform well, the constraints brought up by loss functions increase the difficulty of training. The second point is that the structure of the encoder is slightly simple, while some scenarios may require stronger feature representation capability.

References

[1] D. Landgrebe, "Hyperspectral image data analysis," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 17–28, Jan. 2002.

[2] N. M. Nasrabadi, "Hyperspectral target detection: An overview of current and future challenges," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 34–44, Jan. 2014.

[3] F. M. Lacar, M. M. Lewis, and I. T. Grierson, "Use of hyperspectral imagery for mapping grape varieties in the Barossa Valley, South Australia," in *Proc. Int. Geosci. Remote Sens. Symp.*, 2001, vol. 6, pp. 2875–2877.

[4] A. Brown, M. Walter, and T. Cudahy, "Hyperspectral imaging spectroscopy of a Mars analogue environment at the North Pole Dome, Pilbara Craton, Western Australia," *Aust. J. Earth Sci.*, vol. 52, no. 3, pp. 353–364, 2005.

[5] P. W. Yuen and M. Richardson, "An introduction to hyperspectral imaging and its application for security, surveillance and target acquisition," *Imag. Sci. J.*, vol. 58, no. 5, pp. 241–253, 2010.

[6] O. Carrasco, R. B. Gomez, A. Chainani, and W. E. Roper, "Hyperspectral imaging applied to medical diagnoses and food safety," *Proc. SPIE*, vol. 5097, pp. 215–221, 2003.

[7] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2094–2107, Jun. 2014.

[8] S. Hou, H. Shi, X. Cao, X. Zhang, and L. Jiao, "Hyperspectral imagery classification based on contrastive learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2021, Art. no. 5521213.

[9] L. Samaniego, A. Bárdossy, and K. Schulz, "Supervised classification of remotely sensed imagery using a modified K-NN technique," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 7, pp. 2112–2125, Jul. 2008.

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

[11] O. B. Ozdemir, E. Gedik, and Y. Yardimci, "Hyperspectral classification using stacked autoencoders with deep learning," in *Proc. 6th Workshop Hyperspectral Image Signal Process.: Evol. Remote Sens.*, 2014, pp. 1–4.

[12] G. E. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, pp. 1527–1554, 2006.

[13] J. H. Le, A. P. Yazdanpanah, E. E. Regentova, and V. Muthukumar, "A deep belief network for classifying remotely-sensed hyperspectral data," in *Proc. Int. Symp. Vis. Comput.*, 2015, pp. 682–692.

[14] Y. Chen, X. Zhao, and X. Jia, "Spectral–spatial classification of hyperspectral data based on deep belief network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2381–2392, Jun. 2015.

[15] W. Hu, Y. Huang, W. Li, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *J. Sens.*, vol. 2015, 2015, Art. no. 258619.

[16] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *Comput. Sci.*, 2015, doi: 10.48550/arXiv.1511.06434.

[17] M. Zhang, M. Gong, Y. Mao, J. Li, and Y. Wu, "Unsupervised feature extraction in hyperspectral images based on Wasserstein generative adversarial network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 5, pp. 2669–2688, May 2019.

[18] Q. Sun and S. Bourennane, "Unsupervised feature extraction based on improved Wasserstein generative adversarial network for hyperspectral classification," *Proc. SPIE*, vol. 11059, 2019, Art. no. 110590L.

[19] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 214–223.

[20] T. Song, Y. Wang, C. Gao, H. Chen, and J. Li, "MSLAN: A two-branch multidirectional spectral–spatial LSTM attention network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5528814.

[21] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Deep learning for hyperspectral image classification: An overview," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6690–6709, Sep. 2019.

[22] J. Yang, Y. Zhao, J. C.-W. Chan, and C. Yi, "Hyperspectral image classification using two-channel deep convolutional neural network," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2016, pp. 5079–5082.

[23] Y. Li, H. Zhang, and Q. Shen, "Spectral–spatial classification of hyperspectral imagery with 3D convolutional neural network," *Remote Sens.*, vol. 9, no. 1, 2017, Art. no. 67.

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

[25] L. Fang, S. Li, W. Duan, J. Ren, and J. A. Benediktsson, "Classification of hyperspectral images by exploiting spectral–spatial information of superpixel via multiple kernels," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 12, pp. 6663–6674, Dec. 2015.

[26] S. K. Roy, G. Krishna, S. R. Dubey, and B. B. Chaudhuri, "HybridSN: Exploring 3-D–2-D CNN feature hierarchy for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 277–281, Feb. 2020.

[27] X. Huang, M. Dong, J. Li, and X. Guo, "A 3-D-swin transformer-based hierarchical contrastive learning method for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5411415.

[28] Z. Zhao, D. Hu, H. Wang, and X. Yu, "Convolutional transformer network for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6009005.

[29] M. Nikzad, Y. Gao, and J. Zhou, "An attention-based lattice network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5526215.

[30] 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, 2022, Art. no. 5522214.

[31] Z. Sun, C. Wang, D. Li, and J. Li, "Semisupervised classification for hyperspectral imagery with transductive multiple-kernel learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 11, pp. 1991–1995, Nov. 2014.

[32] M. Seydgar, S. Rahnamayan, P. Ghamisi, and A. A. Bidgoli, "Semisupervised hyperspectral image classification using a probabilistic pseudo-label generation framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5535218.

[33] W. Li, J. Yin, B. Han, and H. Zhu, "Generative adversarial network with folded spectrum for hyperspectral image classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 883–886.

[34] Y. Zhan et al., "Semi-supervised classification of hyperspectral data based on generative adversarial networks and neighborhood majority voting," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2018, pp. 5756–5759.

[35] R. Hanachi, A. Sellami, I. R. Farah, and M. D. Mura, "Semi-supervised classification of hyperspectral image through deep encoder-decoder and graph neural networks," in *Proc. Int. Congr. Adv. Technol. Eng.*, 2021, pp. 1–8.

[36] H. Tang, Z. Huang, Y. Li, L. Zhang, and W. Xie, "A multiscale spatial–spectral prototypical network for hyperspectral image few-shot classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6011205.

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

[38] J. Yin, C. Qi, W. Huang, Q. Chen, and J. Qu, "Multibranch 3D-dense attention network for hyperspectral image classification," *IEEE Access*, vol. 10, pp. 71886–71898, 2022.

[39] H. Liang, W. Bao, X. Shen, and X. Zhang, "HSI-Mixer: Hyperspectral image classification using the spectral–spatial mixer representation from convolutions," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6013005.

[40] S. Zhang, J. Zhang, L. Xun, J. Wang, D. Zhang, and Z. Wu, "AM-FAN: Adaptive multiscale feature attention network for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6012005.

[41] A. Sauer, T. Karras, S. Laine, A. Geiger, and T. Aila, "StyleGAN-T: Unlocking the power of GANs for fast large-scale text-to-image synthesis," in *Proc. Int. Conf. Mach. Learn.*, 2023.

[42] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *Comput. Sci.*, pp. 2672–2680, 2014, doi: 10.48550/arXiv.1411.1784.

[43] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier GANs," in *Proc. Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 2642–2651.

[44] C. Sun, X. Zhang, H. Meng, X. Cao, and J. Zhang, “AC-WGAN-GP: Generating labeled samples for improving hyperspectral image classification with small-samples,” *Remote Sens.*, vol. 14, no. 19, 2022, Art. no. 4910.

[45] Z. He, H. Liu, Y. Wang, and J. Hu, “Generative adversarial networks-based semi-supervised learning for hyperspectral image classification,” *Remote Sens.*, vol. 9, no. 10, 2017, Art. no. 1042.

[46] Y. Yoshida and T. Miyato, “Spectral norm regularization for improving the generalizability of deep learning,” *Stat*, vol. 1050, p. 31, 2017.

[47] S. Zhang, X. Zhang, T. Li, H. Meng, X. Cao, and L. Wang, “Adversarial representation learning for hyperspectral image classification with small-sized labeled set,” *Remote Sens.*, vol. 14, no. 11, 2022, Art. no. 2612.

[48] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2242–2251.

[49] Y. Zhong, X. Hu, C. Luo, X. Wang, J. Zhao, and L. Zhang, “WHU-HI: UAV-borne hyperspectral with high spatial resolution (h^2) benchmark datasets and classifier for precise crop identification based on deep convolutional neural network with CRF,” *Remote Sens. Environ.*, vol. 250, 2020, Art. no. 112012.

[50] Z. Meng, J. Zhang, F. Zhao, H. Liu, and Z. Chang, “Residual dense asymmetric convolutional neural network for hyperspectral image classification,” in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2022, pp. 3159–3162.

[51] M. Ahmad, A. M. Khan, M. Mazzara, S. Distefano, M. Ali, and M. S. Sarfraz, “A fast and compact 3-D CNN for hyperspectral image classification,” *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 5502205.

[52] Y. Xu, B. Du, and L. Zhang, “Robust self-ensembling network for hyperspectral image classification,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 12, no. 6, pp. 1882–1897, Jun. 2019.

[53] T. Li, X. Zhang, S. Zhang, and L. Wang, “Self-supervised learning with a dual-branch ResNet for hyperspectral image classification,” *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 5512905.

Caihao Sun received the B.S. degree in aerospace science and technology in 2021 from Xidian University, Xi’an, China, where he is currently working toward the master’s degree in electronic information.

His main research interests include deep learning, computer vision, and hyperspectral image classification.

Xiaohua Zhang (Member, IEEE) received the B.S. and M.S. degrees in applied mathematics from Northwest University, Xi’an, Shaanxi Province, China, in 2000, and the Ph.D. degree in circuits and systems from Xidian University, Xi’an, China, in 2004.

In 2009, he was a Visiting Scholar with the Georgia Institute of Technology, Atlanta, GA, USA. He is currently an Associate Professor with the School of Artificial Intelligence, Xidian University, and a Member of the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education of China. His research interests include image processing, deep learning, computer vision, and remote sensing image processing.

Dr. Zhang is a Senior Member of the Chinese Institute of Electronics and the China Computer Federation.

Hongyun Meng received the Ph.D. degree in operating research and control from the School of Mathematics and Statistics, Xidian University, Xi’an, China, in 2004.

She is currently an Associate Professor with the School of Mathematics and Statistics, Xidian University. Her research interests include intelligent information processing, natural computing, and multiobjective optimization.

Xianghai Cao (Member, IEEE) received the B.E. and Ph.D. degrees in signal and information processing from the School of Electronic Engineering, Xidian University, Xi’an, China, in 1999 and 2008, respectively.

Since 2008, he has been with Xidian University, where he is currently an Associate Professor with the School of Artificial Intelligence. His research interests include hyperspectral image processing, pattern recognition, and deep learning.

Jinhua Zhang received the B.S. degree in applied mathematics in 2021 from Xidian University, Xi’an, China, where he is currently working toward the master’s degree in computer science.

His main research interests include deep learning, computer vision, and hyperspectral image unmixing.

Licheng Jiao (Fellow, IEEE) received the B.S. degree in electrical and computer science from Shanghai Jiao Tong University, Shanghai, China, in 1982, and the M.S. and Ph.D. degrees in theoretical electrician from Xi’an Jiaotong University, Xi’an, China, in 1984 and 1990, respectively.

Since 1992, he has been a Professor with the School of Electronic Engineering, Xidian University, Xi’an, where he is currently the Director of the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education of China. His research interests include image processing, natural computation, machine learning, and intelligent information processing.

Dr. Jiao is a Foreign Member of the Academia Europaea and the Russian Academy of Natural Sciences. He is a Fellow of the Institution of Engineering and Technology, the Chinese Association for Artificial Intelligence, the Chinese Institute of Electronics, the China Computer Federation, and the Chinese Association of Automation. He is a Councilor of the Chinese Institute of Electronics, a Committee Member of the Chinese Committee of Neural Networks, and an Expert of the Academic Degrees Committee of the State Council. He is the Chairman of the Awards and Recognition Committee and the Vice Board Chairperson of the Chinese Association of Artificial Intelligence.