Journal of Remote Sensing
A SCIENCE PARTNER JOURNAL

RESEARCH ARTICLE

Deep Contrastive Learning Network for Small-Sample Hyperspectral Image Classification

Quanyong Liu[1], Jiangtao Peng[1*], Genwei Zhang[2], Weiwei Sun[3], and Qian Du[4]

[1]Hubei Key Laboratory of Applied Mathematics, Faculty of Mathematics and Statistics, Hubei University, Wuhan, China. [2]Department of Gas Sensors and Chemometrics, State Key Laboratory of NBC Protection for Civilian, Beijing, China. [3]Department of Geography and Spatial Information Techniques, Ningbo University, Ningbo, China. [4]Department of Electrical and Computer Engineering, Mississippi State University, Mississippi State, USA.

*Address correspondence to: pengjt1982@hubu.edu.cn

Citation: Liu Q, Peng J, Zhang G, Sun W, Du Q. Deep Contrastive Learning Network for Small-Sample Hyperspectral Image Classification. *J. Remote Sens.* 2023;3:Article 0025. https://doi.org/10.34133/remotesensing.0025

Submitted 23 July 2022
Accepted 15 January 2023
Published 8 March 2023

Recently, deep learning methods have been widely used in hyperspectral image (HSI) classification and achieved good performance. However, the performance of these methods may be limited because of the scarcity of labeled samples in HSI data. To solve the small-sample classification problem, a deep contrastive learning network (DCLN) method is proposed in this paper. The proposed DCLN method first constructs contrastive groups and trains the network through contrastive learning. Then, it uses the trained network to extract spectral–spatial features of HSI pixels and generates pseudo-label for each unlabeled sample based on the spatial–spectral mixing distance. Finally, the pseudo-labeled samples with higher confidence are selected and added to the original training set to retrain the network. By gradually increasing pseudo-labeled samples and refining the contrastive learning network, the model shows good feature learning ability and classification performance with the limited labeled samples. Experimental results on 4 public HSI datasets demonstrate that the proposed DCLN method can achieve better performance than existing state-of-the-art methods.

Introduction

Hyperspectral image (HSI) is a 3-dimensional (3D) data cube acquired using a spectrometer, which combines spectral information and image information, reflecting the radiation characteristics and the spatial geometric relationship of the target [1–3]. HSI has more spectral bands and higher spectral resolution than traditional RGB images and multispectral remote sensing images [2]. Because of its rich spatial and spectral information, HSI is widely used in many fields such as agriculture, geological exploration, environment, and ecology. In these applications, land cover and land use classification is often needed [4].

The early HSI classification methods classify HSI pixels mainly based on spectral characteristics [5,6]. Because of spectral variations and noise, the spectral features of ground objects may be varied. Thus, it may result in misclassification, only relying on spectral information. To make full use of spatial information, researchers have proposed a series of spatial–spectral-based feature extraction and classification methods, such as composite kernels [7], joint sparse representation [8,9], extended morphological profiles [10], edge-preserving filtering [11], Gabor wavelets [12], local binary patterns [13], and discriminant analysis [14]. These methods are generally more accurate and reliable than the traditional spectral-based classification methods. However, these methods often require specific designed features, such as morphological features, Gabor features, local binary pattern features, etc. It may lead to poor classification performance when these features are not suitable for specific application tasks.

In order to automatically learn optimal features for specific application problems, many deep learning (DL) models are applied for feature extraction and classification of HSIs [15–20]. DL simulates the hierarchical working mode of the human visual system and builds a deep network model with a hierarchical structure on the basis of artificial neural networks. Given a certain number of labeled samples, DL classification algorithms can automatically learn hierarchical features of known samples and effectively predict labels for unknown samples. Chen et al. [16] proposed a deep hyperspectral classification network based on stacked auto-encoder, which uses spatial and spectral information to extract high-level features of HSIs. Considering that hyperspectral data has a 3D data structure, Chen et al. [17] further proposed a convolutional neural network (CNN)-based deep HSI classification method. Different from stacked auto-encoder, the CNN model adopts sparse connectivity and weight sharing to achieve effective feature extraction

and greatly reduces network parameters. After that, many scholars proposed a series of improved CNN models for HSI classification [19–23]. These models include a 5-layer CNN model [19], a 2-channel CNN model [20], a dual-channel CNN model [21], and a 3D CNN model [22,23]. To combine 3D and 2D convolutions, a hybrid spectral network (HybridSN) was proposed to extract discriminating spatial–spectral joint features [24]. In order to solve the gradient vanishing or explosion problems caused by the increase of the number of network layers, residual networks (ResNet) was proposed for HSI classification [25].

The aforementioned DL networks can automatically learn spatial and spectral features from the data. However, they usually require a large amount of labeled samples to train the network to achieve competitive classification performance. For example, HybridSN uses 30% of the total samples of Indian Pines (IP) as the training set, resulting in 3,000 training samples [24]. The large number of labeled samples is usually unavailable. For HSI, sample labels are typically obtained through field investigation or visual interpretation directly from high-resolution images. Field investigations are frequently used to obtain more accurate labels, but it is expensive and time-consuming, which greatly limits the number of training samples. In real situations, we usually face a small-sample classification problem, and it is difficult to obtain sufficient training samples to fully meet the training requirements of a DL network. To solve this problem, scholars have proposed a series of small-sample DL networks for HSI classification, such as data augmentation [26], lightweight network [27–30], few-shot learning [31–33].

Data augmentation techniques can be used to increase the number of labeled samples. The commonly used data augmentation strategies are translation, clipping, flip, rotation, and adding random noise, which can increase both the amount and diversity of samples Li et al. [26] proposed a pixel–block pair (PBP) method to augment the samples, where each pixel is built into a pixel block and a PBP is used for training. Haut et al. [34] proposed a random occlusion data augment (RODA) method for training CNN. It randomly occluded pixels in different rectangular spatial regions to generate training images with various levels of occlusion. Data augmentation methods can increase the number of training samples and, hence, improve the classification performance. However, these methods are prone to introduce noisy data in the process of data augmentation, which may lead to unstable classification performance.

To reduce the dependence on the number of training samples, many lightweight networks are proposed [27–30]. The lightweight network adopts lightweight ideas in network design, such as depthwise separable convolution (DSC) [30], grouped convolution [35], and other lightweight convolution methods, which can reduce network parameters and the amount of computation in the convolution process. Gao et al. [27] proposed a multiscale residual network (MSRN), which replaces the ordinary depthwise convolution in DSC with mixed depthwise convolution (MDConv). The MDConv mixes up multiple kernel sizes in a single depthwise convolution operation [27]. Jiang and Jia [28] proposed a 3D lightweight Siamese network (3DLSN). By simplifying the network, the lightweight model reduces the number of parameters and also the dependence on the training samples. However, the discriminative ability of lightweight network may also be affected.

Few-shot learning is to identify new classes from very few labeled samples [31]. Liu et al. [32] proposed a deep few-shot learning (DFSL) method for small-sample HSI classification, which creates a deep 3D ResNet to learn a metric space where samples from the same class are close and those from different classes are far away. Li et al. [33] proposed a deep cross-domain few-shot learning (DCFSL) method, which incorporates few-shot learning and domain adaptation into a unified framework. It utilized few-shot learning to solve the small-sample classification

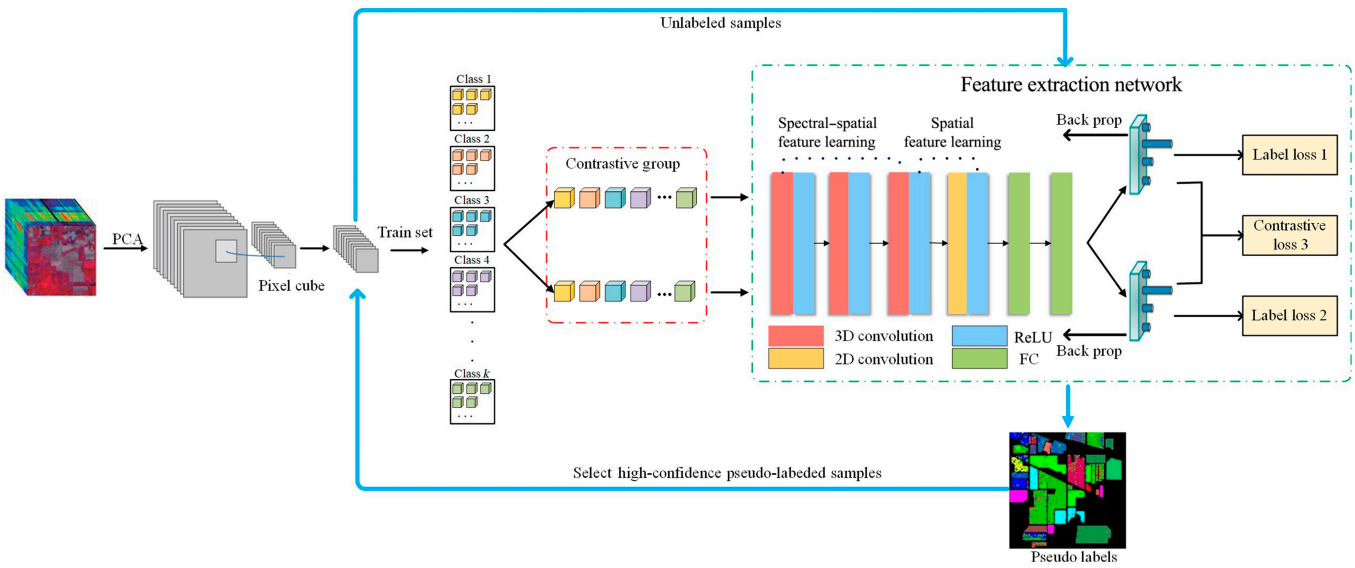

Fig. 1. The flow chart of the deep contrastive learning network (DCLN). The original HSI is first preprocessed to generate adaptive pixel cubes. Then, 2 contrastive groups are constructed and inputted to a spatial–spectral feature extraction network. On the basis of contrastive learning, the available limited labeled samples can be used to train the feature extraction network. The trained feature extraction network is further used to generate feature vectors and pseudo-labels for unlabeled samples. The whole network is refined based on the available labeled samples and the selected high-confidence pseudo-labeled samples. PCA, principal component analysis; FC, fully connected.

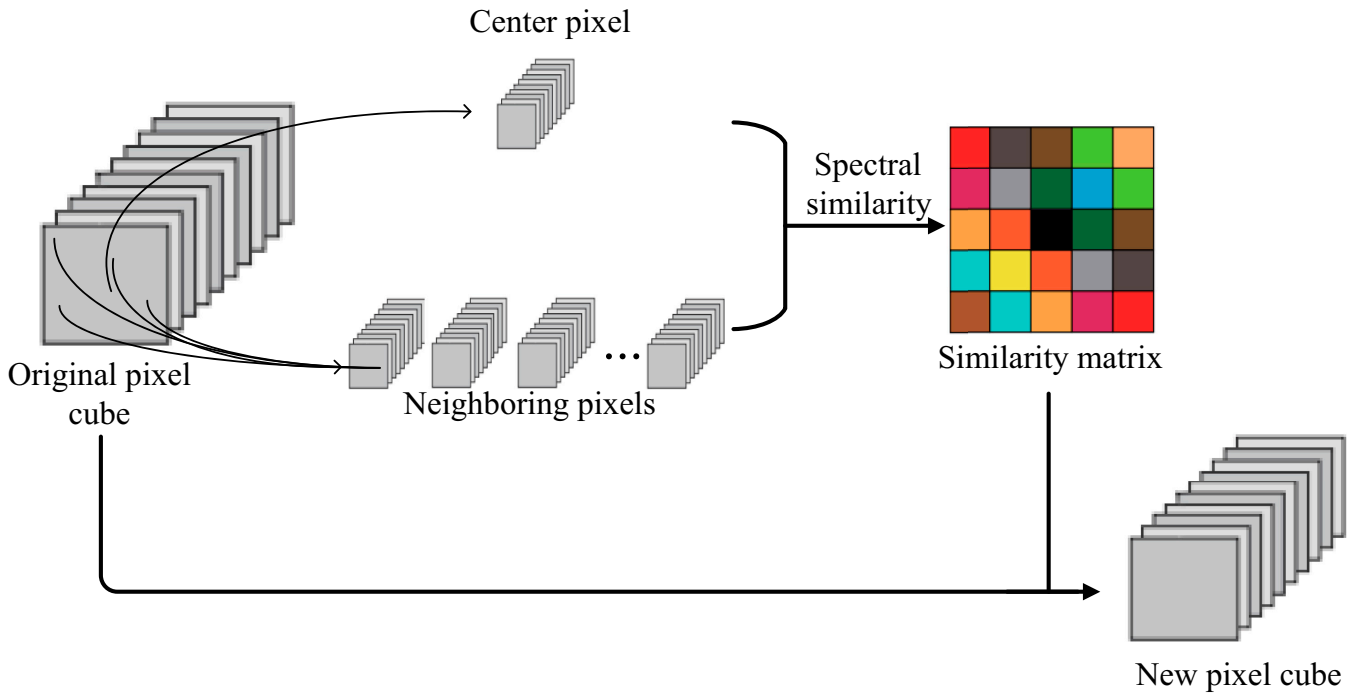

Fig. 2. The construction of new pixel cube.

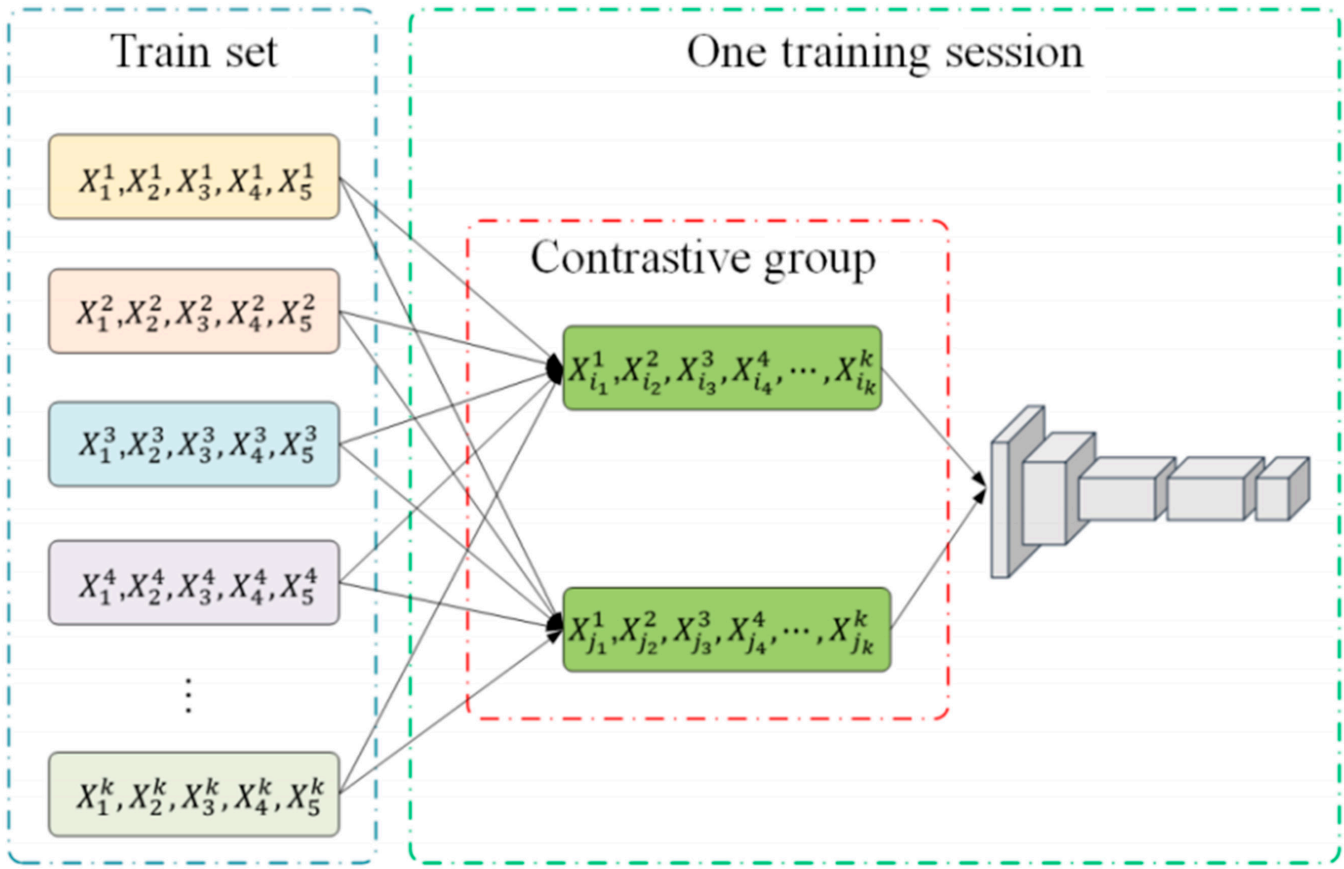

Fig. 3. Contrastive group construction. X_i^c, $i = 1, 2, 3, 4$, and 5 represents labeled samples of the cth class in the training set, and $X_{i_c}^c$, $X_{j_c}^c$ represents 2 samples selected from the cth class in the training set.

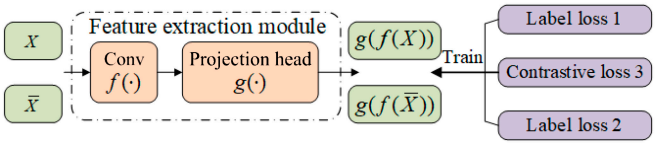

Fig. 4. Pipeline of contrastive learning network.

problem and adopted a conditional adversarial domain adaptation strategy to overcome domain shift between source and target domains [33].

Recently, self-supervised learning (SSL) that can obtain supervised information from the data itself has been proposed for small-sample classification. The SSL-based deep networks usually use a large number of unlabeled samples for pretraining and then use a small number of labeled samples for fine-tuning. Yue et al. [36] proposed an SSL with adaptive distillation (SSL-AD) method for HSI classification. It produces pseudo-labels for unlabeled samples by adaptive knowledge distillation and then uses these samples to train a progressive convolutional network (PCN). Contrastive learning is a branch of SSL that focuses on learning common features between instances of the same class and distinguishing differences between instances from different classes [37]. Liu et al. [38] proposed deep multi-view learning (DMVL) method, which generates multiple views for each sample using the band separation method, and trains the network by optimizing the contrastive learning loss for different views. Hou et al. [3] introduced self-supervised contrastive learning (SSCL) for HSI classification. Zhao et al. [39] introduce a contrastive SSL method for small-sample HSI classification. It designed an HSI-specific data augmentation module to generate sample pairs and developed a contrastive SSL model based on Siamese networks.

The aforementioned contrastive learning methods directly use the original idea of contrastive learning in generating positive and negative sample pairs. For a dataset X of size N, a data augmentation method is used to generate another enhanced dataset X' of size N. In a training batch of size $2N$ (dataset X and enhanced data X'), for sample, x_i in X, only one sample x_i' in X' is the corresponding positive sample, while other $2N-2$ samples are negative samples. For hyperspectral images, the number of available land cover classes (e.g., k) is small. When $k < N$, a training batch will contain more than 2 samples from the same class. In this case, some samples having the same label as x_i will be indicated as negative samples, which will bring a large deviation to the calculation of the contrastive loss. Incorporating label information into the construction of contrastive group is an effective strategy to solve the contrastive loss deviation problem. Khosla et al. [40] proposed a supervised contrastive learning method, which extends the self-supervised batch contrastive method to a supervised case. Different from previous SSCL, the supervised contrastive learning allows multiple positives per anchor. To fully use the available label information, a supervised deep contrastive learning network (DCLN) method is proposed for small-sample HSI classification in this paper. It constructs positive and negative sample pairs by randomly selecting labeled samples, which ensures that there are no negative sample pairs of the same category in each patch. The feature extraction network is trained by optimizing the contrastive loss between pairs of positive and negative samples. Therefore, we can extract the discriminative features of each sample. A spatial–spectral mixing distance (SSMD) is used to

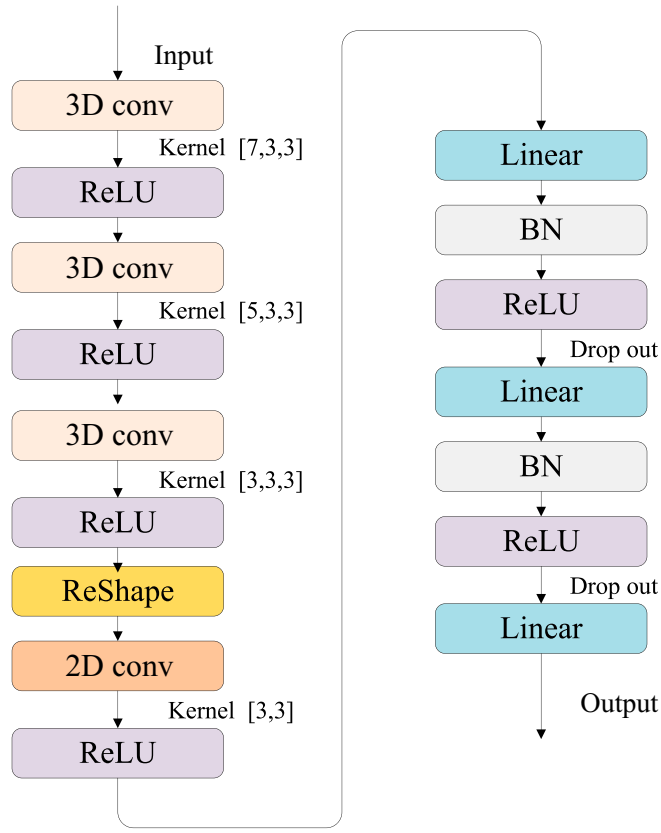

Fig. 5. Structure of the feature extraction module.

generate pseudo-label for each unlabeled sample, and a confidence measure criterion (CMC) is designed to select high-confidence pseudo-labeled samples to retrain the network.

The contributions of this paper are as follows:

1. We design a supervised DCLN method for small-sample HSI classification. It can realize effective spatial–spectral feature extraction, pseudo-label learning, and classification in the case of limited training samples.
2. We propose a new sample generation strategy for contrastive learning, which constructs contrastive groups on the basis of available labeled samples. The construction of the contrast group makes full use of the information of the labeled samples and stabilizes the contrastive learning performance of the model, because the labels of the samples can represent the similarity or dissimilarity of the samples, and the samples at different positions in each group are dissimilar.
3. We introduce an SSMD and a CMC to generate pseudo-labeled samples and to select high-confidence pseudo-labeled samples to retrain the network.

The rest of the paper is organized as follows. Materials and Methods describe the proposed DCLN model. Results provide the experimental results. The discussion on the proposed method is shown in Discussion. Conclusion concludes the paper.

Materials and Methods

Figure 1 shows the procedure of the proposed DCLN method, which includes 4 parts: data preprocessing, contrastive group

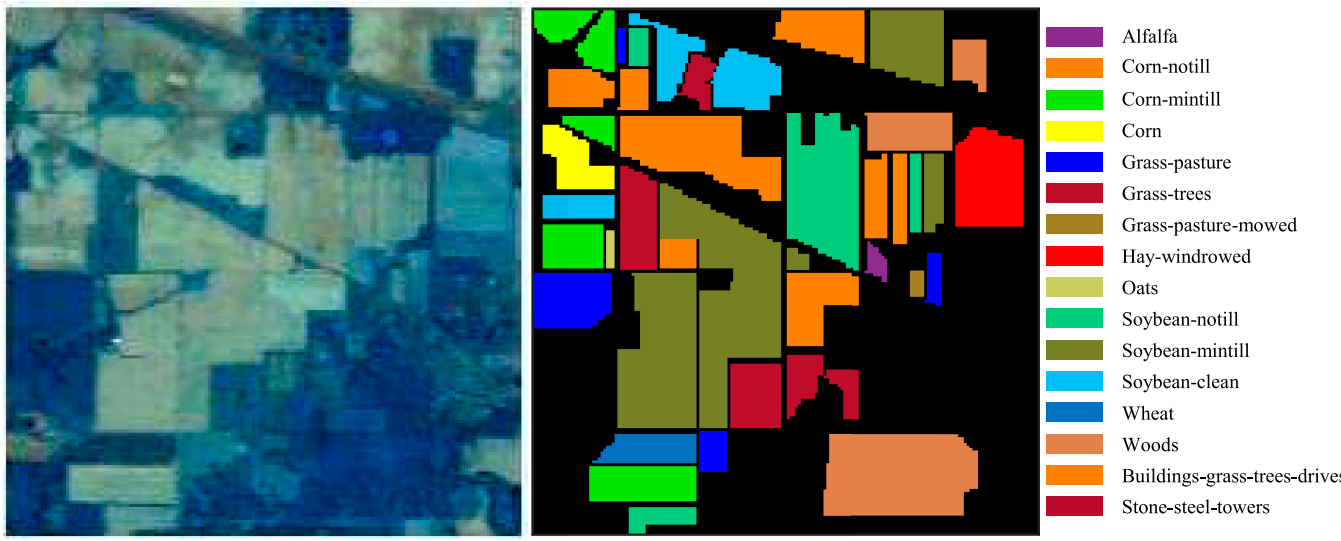

Fig. 6. The pseudo-color composite image and ground-truth map of the IP dataset.

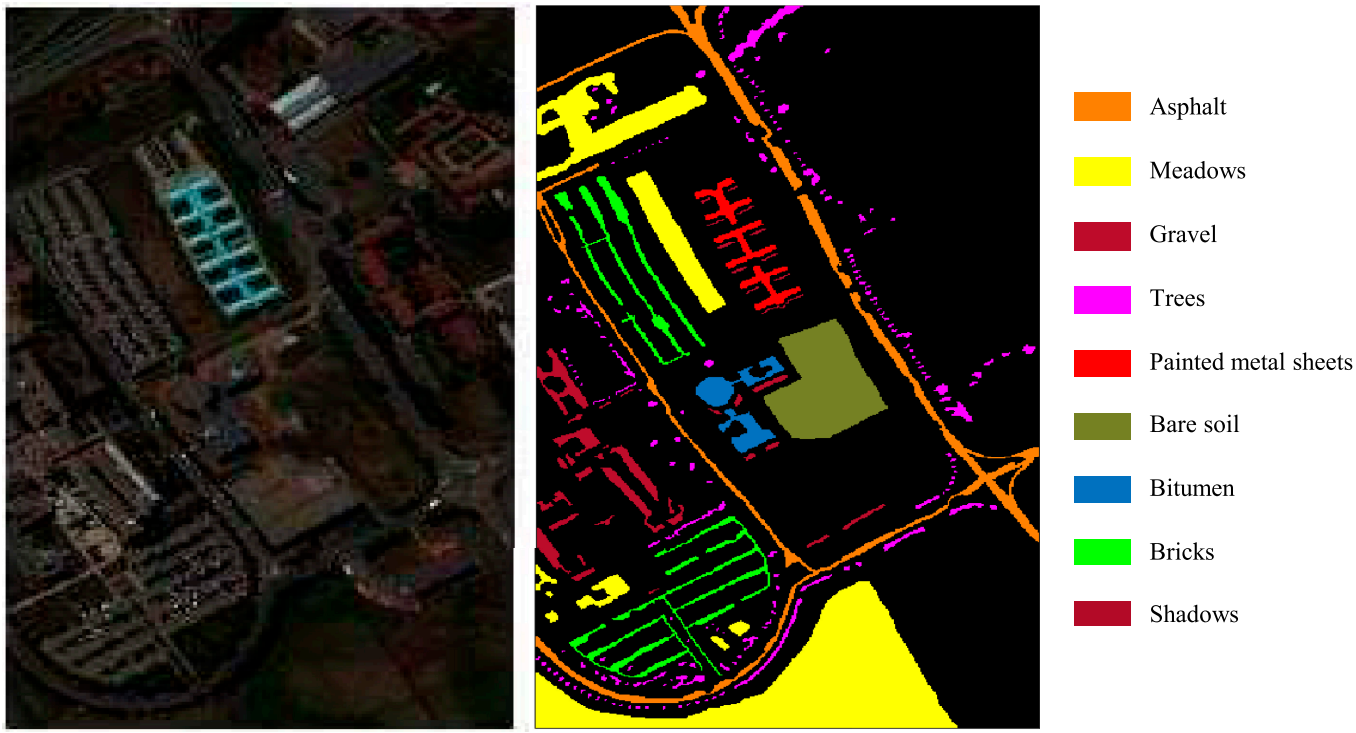

Fig. 7. The pseudo-color composite image and ground-truth map of the UP dataset.

construction, feature extraction network, and pseudo-label learning.

Data preprocessing

To alleviate the effect of high dimensionality of HSI and decrease the computational cost, traditional principal component analysis is applied to reduce the dimensionality of HSI a lower one (e.g., 20). In order to make full use of spatial neighborhood information, the dimension-reduced HSI data are further divided into small overlapping 3D pixel cubes of size $w \times w \times 20$ ($w = 11$ is used in the experiment), and the cube label is determined by the label of the center pixel in the cube.

In each 3D pixel cube, to eliminate the influence of inhomogeneous pixels, we compute the similarity between neighboring pixels and the center pixel in the cube and highlight the pixels with the highest similarities to construct the new pixel cube, as shown in Fig. 2. The similarity between 2 pixels x_i and x_j is calculated as:

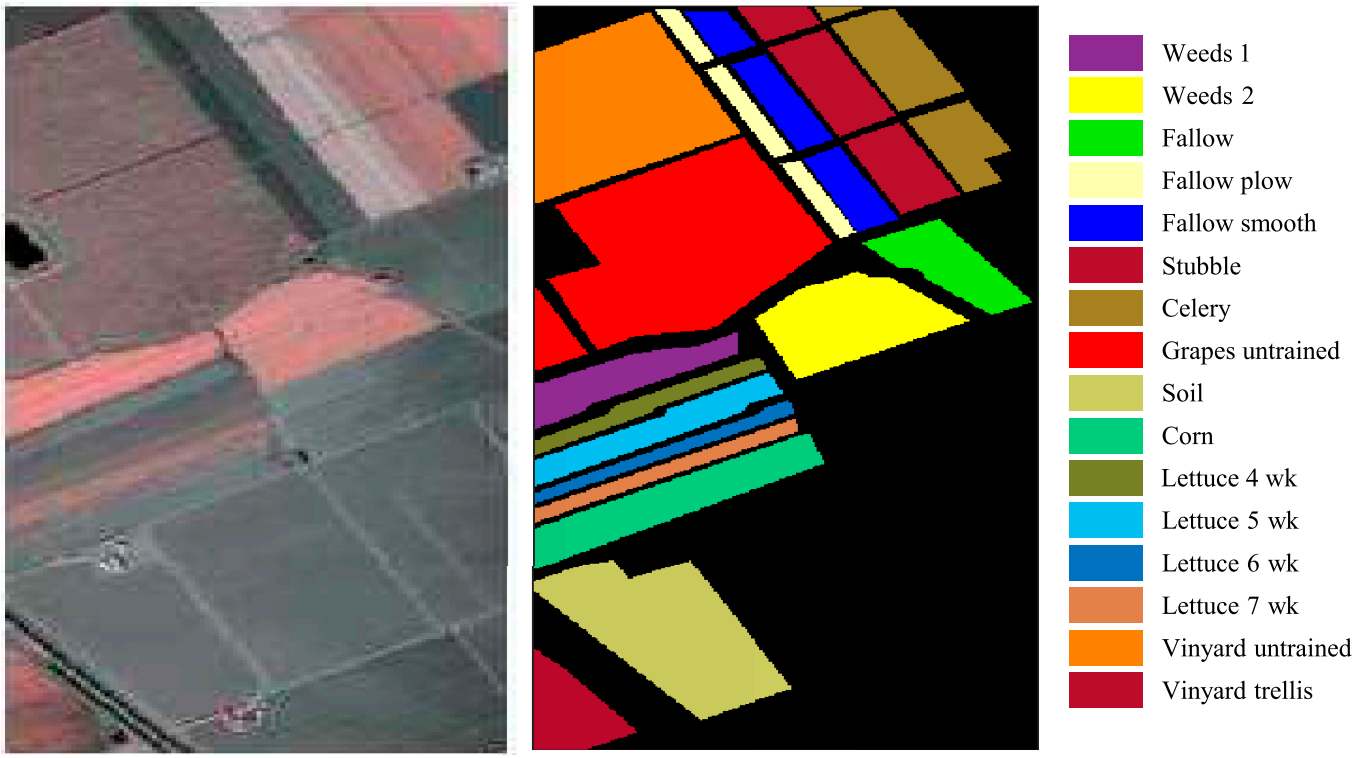

Fig. 8. The pseudo-color composite image and ground-truth map of the SA dataset.

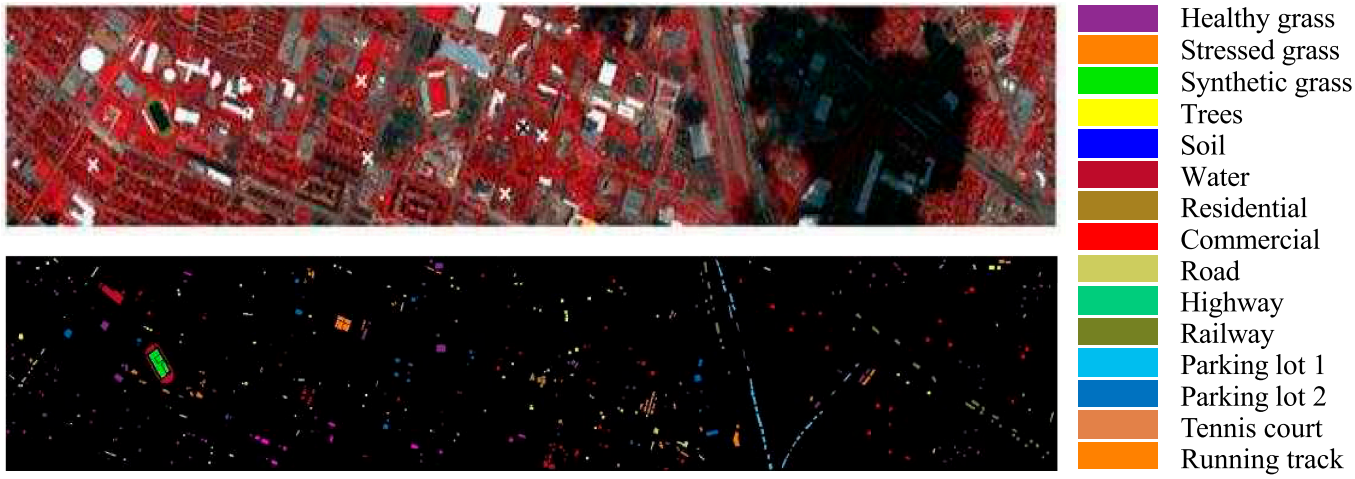

Fig. 9. The pseudo-color composite image and ground-truth map of the HOU dataset.

$$s_{i,j} = \exp\left(- ||x_i - x_j|| \right) \quad (1)$$

By ranking the pixels in the cube based on $s_{i,j}$ and selecting the first m similarity pixels, a new pixel cube is obtained by setting the spectral information of the remaining dissimilar pixels to 0. In the experiment, m is set as the 80% of the number of neighboring pixels.

Contrastive group construction

The contrastive group construction is shown in Fig. 3. Assume that a dataset has k classes and each class has several labeled training samples, say, 5, i.e., a small-sample case as shown in Results. Thus, the total number of training samples is $5 \times k$. We randomly select 2 labeled samples from each class in the training set and put these 2 samples into 2 individual groups. Then, we can obtain 2 contrastive groups, where each group has k samples from different classes. The 2 contrastive groups consist of a training batch.

Even if there are only 5 labeled samples in each class, the possible number of different training batches is 5^k, which greatly increases the number of nonrepetitive training of the model. In addition, we use the labeled training samples to generate contrastive groups to ensure that the samples in the same

Table 1. The number of training and testing sample for each dataset.

Indian Pines				University of Pavia			Salinas			Houston		
No.	Class	Train	Test	Class	Train	Test	Class	Train	Test	Class	Train	Test
1	Alfalfa	5	41	Asphalt	5	6,626	Brocoli 1	5	2,004	Healthy grass	5	1,251
2	Corn-notill	5	1,423	Meadows	5	18,644	Brocoli 2	5	3,721	Stressed grass	5	1,254
3	Corn-mintill	5	825	Gravel	5	2,094	Fallow	5	1,971	Synthetic grass	5	697
4	Corn	5	232	Trees	5	3,059	Fallow plow	5	1,389	Trees	5	1,244
5	Grass-pasture	5	478	Metal sheets	5	1,340	Fallow smooth	5	2,673	Soil	5	1,242
6	Grass-trees	5	725	Bare soil	5	5,024	Stubble	5	3,954	Water	5	325
7	Grass-pasture-mowed	5	23	Bitumen	5	1,325	Celery	5	3,574	Residential	5	1,268
8	Hay-windrowed	5	473	Bricks	5	3,677	Grapes untrained	5	11,266	Commer-cial	5	1,244
9	Oats	5	15	Shadows	5	942	Soil	5	6,198	Road	5	1,252
10	Soybean-notill	5	967				Corn	5	3,273	Highway	5	1,227
11	Soybean-mintill	5	2,450				Lettuce 4 wk	5	1,063	Railway	5	1,235
12	Soybean-clean	5	588				Lettuce 5 wk	5	1,922	Parking lot 1	5	1,233
13	Wheat	5	200				Lettuce 6 wk	5	911	Parking lot 2	5	469
14	Woods	5	1,260				Lettuce 7 wk	5	1,065	Tennis court	5	428
15	Buildings-grass	5	381				Vinyard untrained	5	7,263	Running track	5	660
16	Stone-steel	5	88				Vinyard vertical	5	1,802			
Total		80	10,169		45	42,731		80	54,049		75	15,029

group are from different classes and, hence, dissimilar, and samples only at the corresponding position of 2 groups are similar. That is, for each $X_{i_c}^c$, except for one sample $X_{j_c}^c$ at the corresponding position, the remaining $2k-2$ samples are negative samples. The construction of 2 contrastive groups can facilitate the feature contrastive learning because the label of samples can indicate their similarity or dissimilarity, and samples at different positions of each group are dissimilar.

Feature extraction module based on contrastive learning

Here, we introduce the feature extraction module. For convenience, we specify the index of each sample and express 2 contrastive groups as $X = \left\{x_1^1, x_3^2, x_5^3 \cdots x_{2i-1}^i \cdots x_{2k-1}^k\right\}$ and $\overline{X} = \left\{x_2^1, x_4^2, x_6^3 \cdots x_{2i}^i \cdots x_{2k}^k\right\}$.

The flow chart of feature contrastive learning network is shown in Fig. 4. For input contrastive groups X and $\overline{X}$, the feature extraction module outputs the features $g(f(X))$ and $g\left(f\left(\overline{X}\right)\right)$ through a series of convolutional layers followed by a series of fully connected layers. On the basis of the deep features, the label losses and contrastive loss are computed and used to train the contrastive learning network.

Feature extraction module

As shown in Fig. 5, the 3D and 2D convolutional layers are employed to extract the spatial and spectral features of HSI. The model uses the rectified linear unit (ReLU) as the activation function to enhance the stability and uses the batch normalization layer and the dropout technique to suppress the overfitting and improve the generalization ability of the model. In addition, the batch normalization layer can accelerate the convergence speed of the model.

Contrastive learning

As shown in Fig. 4, the parameters of the network are updated by minimizing the sum of 3 losses. The label losses l_1 and l_2 refer

Table 2. Classification results (%) on the IP dataset. Values in boldface indicate the best accuracy among all methods.

Class	RODA	3DLSN	SSCL	TRCL	DMVL	A2S2K	DFSL	SSL-AD	DCLN
1	92.68	91.43	87.18	93.10	92.31	36.53	82.41	**100**	**100**
2	55.31	58.22	36.31	65.06	67.76	68.15	57.12	67.24	**95.47**
3	51.52	66.54	54.56	**87.58**	77.28	65.64	62.47	65.12	85.36
4	37.50	87.61	34.78	91.82	**95.85**	49.04	89.24	55.17	82.33
5	85.36	92.16	73.53	**91.85**	91.58	86.01	64.24	81.75	72.23
6	88.14	99.30	89.21	**99.44**	93.94	97.58	70.15	80.12	89.66
7	**100**	**100**	**100**	**100**	**100**	19.00	97.42	**100**	92.39
8	61.31	99.14	96.82	98.92	99.34	47.06	91.42	**100**	**100**
9	93.33	**100**	**100**	**100**	**100**	25.86	93.24	44.26	91.67
10	15.31	60.56	49.74	65.13	75.42	64.71	64.70	66.48	**88.90**
11	30.57	34.25	33.17	47.74	61.07	74.96	60.17	83.47	**95.70**
12	9.18	58.08	43.17	68.23	85.34	56.76	67.04	71.69	**96.55**
13	96.00	97.42	99.49	99.47	**100**	76.92	84.28	95.42	93.02
14	68.17	63.80	74.40	75.56	78.71	92.32	86.27	**95.23**	73.57
15	40.68	67.20	65.96	72.09	**90.98**	51.46	92.84	84.15	79.22
16	**100**	95.12	96.51	96.05	98.63	57.14	84.57	81.69	87.13
OA	48.74	62.68	55.00	71.30	76.85	69.08	70.25	79.81	**83.74**
AA	64.07	79.43	70.93	84.50	88.01	60.57	77.97	79.49	**88.95**
κ	42.44	58.83	49.92	68.02	73.98	65.21	68.38	77.81	**80.23**

to the cross entropy between the label of samples in the contrastive groups and the features $g(f(X))$ and $g\left(f\left(\overline{X}\right)\right)$, respectively. These 2 label losses ensure that the extracted features have the class discriminant ability. The contrastive loss l_3 refers to the difference between the features extracted from 2 contrastive groups and is calculated in the form of noise contrastive estimation (NCE)-based loss function. Let z_i and z_j denote feature vectors extracted by the network corresponding to inputs x_i^c and x_j^c, respectively.

The NCE loss L_{ij} between features z_i and z_j is:

$$L_{ij} = -\log \frac{\exp\left(\frac{s\left(z_i,z_j\right)}{\tau}\right)}{\sum_{p=1}^{2k} 1_{[p\neq i]} \exp\left(\frac{s\left(z_i,z_p\right)}{\tau}\right)} \tag{2}$$

where τ is a temperature coefficient, and $s(z_i,z_j)$ measures the similarity between z_i and z_j. Here, the cosine similarity is used:

$$s\left(z_i,z_j\right) = \frac{z_i \cdot z_j}{\|z_i\| \times \left\|z_j\right\|} \tag{3}$$

For the positive samples x_{2i-1}^i and x_{2i}^i, other $2k-2$ samples in a batch are from different classes and considered as negative samples. Considering that $L_{2i-1,2i} \neq L_{2i,2i-1}$ in Eq. 2, the final average contrastive loss l_3 of a batch is calculated by the following equation:

$$l_3 = \frac{1}{2k} \sum_{c=1}^{k} \left(L_{2c-1,2c} + L_{2c,2c-1}\right) \tag{4}$$

It is clear that the contrastive loss l_3 ensures that the extracted features from the same class are more similar and the extracted features from different classes are dissimilar.

Acquisition of pseudo-labeled samples

For convenience, it is assumed that an unlabeled sample set can be expressed as $U = \left\{u^1,u^2,u^3 \cdots u^{n_1}\right\}$ and a labeled sample set is expressed as $V = \left\{v^1,v^2,v^3 \cdots v^{n_2}\right\}$, where n_1 and n_2 are the number of unlabeled and labeled samples, respectively. The acquisition of pseudo-labeled samples includes 2 steps. The first step is the generation of pseudo-labels for all unlabeled samples in the set U. Considering that the pseudo-labels are not real labels and may be inaccurate, we only select part of pseudo-labeled samples with high confidence for the next training.

Generation of pseudo-labels

Here, we calculate the SSMD between unlabeled and labeled samples [35], obtain the similarity between unlabeled samples and each class, and generate the pseudo-label for each unlabeled sample.

We first calculate the SSMD between unlabeled and labeled samples. The Euclidean distance (ED) is used to measure the spatial distance between pixels u^i and v^j:

$$\mathrm{ED}\left(u^i,v^j\right) = \left\|u^i - v^j\right\|_2 \tag{5}$$

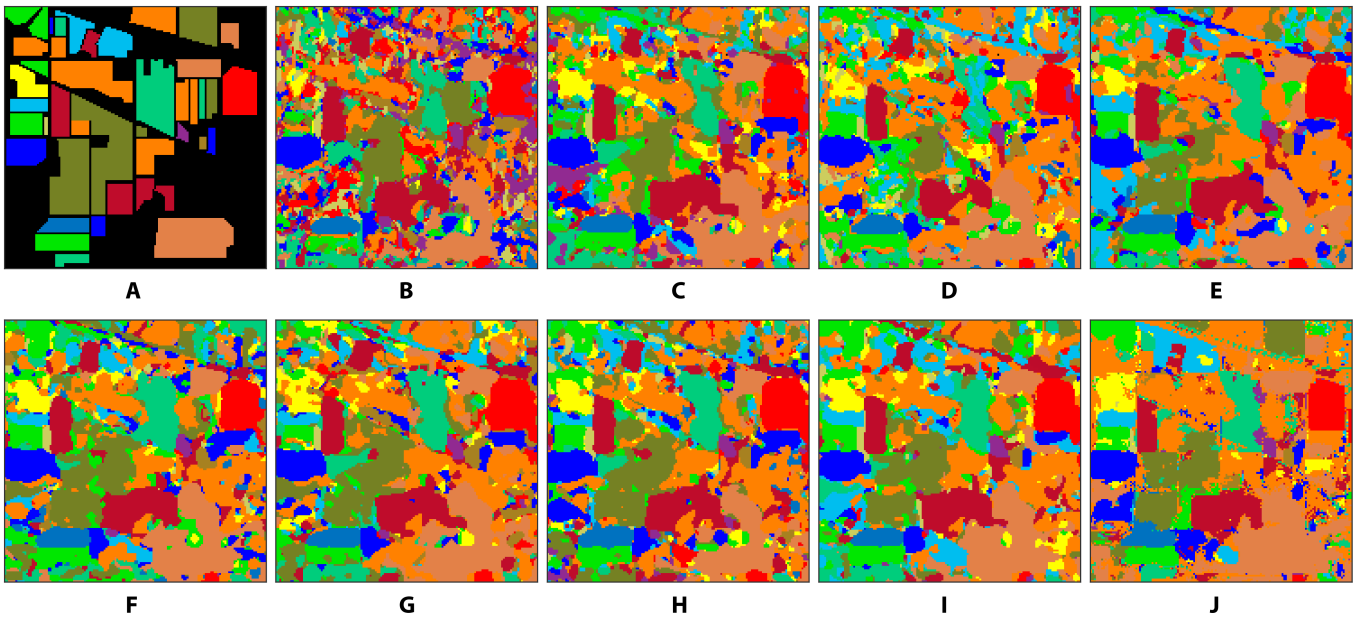

Fig. 10. The classification map on the IP dataset. (A) Ground truth, (B) RODA (48.74%), (C) 3DLSN (62.68%), (D) SSCL (55.00%), (E) TRCL (71.30%), (F) DMVL (76.85%), (G) A2S2K (69.08%), (H) DFSL (70.25%), (I) SSL-AD (79.81%), and (J) DCLN (83.74%).

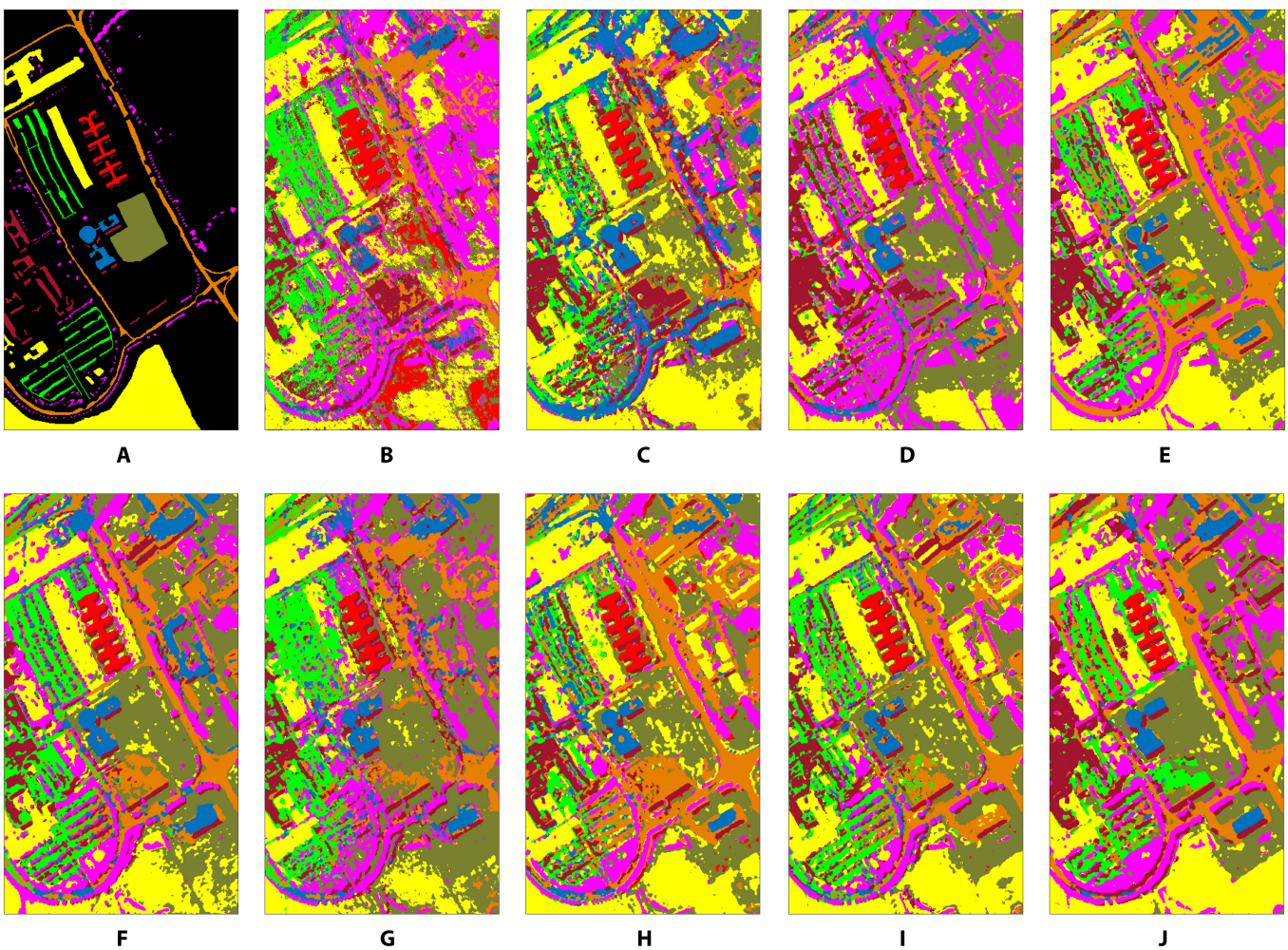

Fig. 11. The classification map on the UP dataset. (A) Ground truth, (B) RODA (54.30%), (C) 3DLSN (67.56%), (D) SSCL (62.65%), (E) TRCL (87.23%), (F) DMVL (85.55%), (G) A2S2K (68.50%), (H) DFSL (78.58%), (I) SSL-AD (82.41%), and (J) DCLN (89.70%).

Table 3. Classification results (%) on the UP dataset.

Class	RODA	3DLSN	SSCL	TRCL	DMVL	A2S2K	DFSL	SSL-AD	DCLN
1	35.53	57.48	56.81	76.02	82.08	92.51	**98.24**	61.59	80.68
2	51.80	70.35	64.51	90.61	90.45	84.82	**98.45**	95.34	89.81
3	84.78	**89.02**	79.14	87.99	93.19	55.66	48.04	54.81	87.15
4	86.77	61.60	73.50	**96.31**	88.72	69.45	50.94	85.13	95.98
5	**100**	**100**	**100**	**100**	**100**	96.32	99.92	99.93	**100**
6	49.78	78.95	62.09	84.13	85.32	57.00	77.96	86.39	**99.58**
7	94.72	95.85	91.78	**100**	98.63	40.64	73.53	80.12	99.25
8	20.44	19.06	14.38	78.97	35.52	45.85	84.24	**90.58**	75.12
9	96.82	97.35	92.47	99.46	97.86	68.21	75.68	82.53	**100**
OA	54.30	67.56	62.65	87.23	85.55	68.50	78.58	82.41	**89.70**
AA	68.96	74.41	70.52	90.39	85.75	67.83	78.56	81.82	**91.95**
κ	44.88	59.53	53.93	83.32	81.12	64.28	76.17	80.74	**86.32**

Table 4. Classification results (%) on the SA dataset.

Class	RODA	3DLSN	SSCL	TRCL	DMVL	A2S2K	DFSL	SSL-AD	DCLN
1	**100**	**100**	**100**	**100**	**100**	**100**	97.45	99.6	**100**
2	99.54	96.27	99.84	99.97	**100**	99.80	94.87	99.15	**100**
3	94.38	98.83	89.15	99.54	**100**	96.40	98.74	**100**	**100**
4	56.82	85.19	83.74	84.46	**99.93**	83.27	92.47	98.37	99.21
5	98.8	94.99	97.49	**99.93**	98.98	89.22	97.85	99.74	98.99
6	98.48	99.14	95.35	99.85	99.24	**99.97**	96.84	99.91	99.90
7	99.5	**100**	97.06	99.83	99.92	98.48	98.74	**100**	**100**
8	70.88	72.57	62.25	25.07	89.16	89.48	92.14	**94.01**	81.48
9	81.02	94.92	99.94	99.74	**100**	99.45	97.57	98.98	**100**
10	37.85	44.79	94.2	96.52	**97.82**	90.05	93.47	92.52	94.78
11	87.52	91.27	99.43	94.45	97.41	97.06	90.47	94.25	**99.44**
12	98.29	95.01	79.72	71.87	**100**	97.41	94.15	96.29	98.28
13	83.48	99.56	99.89	**100**	99.89	97.09	87.45	89.69	95.94
14	83.52	94.85	96.53	92.12	**99.43**	81.96	70.47	67.79	98.69
15	37.93	67.56	82.83	93.12	68.27	60.39	68.75	52.08	**95.51**
16	87.37	96.18	91.68	91.13	99.44	99.48	95.84	**99.94**	96.95
OA	76.77	84.42	86.21	81.22	94.13	88.08	84.44	85.95	**94.88**
AA	82.21	89.45	91.83	90.48	96.84	92.47	91.70	92.65	**97.45**
κ	74.27	82.73	84.74	79.37	93.46	87.83	81.53	84.46	**94.32**

The spectral similarity is obtained by calculating the relative entropy between feature vectors of pixels. Here, the Kullback–Leibler (KL) divergence is used to measure the spectral distance between pixels,

$$\mathrm{KL}\left(u^i \parallel v^j\right) = \sum_{q=1}^{d} u_q^i \left(\log \frac{u_q^i}{v_q^j}\right) \tag{6}$$

$$\mathrm{KL}\left(v^j \parallel u^i\right) = \sum_{q=1}^{d} v_q^j \left(\log \frac{v_q^j}{u_q^i}\right) \tag{7}$$

where u_q^i, v_q^j represent the qth component of feature vectors u^i and v^j, respectively, and d is the dimension of u^i and v^j. Because the KL divergence is asymmetric, the final spectral distance is represented as:

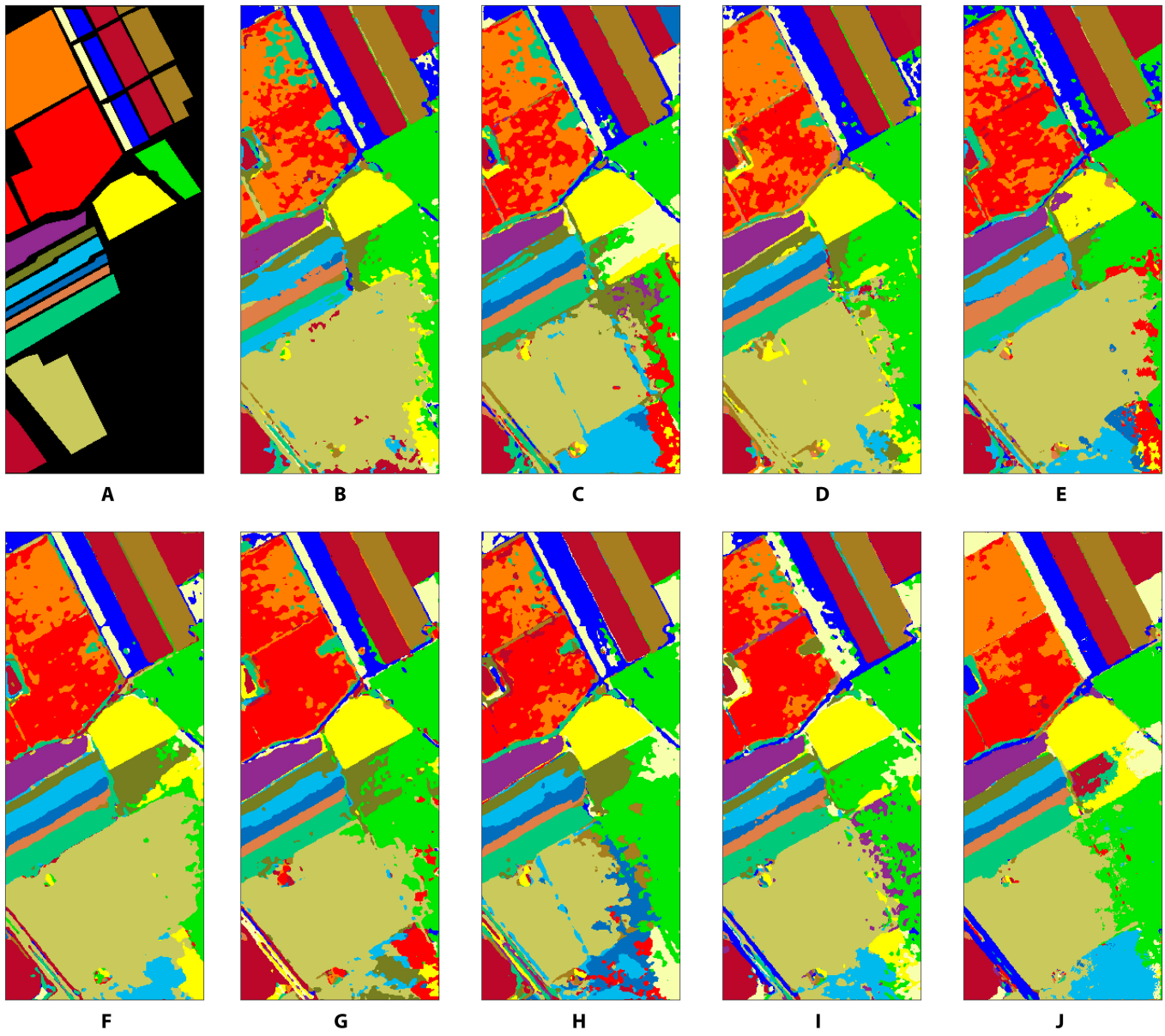

Fig. 12. The classification map on the SA dataset. (A) Ground truth, (B) RODA (76.77%), (C) 3DLSN (84.42%), (D) SSCL (86.21%), (E) TRCL (81.22%), (F) DMVL (93.13%), (G) A2S2K (88.08%), (H) DFSL (84.44%), (I) SSL-AD (85.95%), and (J) DCLN (94.88%).

$$\mathrm{KL}(u^i, v^j) = \mathrm{KL}(u^i \parallel v^j) + \mathrm{KL}(v^j \parallel u^i) \tag{8}$$

Combining the spatial and spectral distances in Eqs. 5 and 8, the SSMD is obtained:

$$\mathrm{SSMD}(u^i, v^j) = \sqrt{\mathrm{ED}(u^i, v^j) \times \mathrm{KL}(u^i, v^j)} \tag{9}$$

On the basis of the SSMD, we define the distance from each unlabeled sample to each class as:

$$\mathrm{SSMD}(u^i, c) = \min\left\{ \mathrm{SSMD}\left(u^i, v_{ch}^j\right), h = 1, \cdots, n_c \right\} \tag{10}$$

where v_{ch}^j represents the labeled sample belonging to the cth class. The distance can be further transferred to a probability as:

$$p(u^i, c) = \frac{\exp\left(-\mathrm{SSMD}(u^i, c)\right)}{\sum_{c=1}^{k} \exp\left(-\mathrm{SSMD}(u^i, c)\right)}, c = 1, \cdots, k \tag{11}$$

The pseudo-label (PseLab) of an unlabeled sample u^i is calculated as:

$$\mathrm{PseLab}(u^i) = \underset{c}{\mathrm{argmax}}\left\{ p(u^i, c), c = 1, \cdots, k \right\} \tag{12}$$

Selection of pseudo-labeled samples

Considering that mistakenly labeled samples have a negative impact on the training of the model, only a small number of pseudo-labeled samples with high confidence are selected and added to the training set to retrain the model for better performance.

Table 5. Classification results (%) on the HOU dataset.

Class	RODA	3DLSN	SSCL	TRCL	DMVL	A2S2K	SSL-AD	**DCLN**
1	52.13	66.48	56.82	74.98	78.44	52.94	47.03	**89.33**
2	85.76	81.44	85.99	**96.87**	92.86	87.80	96.31	84.15
3	96.10	97.40	95.52	97.53	94.78	97.97	97.83	**100**
4	84.19	87.58	87.25	90.86	87.78	**92.07**	87.47	89.27
5	98.71	**100**	95.39	**100**	**100**	**100**	**100**	**100**
6	97.20	89.72	99.69	86.12	92.11	90.22	88.05	**100**
7	35.05	51.66	50.04	45.48	47.46	53.89	67.09	**90.18**
8	36.13	40.16	51.17	44.90	**61.73**	50.32	55.05	40.52
9	**80.69**	54.01	80.43	64.55	66.32	50.24	44.18	74.02
10	43.83	74.73	79.79	87.28	84.00	62.76	77.13	**98.28**
11	58.90	57.35	73.58	80.20	67.56	77.51	68.49	**83.74**
12	44.75	64.85	51.95	**78.20**	58.37	55.27	83.20	38.76
13	73.76	70.32	75.86	70.93	84.82	86.98	59.09	**91.38**
14	88.92	96.23	80.85	99.05	96.67	97.86	97.62	**100**
15	**100**	**100**	**100**	**100**	**100**	**100**	**100**	99.85
OA	67.15	71.93	74.62	78.97	77.80	72.90	75.54	**82.19**
AA	71.74	75.46	77.62	81.13	80.86	77.06	77.90	**85.30**
κ	64.56	69.69	72.58	77.27	76.01	70.72	73.55	**80.79**

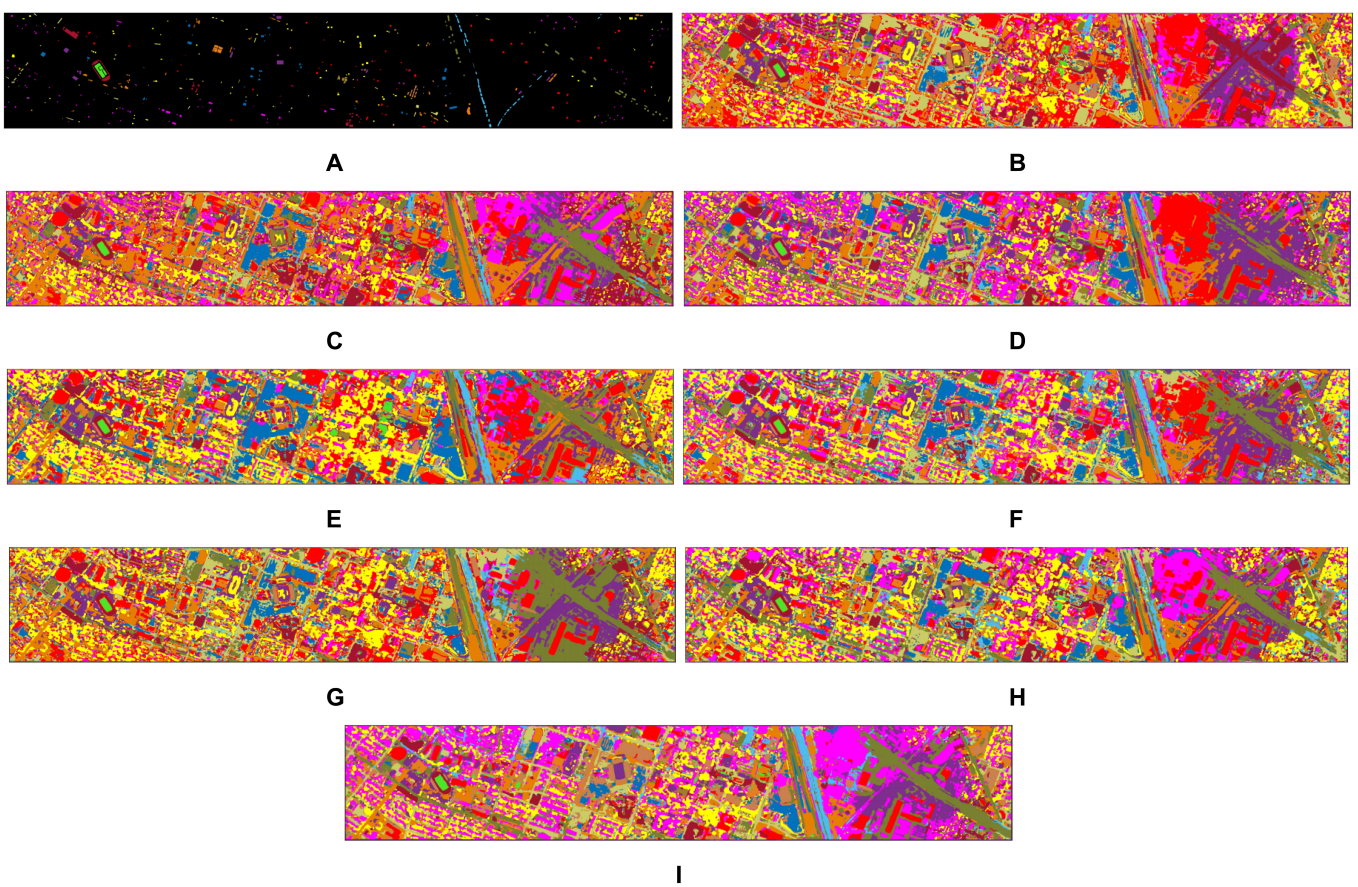

Fig. 13. The classification map on the HOU dataset. (A) Ground truth, (B) RODA (67.15%), (C) 3DLSN (71.93%), (D) SSCL (74.62%), (E) TRCL (78.97%), (F) DMVL (77.80%), (G) A2S2K (72.90%), (H) SSL-AD (75.54%), and (I) DCLN (82.19%).

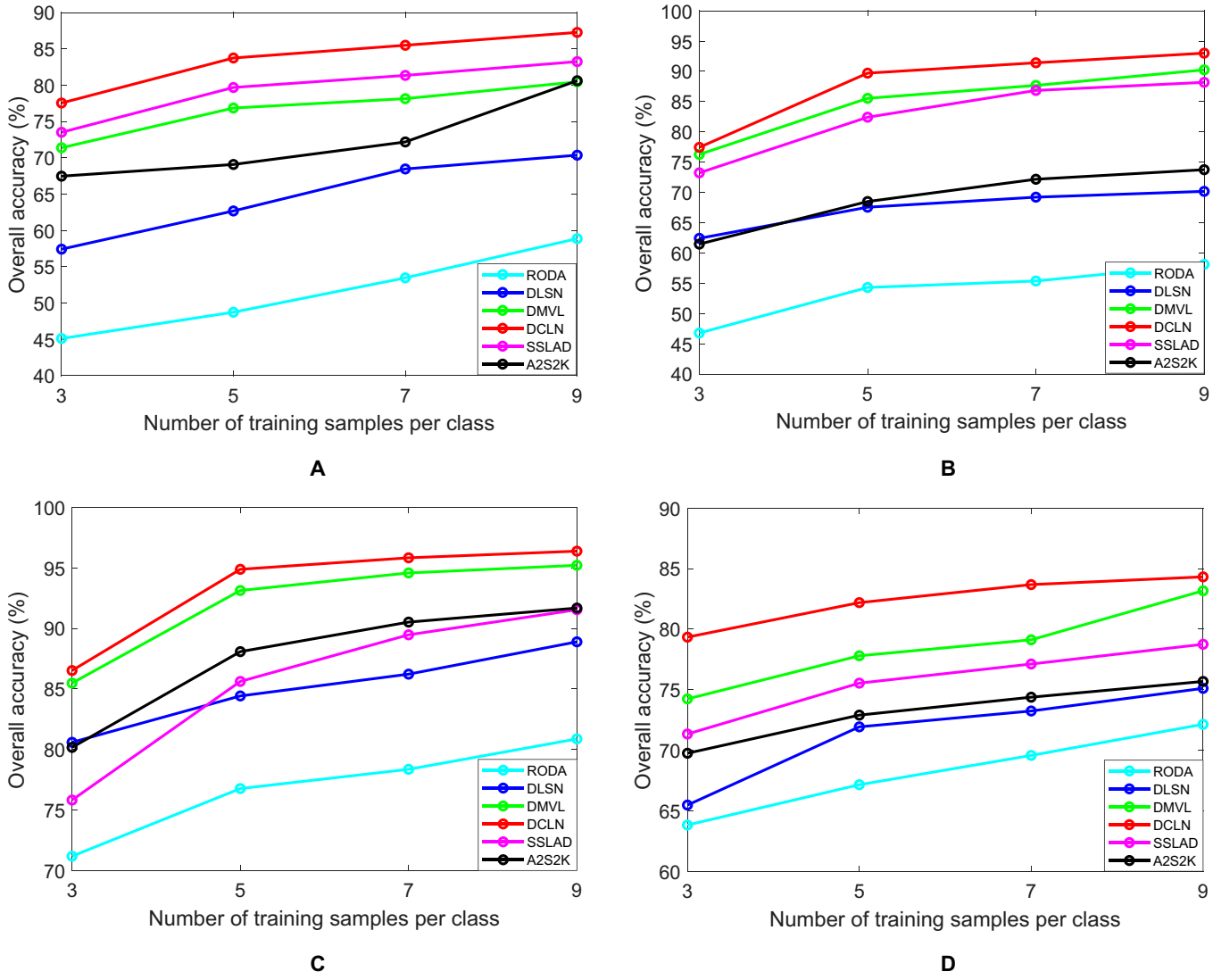

Fig. 14. The OA versus the number of training samples per class on the 4 datasets: (A) IP, (B) UP, (C) SA, and (D) HOU.

We propose a pseudo-labeled sample selection method based on the best versus second-best (BvSB) strategy [27] and design a CMC for each pseudo-labeled sample u^i as:

$$\mathrm{CMC}\left(u^i\right)=\max\left\{p\left(u^i\right)\right\}\cdot\left[\max\left\{p\left(u^i\right)\right\}-\sec\left\{p\left(u^i\right)\right\}\right] \quad (13)$$

where $p(u^i)=[p(u^i,1),\cdots p(u^i,k)]^T$, and $\max(p(u^i)$ and $\sec(p(u^i))$ are the first and the second maximum values in $p(u^i)$.
On the basis of the CMC, it tends to select the sample with the highest class membership probability while having a large difference between the first and second most likely classes. That is, the CMC is more likely to discriminate the confused classes.

By sorting each pseudo-labeled sample according to the CMC, we can select the pseudo-labeled samples with high confidence. Then, we retrain the model with the available limited labeled samples and the selected pseudo-labeled samples.

Results

Datasets

To demonstrate the effectiveness of the proposed method, experiments are conducted on 4 well-known HSI datasets including IP, University of Pavia (UP), Salinas (SA), and Houston (HOU).

IP

This dataset was acquired using Airborne Visible Infrared Imaging Spectrometer (AVIRIS) sensors with a spatial resolution of 20 m. The scene has a spatial size of 145 × 145 pixels and 224 spectral bands, where 200 spectral bands are used in the experiment. The dataset contains 16 classes. The pseudo-color composite image and the corresponding ground-truth map are shown in Fig. 6.

UP

This dataset was acquired using ROSIS-03 sensors with a spatial resolution 1.3 m at the University of Pavia. The scene hasaspatial size of 610 × 340 pixels and 115 spectral bands, where 103

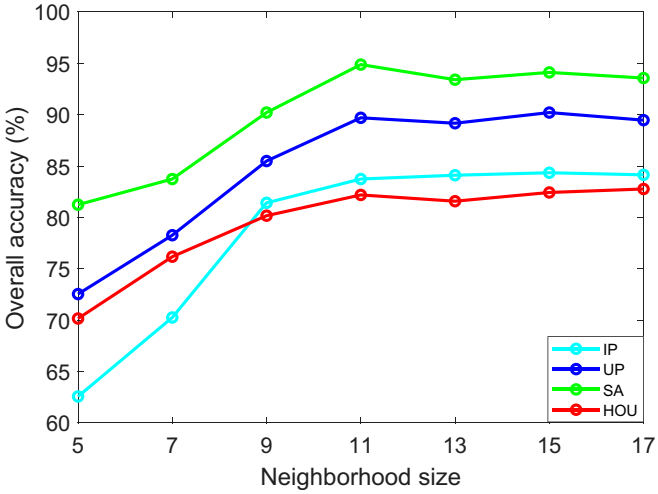

Fig. 15. The OA versus the window size on the 4 datasets.

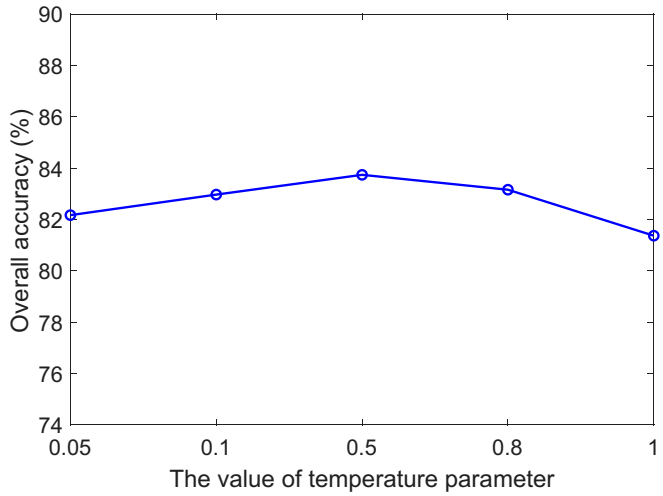

Fig. 16. The OA versus τ on the IP dataset.

spectral bands are used in the experiment. The dataset contains 9 classes. The composite image and ground-truth map are shown in Fig. 7.

SA

This dataset was acquired using AVIRIS sensors with a spatial resolution 3.7 m at the Salinas Valley. The scene has a spatial size of 512 × 217 pixels and 224 spectral bands, where 204 spectral bands are used in the experiment. The dataset contains 16 classes. The pseudo-color composite image and ground-truth map are shown in Fig. 8.

HOU

This dataset was acquired using ITRES CASI-1500 sensors with a spatial resolution 2.5 m at the University of Houston campus. The scene has a spatial size of 349 × 1905 pixels and 144 spectral bands. The dataset contains 15 classes. The pseudo-color composite image and ground-truth map are shown in Fig. 9.

In each classification task, only 5 labeled samples of each class are randomly sampled for training, and the rest of the samples are used for testing. Table 1 shows the number of training and testing samples for different datasets.

Comparison methods and experimental setting

In order to evaluate the performance of the proposed method, 8 DL-based HSI classification algorithms are used for comparison, namely, RODA [34], 3DLSN [28], SSCL [3], transformation-based contrastive learning (TRCL) [39], DMVL [38], attention-based adaptive spectral–spatial kernel ResNet (A2S2K) [41], DFSL [32], and SSL-AD [36]. The RODA is a data augment method, 3DLSN is a lightweight method, and DFSL is a few-shot learning method. SSCL, TRCL, DMVL, and SSL-AD are SSL methods, where the first 3 methods are contrastive learning methods.

The overall accuracy (OA), average accuracy (AA), and κ coefficient are used to evaluate the classification performance of different methods. We randomly run each experiment 10 times and report the averaged results. Because the DFSL used the HOU dataset as the source domain, the article does not compare the classification results of DFSL on the HOU dataset.

In the model training processing of DCLN, a data augmentation strategy is employed to increase the number of samples in the training set. Each training pixel cube is rotated 90 and 270° in spatial domain to augment the training set. All experiments are performed on the 2.99-GHz CPU, GeForce RTX 2080 Ti GPU, and 64-GB memory computer with PyTorch.

Classification results

Table 2 shows the experimental results on the IP dataset. It can be seen that the proposed DCLN method shows the best results in terms of OA, AA, and κ coefficient. From the classification accuracies of different classes, we can see that the proposed DCLN method shows better results than comparison methods on some spectrally similar classes. For example, the second and third classes belong to the subcategories of corn (i.e., "Corn-notill" and "Corn-mintill"), and the 11th and 12th classes of objects belong to the subcategories of soybean (i.e., "Soybean-mintill" and "Soybean-clean"), so the spectral signatures of these subcategories are very similar. In the case of limited training samples, the existing DL methods show poor performance on these 4 classes while our proposed DCLN method provides superior performance. Compared with the recently proposed small-sample contrastive learning methods, i.e., SSCL, TRCL, and DMVL, our DCLN method improves the classification accuracy in the second class by nearly 30%. This demonstrates that our labeled sample-based contrastive learning network improves the feature discriminate ability between spectrally similar samples. Figure 10 visually shows the classification maps of different methods. It can be clearly seen that the proposed DCLN method provides more consistent classification results with little "Salt & Pepper" noise.

Table 3 lists the experimental results on the UP dataset. Compared with other comparison methods, the proposed DCLN method still shows the best results with an OA of 89.70% and has a significant improvement in the classification performance of the sixth categories. In particular, the OA of DCLN is nearly 2.5% higher than TRCL and 4% higher than DMVL. Because of the highly spectral similarity between the class 8 ("Bricks") and class 3 ("Gravel"), many samples in the class 8 are misclassified into class 3, and our method shows relatively poor results. Figure 11 visually demonstrates the classification performance of different algorithms, where DCLN method shows better overall results than other methods.

Table 4 lists the classification results on the SA dataset. As can be seen from the table, in the case of fewer training samples,

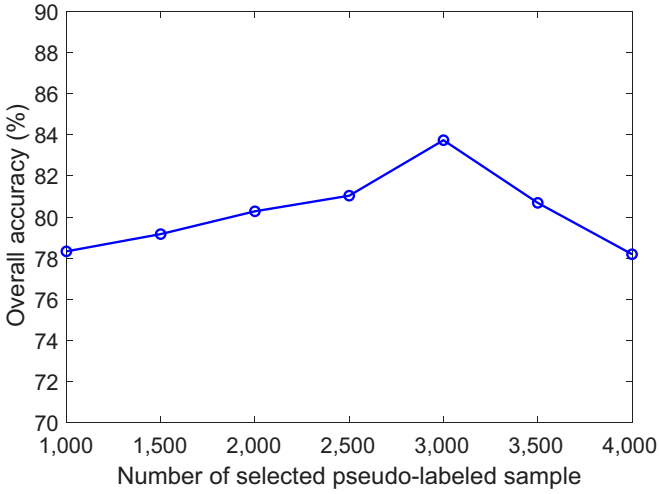

Fig. 17. The OA versus the number of pseudo-labeled samples on the IP dataset.

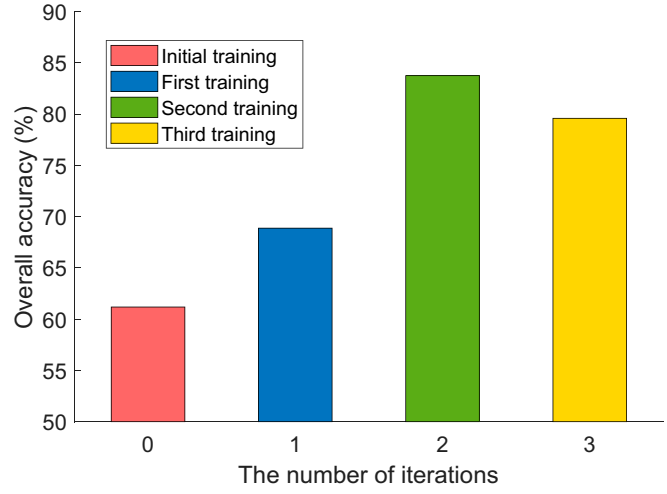

Fig. 18. The OA versus training rounds on the IP dataset.

although some DL algorithms can achieve good performance because of the dense distribution of categories in the SA dataset, the proposed DCLN method still shows the best performance in all 3 indicators. Compared with SSL-AD, DCLN shows better results on 12 classes and improves the OA and κ by more than 9% and 10%, respectively. Figure 12 visually shows the classification maps of different algorithms. It is clear that DCLN yields much better results on the 2 large classes in the upper left corner of the map.

Table 5 lists the classification results on the HOU dataset. The proposed method DCLN has the highest classification accuracy across 9 categories. The improvement is more obvious in the 10th and 11th categories. Figure 13 visually shows the classification maps of different algorithms.

In order to see the effect of the number of labeled training samples, we show the classification OA versus the number of labeled training samples in Fig. 14, where 3, 5, 7, and 9 labeled samples per class are considered. It can be seen that, with the increase of the labeled samples on the IP dataset, the OA of SSL-AD, DMVL, and DCLN methods increases at a similar rate, and the proposed DCLN always maintains an advantage of 4% to 5%, while the OA of A2S2K and DMVL is lower when there are few labeled samples and reaches 80% in the case of 9 labeled samples per class. On the UP dataset, the OA of DCLN is more than 90% when the number of labeled samples per class is larger than 5. On the SA and HOU datasets, DCLN also shows obvious advantages in different numbers of labeled training samples.

Discussion

In this section, we discuss the implementation details of the proposed DCLN method.

The effect of window size

Figure 15 shows the OA versus the window size on 4 datasets, where the window widths of 5, 7, 9, 11, 13, 15, and 17 are considered. It can be seen that the OA increases as the window width increases and keeps stable when the window width is larger than 11. For consistency, the neighborhood window size on 4 datasets are set to be 11×11.

The effect of temperature parameter

An adjustable temperature parameter τ is involved in the calculation of the contrastive loss, and its range is $[-1, 1]$. The role of τ is to adjust the degree of attention to negative samples. The smaller is τ, the more the model pays attention on the negative samples with greater similarity. Figure 16 shows the OA of DCLN on the IP dataset at $\tau = 0.05$, 0.1, 0.5, 0.8, and 1. It is clear that the classification accuracy of the model is the best when the value of τ is 0.5. In the experiments, $\tau = 0.5$ is used for 4 datasets.

The effect of the number of pseudo-labeled samples

The paper generates pseudo-label for each unlabeled sample and then selects part of pseudo-labeled samples with high confidence to retrain the model. We retrain the model twice. In the first training stage of the model, 1,000 pseudo-labeled samples with high confidence are selected. In this stage, the selected samples are generally easily classified samples, and they are almost correctly classified and can be used to improve the model performance. In the second training stage, different numbers of pseudo-labeled samples have a certain impact on the model. If too many samples are selected, then some pseudo-labeled samples with low confidence (e.g., difficult to classify samples) will degrade the performance of the model. If too few samples are selected, then the useful information that the model obtained is insufficient and the classification accuracy will not be satisfactory. We investigate the effect of the number of pseudo-labeled samples in the second stage on IP dataset in Fig. 17, where different numbers (i.e., 1,000, 1,500, . . ., 4,000) of pseudo-labeled samples with the highest confidence are investigated. As can be seen from the figure, when the number of pseudo-labeled samples is less than 3,000, the classification accuracy increases with more pseudo-labeled samples and reaches the maximum at 3,000. However, when the number further increases, the OA has dropped significantly. Therefore, we select 3,000 pseudo-labeled samples with high confidence in the second time.

The effect of training rounds

The model uses 5 labeled samples per class to train the initial model and then adds 1,000 and 3,000 pseudo-labeled samples to the training set to retrain the model twice. We investigate

Table 6. Execution times of training and feature extraction procedures on 3 datasets.

Indian Pines	DFSL	DMVL	SSL-AD	DCLN
Training (min)	106.2	141.23	480	**10.15**
Feature extraction (s)	11.14	27.31	13.35	**2.11**
Pavia University				
	DFSL	DMVL	SSL-AD	DCLN
Training (min)	106.2	197.71	480	**32.34**
Feature extraction (s)	40.82	50.09	21.17	**2.82**
Salinas				
	DFSL	DMVL	SSL-AD	DCLN
Training (min)	106.2	243.17	480	**60.15**
Feature extraction (s)	52.31	77.74	34.32	**3.38**

the classification OA of the model in different training rounds on the IP dataset in Fig. 18. If there are only 5 labeled samples per class for training, then the initial model performance is very poor with an OA of 61.18%. By adding 1,000 pseudo-labeled samples into the training set, the OA is improved to 68.87%. At this time, the model's performance is also unsatisfactory because the number of labeled and pseudo-labeled samples is still insufficient for network training. By retraining the network again with adding 3,000 pseudo-labeled samples, the OA achieves 83.74%. When we train the model in the third time by continually adding 1,000 pseudo-labeled samples, the performance of the model degrades because this will introduce some misclassified samples into the training set, which affects the training of the model. Therefore, we train the model for 2 rounds.

Computational complexity

We compare the training and feature extraction times of the proposed method with DFSL, DMVL, and SSL-AD methods. As shown in Table 6, the proposed DCLN method is much faster than the other 3 methods on the IP, UP, and SA datasets. The main network of the proposed DCLN algorithm consists of only 4 convolutional layers, so our algorithm has fewer model parameters and faster feature extraction time.

Conclusion

This paper has proposed a DCLN method for the classification of HSIs with limited labeled samples. In the proposed DCLN method, the available limited labeled samples are directly used to construct contrastive groups, which facilitate the feature contrastive learning because the label of samples can indicate the similarity or dissimilarity of samples. After contrastive learning between 2 groups in a deep feature extraction framework, spatial–spectral features of HSI are learned and used to form a SSMD for the generation of pseudo-labeled samples. Finally, a CMC is designed to select high-confidence pseudo-labeled samples to retrain the network. Experimental results on 4 public HSI datasets demonstrated that the proposed method outperforms the existing popular HSI classification methods in the case of limited labeled samples.

Acknowledgments

Funding: This work was supported in part by the National Natural Science Foundation of China under grant nos. 42171351, 42122009, and 41971296; by the Natural Science Foundation of Hubei Province under grant 2021CFA087; and by the Public Projects of Ningbo City under grant 2021S089. **Author contributions:** Q.L., J.P., G.Z., W.S., and Q.D. conceived the idea and designed the experiments. Q.L. and J.P. implemented the experiments. Q.L., J.P., and Q.D. wrote the manuscript. All authors read and approved the final manuscript. **Competing interests:** The authors declare that there is no conflict of interest regarding the publication of this article.

Data Availability

The IP, UP, and SA datasets can be obtained from http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes. The HOU dataset is provided by the 2013 IEEE GRSS Data Fusion Contest.

References

1. Bioucas-Dias JM, Plaza A, Camps-Valls G, Scheunders P, Nasrabadi NM, Chanussot J. Hyperspectral remote sensing data analysis and future challenges. *IEEE Geosci Remote Sens Mag*. 2013;1(2):6–36.
2. Plaza A, Benediktsson JA, Boardman JW, Brazile J, Bruzzone L, Camps-Valls G, Chanussot J, Fauvel M, Gamba P, Gualtieri A, et al. Recent advances in techniques for hyperspectral image processing. *Remote Sens Environ*. 2009;113:S110–S122.
3. Hou S, Shi H, Cao X, Zhang X, Jiao L. Hyperspectral imagery classification based on contrastive learning. *IEEE Trans Geosci Remote Sens*. 2022;60:5521213.
4. Peng J, Sun W, Li H-C, Li W, Meng X, Ge C, Du Q. Low-rank and sparse representation for hyperspectral image processing: A review. *IEEE Geosci Remote Sens Mag*. 2022;10(1):10–43.
5. Camps-Valls G, Bruzzone L. Kernel-based methods for hyperspectral image classification. *IEEE Trans Geosci Remote Sens*. 2005;43(6):1351–1362.
6. Li W, Tramel EW, Prasad S, Fowler JE. Nearest regularized subspace for hyperspectral classification. *IEEE Trans Geosci Remote Sens*. 2014;52(1):477–489.
7. Camps-Valls G, Gomez-Chova L, Muñoz Maŕe J, Vila-Francés J, Calpe-Maravilla J. Composite kernels for hyperspectral image classification. *IEEE Geosci Remote Sens Lett*. 2006;3(1):93–97.
8. Peng J, Sun W, Du Q. Self-paced joint sparse representation for the classification of hyperspectral images. *IEEE Trans Geosci Remote Sens*. 2019;57(2):1183–1194.
9. Peng J, Li L, Tang YY. Maximum likelihood estimation based joint sparse representation for the classification of hyperspectral remote sensing images. *IEEE Trans Neural Netw Learn Syst*. 2019;30(6):1790–1802.
10. Benediktsson JA, Palmason JA, Sveinsson JR. Classification of hyperspectral data from urban areas based on extended morphological profiles. *IEEE Trans Geosci Remote Sens*. 2005;43(3):480–491.

11. Kang X, Li S, Benediktsson JA. Spectral–spatial hyperspectral image classification with edge-preserving filtering. *IEEE Trans Geosci Remote Sens*. 2014;52(5):2666–2677.
12. Shen L, Jia S. Three-dimensional Gabor wavelets for pixel-based hyperspectral imagery classification. *IEEE Trans Geosci Remote Sens*. 2011;49(12):5039–5046.
13. Li W, Chen C, Su H, Du Q. Local binary patterns and extreme learning machine for hyperspectral imagery classification. *IEEE Trans Geosci Remote Sens*. 2015;53(7):3681–3693.
14. Wang Q, Meng Z, Li X. Locality adaptive discriminant analysis for spectral–spatial classification of hyperspectral images. *IEEE Geosci Remote Sens Lett*. 2017;14(11):2077–2081.
15. Li S, Song W, Fang L, Chen Y, Ghamisi P, Benediktsson JA. Deep learning for hyperspectral image classification: An overview. *IEEE Trans Geosci Remote Sens*. 2019;57(9): 6690–6709.
16. Chen Y, Lin Z, Zhao X, Wang G, Gu Y. Deep learning-based classification of hyperspectral data. *IEEE J Sel Top Appl Earth Obs Remote Sens*. 2014;7(6):2094–2107.
17. Chen Y, Jiang H, Li C, Jia X, Ghamisi P. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Trans Geosci Remote Sens*. 2016;54(10):6232–6251.
18. Cheng G, Yang C, Yao X, Guo L, Han J. When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNs. *IEEE Trans Geosci Remote Sens*. 2018;56(5):2811–2821.
19. Mei S, Ji J, Bi Q, Hou J, Du Q, Li W. Integrating spectral and spatial information into deep convolutional neural networks for hyperspectral classification. Paper presented at: IGARSS 2016. Proceedings of the IEEE International Geoscience and Remote Sensing Symposium; 2016 Jul 10–15; Beijing, China; p. 5067–5070.
20. Yang J, Zhao Y, Chan JC-W, Yi C. Hyperspectral image classification using two-channel deep convolutional neural network. Paper presented at: IGARSS 2016. Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS); 2016 Jul 10–15; Beijing, China; p. 5079–5082.
21. Zhang H, Li Y, Zhang Y, Shen Q. Spectral-spatial classification of hyperspectral imagery using a dual-channel convolutional neural network. *Remote Sens Lett*. 2017;8(5):438–447.
22. Li Y, Zhang H, Shen Q. Spectral–spatial classification of hyperspectral imagery with 3D convolutional neural network. *Remote Sens*. 2017;9(1):67.
23. Paoletti M, Haut J, Plaza J, Plaza A. A new deep convolutional neural network for fast hyperspectral image classification. *ISPRS J Photogramm Remote Sens*. 2018;145:120–147.
24. Roy SK, Krishna G, Dubey SR, Chaudhuri BB. HybridSN: Exploring 3-D–2-D CNN feature hierarchy for hyperspectral image classification. *IEEE Geosci Remote Sens Lett*. 2020;17(2):277–281.
25. Zhong Z, Li J, Luo Z, Chapman M. Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework. *IEEE Trans Geosci Remote Sens*. 2018;56(2):847–858.
26. Li W, Chen C, Zhang M, Li H, Du Q. Data augmentation for hyperspectral image classification with deep CNN. *IEEE Geosci Remote Sens Lett*. 2018;16(4):593–597.
27. Gao H, Yang Y, Li C, Gao L, Zhang B. Multiscale residual network with mixed depthwise convolution for hyperspectral image classification. *IEEE Trans Geosci Remote Sens*. 2021;59(4):3396–3408.
28. Jiang S, Jia S. A 3D lightweight Siamese network for hyperspectral image classification with limited samples. Paper presented at: ICCPR 2021. Proceedings of the 10th International Conference on Computing and Pattern Recognition; 2021 Oct 15–17; p. 142–148.
29. Jia S, Lin Z, Xu M, Huang Q, Zhou J, Jia X, Li Q. A lightweight convolutional neural network for hyperspectral image classification. *IEEE Trans Geosci Remote Sens*. 2021;59(5):4150–4163.
30. Cui B, Dong X-M, Zhan Q, Peng J, Sun W. Litedepthwisenet: A lightweight network for hyperspectral image classification. *IEEE Trans Geosci Remote Sens*. 2022;60:5502915.
31. Li X, Sun Z, Xue J-H, Ma Z. A concise review of recent few-shot meta-learning methods. *Neurocomputing*. 2021;456:463–468.
32. Liu B, Yu X, Yu A, Zhang P, Wan G, Wang R. Deep few-shot learning for hyperspectral image classification. *IEEE Trans Geosci Remote Sens*. 2019;57(4):2290–2304.
33. Li Z, Liu M, Chen Y, Xu Y, Li W, Du Q. Deep cross-domain few-shot learning for hyperspectral image classification. *IEEE Trans Geosci Remote Sens*. 2022;60:5501618.
34. Haut JM, Paoletti ME, Plaza J, Plaza A, Li J. Hyperspectral image classification using random occlusion data augmentation. *IEEE Geosci Remote Sens Lett*. 2019;16(11):1751–1755.
35. Li X, Ding M, Pižurica A. Group convolutional neural networks for hyperspectral image classification. Paper presented at: ICIP 2019. Proceedings of the IEEE International Conference on Image Processing (ICIP); 2019 Sep 22–25; Taipei, Taiwan; p. 639–643.
36. Yue J, Fang L, Rahmani H, Ghamisi P. Self-supervised learning with adaptive distillation for hyperspectral image classification. *IEEE Trans Geosci Remote Sens*. 2022;60:5501813.
37. Chen T, Kornblith S, Norouzi M, Hinton G. A simple framework for contrastive learning of visual representations. Paper presented at: ICML 2020. Proceedings of the 37th International Conference on Machine Learning; 2020 Jul 13–18; p. 1597–1607.
38. Liu B, Yu A, Yu X, Wang R, Gao K, Guo W. Deep multiview learning for hyperspectral image classification. *IEEE Trans Geosci Remote Sens*. 2021;59(9):7758–7772.
39. Zhao L, Luo W, Liao Q, Chen S, Wu J. Hyperspectral image classification with contrastive self-supervised learning under limited labeled samples. *IEEE Geosci Remote Sens Lett*. 2022;19:6008205.
40. Khosla P, Teterwak P, Wang C, Sarna A, Tian Y, Isola P, Maschinot A, Liu C, Krishnan D. Supervised contrastive learning. *Adv Neural Inf Proces Syst*. 2020;33:18661–18673.
41. Roy SK, Manna S, Song T, Bruzzone L. Attention-based adaptive spectral–spatial kernel ResNet for hyperspectral image classification. *IEEE Trans Geosci Remote Sens*. 2021;59(9):7831–7843.