Hyperspectral Image Classification Based on Unsupervised Regularization

Jian Ji, Shuiqiao Liu, Fangrong Zhang, Xianfu Liao, Shuzhen Wang, and Junru Liao

***Abstract*—Due to the powerful feature expression ability of deep learning and its end-to-end nonlinear mapping relationship, deep-learning-based methods have become the mainstream method for hyperspectral image (HSI) classification tasks. However, the accuracy of deep learning methods greatly depends on the use of a large number of labeled samples to train the model. Also, HSIs have few labeled samples and unbalanced categories, which make the depth model prone to overfittingand seriously affect the classification accuracy. Therefore, how to alleviate the overfitting phenomenon caused by small samples in the classification problem based on deep learning is still a problem that needs to be solved. Considering that it is relatively easier to obtain a large number of unlabeled samples in the field of remote sensing, making full use of the unsupervised information learned from unlabeled data can regularize the supervised classification model, which can effectively alleviate the overfitting phenomenon caused by the small samples problem. In the supervised training process, unsupervised information from the overall distribution of the sample is introduced to guide the regularization of the model, so as to realize the effective classification of the data in the case of a small number of labeled samples. Experimental results demonstrate the effectiveness of the proposed method in terms of HSI classification with few training samples.**

***Index Terms*—Few samples, hyperspectral image (HSI) classification, model regular, unsupervised information.**

I. Introduction

HYPERSPECTRAL image (HSI) has a strong pixel representation ability. Hyperspectral imaging is based on the spectral reflectance of ground objects, so it has strong ground penetration and its resolution is not easily affected by color shading. It has unique advantages in military reconnaissance, agricultural observation, geological prospecting, and transportation planning [1]. As a prerequisite for the practical application of HSIs, the classification of HSIs is of great significance. HSI classification refers to dividing each pixel into a specific feature category according to the spectral curve provided by each pixel. There are many research works on HSI classification, but because the ground annotation of remote sensing images is expensive, HSI classification lacks enough training samples. Also, the high dimension and spectral redundancy of HSIs lead to their high data volume characteristics. Small samples are associated with high-dimensional characteristics, which is easy to cause a "dimension disaster" [2], that is, the dimensionality is too high and the samples are too few, so that the accuracy of the classification task is reduced due to the overfitting of the model [16].

In recent years, a lot of algorithms have emerged for HSI classification, including traditional algorithms based on statistical theory and algorithms based on deep learning. According to whether the classification algorithm is pixel-by-pixel or uses the semantic information of the pixels around the pixel, the existing HSI classification can be divided into two types: 1) pixel-level classification and 2) super-pixel-level classification [4].

The traditional methods of pixel-level classification include the following: 1) Linear classifiers: such as logistic regression [5] and Gaussian maximum likelihood classification [6], respectively, assuming that the sample obeys the Bernoulli distribution and the normal distribution, and constructing the likelihood function to find the decision. The boundary then classifies each pixel. 2) Distance-based classifiers: such as K-nearest neighbor classification [25], minimum distance classification [8], and support vector machine (SVM) [9]. The main idea is to use the distance between the test sample and each class as a decision. The model determines the test sample as the closest class to it, where SVM uses the distance of the sample in the feature space after kernel function mapping. Except for SVM, these methods cannot solve the "curse of dimensionality" of HSIs. But this problem can be alleviated by dimensionality reduction or band selection. That is, first perform feature extraction on the input sample and then perform the classification operation. Feature extraction methods include feature extraction method based on binary discrete wavelet transform [10] and fast dimensionality reduction method based on dynamic programming [11]. In addition to the dimensionality reduction of the data [27], another method for high-dimensional problems is band selection, such as independent component analysis for band selection [28]; the bands containing more information are selected by evaluating the average weight coefficient of each band, using an adaptive band weight measurement method based on information entropy [14]. These methods of deleting redundant bands reduce the computational complexity of HSI classification and ease the high-dimensional problem to a certain extent [26].

Manuscript received 17 July 2022; revised 22 September 2022, 5 November 2022, 11 December 2022, and 8 January 2023; accepted 27 January 2023. Date of publication 2 February 2023; date of current version 15 February 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62273268 and in part by the Key Research and Development Program of Shaanxi Province under Grant 2022GY-059. *(Corresponding author: Jian Ji.)*

Jian Ji, Fangrong Zhang, Xianfu Liao, Shuzhen Wang, and Junru Liao are with the College of Computer Science and Technology, Xidian University, Xi'an 710126, China (e-mail: jji@xidian.edu.cn; zzzfangrong@163.com; 2226493132@qq.com; shuzhenwang@xidian.edu.cn; 623238024@qq.com).

Shuiqiao Liu is with the AVIC Xi'an Aeronautical Computing Technology Research Institute, Xi'an 710076, China (e-mail: 892358507@qq.com).

Digital Object Identifier 10.1109/JSTARS.2023.3241662

The task of HSI classification based on deep learning mainly focuses on proper feature representation and effective classifier design. For example, the deep belief network (DBN) [32] uses a network of restricted Boltzmann machines, which learns layer by layer to extract robust nonlinear features in HSIs. However, DBN adopts a fully connected (FC) structure [17], which leads to too many parameters in the model, and the effect is not good in the case of small samples. Different from FC networks, convolutional neural networks (CNN) [40] use partial connections to share weights, thereby reducing the number of parameters. Since the local connection of CNN is suitable for dealing with the situation where there are few available training samples for HSIs [19], a series of classification methods based on CNN and its variants have appeared in HSI classification, such as AlexNet [15] designed deeper on the basis of the original CNN structure, so it has better feature representation capabilities but it brings more parameters. In addition, there are GoogleNet [13], VGGNet [3], DensNet [7], recurrent neural network, etc. These networks make the original CNN wider or use convolutional kernels of the same size, fixed pool size, and direct connection structure to reduce the difficulty of training, compared with the general deep network to achieve better scores classification results [24].

Considering that it is relatively easier to obtain a large number of unlabeled samples in the field of remote sensing, making full use of the unsupervised information learned from unlabeled data can regularize the supervised classification model, which can effectively alleviate the phenomenon of small samples and the overfitting problem. Based on this clustering assumption that adjacent samples have similar output values on the same manifold structure, this chapter designs an HSI classification framework that shares unsupervised information, that is, introduces unsupervised training from the overall distribution of samples in the supervised training process. Supervise the information and guide the regularization of the model, so as to effectively classify the data in the case of a small number of label samples.

The main work of this article is embodied in the following three aspects.

1) A shared feature extraction module (SFEM) is designed based on the Kullback–Leibler (KL) sparse stack autoencoder (AE) structure, which is used to extract both the labeled data and the unlabeled data in the feature extraction stage to obtain their consistency information. That is, the structural information of the sample is put into the process of supervised learning as prior knowledge to provide regularity for the learning process.
2) Use the supervised learning and unsupervised clustering processes to classify and cluster the two kinds of data in the same dataset. The information obtained from unsupervised learning that causes classification loss due to the introduction of unsupervised clustering is introduced into the supervised learning process. The process provides information on similarities and differences between classes, and more information can be obtained from the supervised learning part.
3) Test the effectiveness of this scheme in the mainstream dataset of HSIs. In particular, when dividing samples, focusing on the categories with a small number of tags, we can see the superior performance of this method in the case of small samples.

II. Related Work

In HSI classification, traditional methods require fewer parameters and low computational complexity. However, because HSIs collected under natural conditions do not generally have a certain distribution law in the sense of mathematics and statistics, it is based on the distribution assumption or the traditional method of Vapnik–Chervonenkis (VC) dimensional theory is limited. Deep-learning-based methods can extract high-level features of data through multiple hidden deep networks, but deep-learning-based methods require more labeled data for training. The classification effect depends on the number of effective training samples, so it appeared a lot of labeled training samples. The lack of image and the imbalance of HSI categories have become important factors limiting its development.

The method based on deep learning merges the spatial features into the classifier, extracts the spatial features and spectral features separately during feature extraction, and then uses feature fusion technology to perform joint classification. For example, using 2-D CNN to perform feature extraction on the machine neighborhood of hyperspectral pixels containing spatial information while performing principal component analysis (PCA) operations on the spectral dimension can extract discriminative spatial features while reducing the computational cost. Based on this, Liang et al. [12] introduced sparse representation technology to encode deep spatial features extracted by CNN into low-dimensional sparse features to improve feature representation capabilities. Long et al. [18] used the trained fully convolutional networks 8 (FCN8) to explore deep multiscale spatial structure information and used a weighted fusion mechanism to fuse the original spectral features and deep multiscale spatial features, and finally input the fused features into the classifier for execution classification prediction.

Recently, model regularization (MR) is a method to effectively alleviate the model overfitting caused by the small sample problem. In deep learning, too few training samples and more parameters in the model will cause the model to change from the limited training data. The model learned in the medium lacks generalization ability, that is, the phenomenon of overfitting. In response to this problem, there are several strategies in HSI processing: transfer learning (TL) [22], active learning (AL), and model optimization [30]. TL is a method that learns useful information from auxiliary data and introduces it into the target dataset to effectively reduce the data dependence of the algorithm. Deng et al. [48] used the initial values of deep network parameters trained on other remote sensing datasets to initialize the 2-D CNN for classification. Compared with random initialization, the TL algorithm converged faster after parameter migration [29]. AL is based on the selection of training samples, adding unlabeled data samples as new training samples to the training dataset, thereby adding labeled training samples. Ma et al. [20] combined AL with iterative training sampling, expanded the multidimensional dataset by

iteratively incorporating other spatial classification information into the unlabeled data samples enhanced by AL, and updated the current training samples in a single iteration. While further improving the accuracy of classification, it reduces the inconsistency of classification. Finally, it is a method based on MR. MR refers to adding some restriction rules to the target function that needs to be trained, reducing the parameter space, and then constraining the solution space and minimizing classification errors. In a hyperspectral dataset, different collected light, weather, and shading will cause the spectral reflectance of the same object to deviate. The efficiency of TL relies on the consistency of the dataset, so the use of TL to solve the small sample problem in HSI classification has the problem of auxiliary dataset selection. The method of AL to expand training samples is to select data that are helpful for classification from unlabeled samples, query human experts, and obtain the label of the sample. Additional information is required, which is often not available in actual classification applications. Therefore, choosing a simpler and more feasible MR can alleviate the overfitting problem caused by small samples by introducing prior knowledge into the sample and reducing the loss of model structure.

Currently, making use of the unsupervised information contained in the dataset can improve the classification performance. The machine learning algorithm can be divided into two categories according to whether the training process uses labeled training samples: 1) supervised learning and 2) unsupervised learning. First, supervised learning establishes training sets based on samples in different categories and then makes decisions based on training parameters. The unsupervised training process does not require labeled samples, and the main purpose is to extract useful features from a large amount of unlabeled data. Because of the high cost of remote sensing data processing and labeling, the number of unlabeled samples in hyperspectral data is much larger than that of labeled samples [31]. To deal with this problem, more and more research works are devoted to designing an unsupervised deep learning framework for HSI data to realize an encoder–decoder that can learn without using label information, and at the same time through migrate the trained network and fine-tune the labeled dataset to improve classification performance [34]. The advantage of the unsupervised algorithm is that it does not need to have label data to obtain its own distribution information in the sample, but it also has the disadvantage that the classification is not accurate enough and the category needs to be determined manually. Therefore, we consider using a clustering algorithm to initially extract the difference information of the sample in the feature space and introduce it as a regular term into the supervised classification process to improve the classification accuracy.

HSI has a large amount of data and many feature channels. Liu et al. used the Ghost module to reduce the complexity of the model, and combined with the extended morphological profile (EMP) features, propose an HSI classification method based on EMP features and Ghost module (GhostEMP). GhostEMP can improve the efficiency of operation [33]. Also, Shen et al. proposed a method named GLSESP to improve the performance of the supervised classification. They used the global spatial and local spectral similarity to extend the labeled sample size. Also, in order to alleviate band redundancy, they extended subspace projection, which projects the original image to a lower-dimensional subspace. GLSESP is also very practical and effective in HSI classification [23]. Recently, CNNs have been widely used for HSI classification due to their detailed representation of features. However, the current CNN-based HSI classification methods mainly follow a patch-based learning framework. These methods not only limit the use of global information but also require a high computational cost. So, Xu et al. used an image-based global learning framework for HSI classification. They proposed a dual-channel convolutional network (DCCN) for HSI classification to maximize the exploitation of the global and multiscale information of HSI [35].

Also, CNNs have emerged as a popular choice for HSI analysis now. However, the performances of traditional CNN-based patchwise classification methods are limited by insufficient training samples, and the evaluation strategies tend to provide overoptimistic results due to training-test information leakage. To address these concerns, Liang Zou et al. proposed a novel spectral–spatial 3-D fully convolutional network to jointly explore the spectral–spatial information and the semantic information. It takes small patches of original HSI as inputs and produces the corresponding sized outputs, which enhances the utilization rate of the scarce labeled images and boosts the classification accuracy [36].

III. Proposed Method

The core idea of this article is to introduce the unsupervised information contained in the whole sample consisting of a small amount of labeled data and a large amount of unlabeled data as a regularization constraint in the training process of supervised HSI classification and use the classification loss to backpropagate. The supervised classification model can learn the unsupervised information of the full set of samples in addition to the information contained in the labeled samples and alleviate the overfitting caused by small samples. For a given hyperspectral dataset $x \in i^{w\times h\times d}$, where $w \times h$ represents the width and height of the image, and d represents the number of spectral bands. There are a total of n samples in the input dataset, of which l samples belong to the labeled dataset $X_L = x_1, x_2,..., x_l, x_i \in X$. The corresponding label is $Y_L = y_1, y_2, \ldots, y_l, y_i \in C$, C is the number of data categories. The $n - l$ samples other than the labeled samples constitute the unlabeled dataset, $X_U = x_{l+1}, x_{l+2}, \ldots, x_n, x_i \in X$. The purpose of the classification task is to train a classifier by X_L and use this classifier to correctly classify X_U.

A. Unsupervised Pretraining Based on Stacked AE

In order to make better use of the information contained in unlabeled samples, this article first designs a stacked AE (SAE), which uses backpropagation to perform unsupervised learning on labeled samples and unlabeled samples and perform feature extraction together. Then, input the features into the corresponding classifier for end-to-end training. The purpose of the AE is to learn an effective representation of the input data. AE uses a fictitious three-layer network, assuming that the

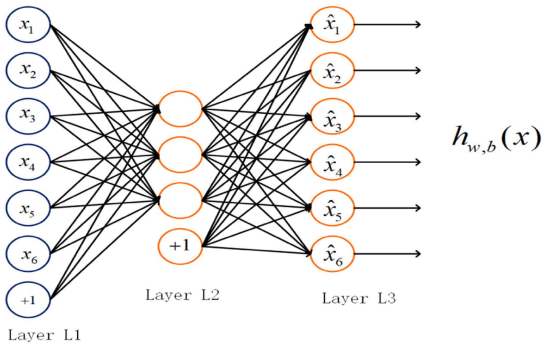

Fig. 1. Self-encoder structure.

original data are also the target output, and uses the loss of the real output and the target output to construct a supervision error for training. After the training is completed, the output layer is removed to obtain the feature expression of the input data. The structure of AE is shown in Fig. 1.

The training process of the AE is expressed as follows:

$$y = f_\theta(x) = h\left(Wx + b\right) \tag{1}$$

$$\hat{x} = g_{\theta'}(y) = h\left(W'y + b'\right) \tag{2}$$

$$L(x,\ \hat{x}) = L(x,\ g(f(x))). \tag{3}$$

In order to learn more meaningful expressions and prevent AE from becoming a linear encoder and learn identity expressions, this chapter adds regular constraints to the hidden Layer L2 and constructs AE as a sparse AE. Specifically, by adding a sparsity penalty during training to reconstruct the error as (3). In order to limit the sparsity of AE hidden layer neurons, use KL divergence to constrain the average activation value of most of the hidden layer neurons: Specify a sparsity parameter that represents the average of hidden neurons on the training set activity, use KL divergence to measure the relative entropy of the expected activation and the actual activation of the actual neuron, and then add it as a regular term to the objective function. Finally, the loss function of AE can be expressed as

$$\begin{aligned} L &= L(x, g(f(x)) + \beta \sum_{j=1}^{h} KL(\rho||\hat{\rho}) \\ &= L(x, g(f(x)) + \beta \sum_{j=1}^{h} \left(\rho \log \frac{\rho}{\hat{\rho}_j} + (1-\rho)\log \frac{1-\rho}{1-\hat{\rho}_j}\right) \end{aligned} \tag{4}$$

$$\hat{\rho} = \frac{1}{m}\sum_{i=1}^{m}(a_j(x_i)). \tag{5}$$

m represents the dimension of the input data, $a_j$represents the activation value on the hidden layer neurons j, and β is the weight of the sparsity penalty item. Since KL divergence is a measure of the asymmetry of the difference between two probability distributions, the introduction of the sparse regular AE of KL divergence in this chapter can better learn the similarity information of samples of X_L and X_U as the same kind. In deep learning, the deep network can learn multiple expressions of the original data layer by layer. Based on the same principle, this article uses an SAE structure to stack three AEs, the input of each AE is based on the output of the previous AE, learning a more abstract representation of features. The structure of the stack AE is shown in Fig. 2.

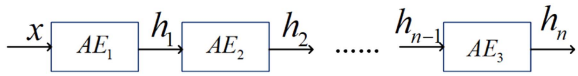

Fig. 2. SAE structure.

In order to implement an SAE for shared feature extraction and layer-by-layer unsupervised pretraining, a labeled dataset and an unlabeled dataset are used to train the AEs that have undergone KL divergence sparse regularization step by step. For the input vector x, the high-level representation of x is first obtained, and then, the second-order feature representation of the original data x is obtained in the input. The last self-encoder is input, and after the processor, a softmax layer is added. The output of this softmax layer serves as the input to the next layer, and the high-level feature representation of x is output after processing. After unsupervised and training, the softmax layer and the last AE are canceled, and the final SAE with a three-layer structure is obtained. The pretrained SAE fits the structure of the training data to a large extent. This SAE serves as the shared feature extractor of the labeled data and unsupervised data of the image, which can well obtain the unsupervised data contained in the whole sample. The supervision information reflects the relationship between sample similarity and corresponding label similarity.

B. Few Shot HSI Classification Framework With Shared Unsupervised Information

In order to alleviate the overfitting problem caused by the small number of labeled samples, this article proposes a training framework (Shared Unsupervised Information Classification Framework, SUICF) that shares the unsupervised information contained in all samples into the supervised classification process, using the loss function. The way to guide the supervised classification model is as shown in Fig. 3.

The model is completed in two steps. On the one hand, SAE uses backpropagation to perform feature extraction on raw data through unsupervised learning, and the extracted features of labeled data and unlabeled data are input into the supervised feature extractor and unsupervised feature extractor respectively. On the other hand, all data are extracted. This article uses the *K*-means clustering algorithm, the parameter k is set to the number of categories of the input data, after clustering, all the data get the pseudolabel of their own category.

Specifically, based on KL sparse stack automatic encoder structure, an SFEM is designed to extract tagged data and unlabeled data in the feature extraction phase to obtain their consistency information. The supervised learning and unsupervised clustering processes are used to classify and cluster two kinds of data in the same dataset. Because unsupervised clustering is

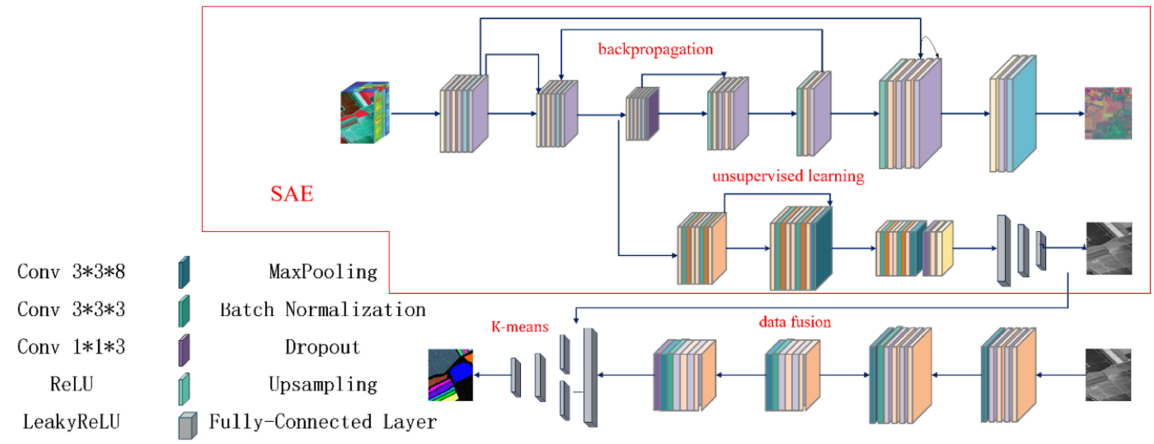

Fig. 3. Few shot HSI classification framework for sharing unsupervised information. The upper part is SAE: Layer-by-layer pretrained network based on backpropagation algorithm to update parameters, and the lower part is *K*-means: to get pseudolabels from the fused data.

introduced, the information obtained from unsupervised learning is introduced into the supervised learning process in the way of classification loss. The interclass similarity between the data captured by the *K*-means algorithm is trained by CNN using the pseudotags generated by clustering, and the unsupervised information input is effectively strengthened in a supervised way.

The pretrained stack self-encoder fits the structure of training data to a large extent. As the shared feature extractor of image-tagged data X_L and unsupervised data X_U, this SAE can well obtain the unsupervised information contained in the whole sample, that is, the relationship between sample similarity and corresponding tag similarity.

The interclass similarity between the data captured by the *K*-means algorithm is trained by the CNN using the pseudolabels generated by the clustering, and the unsupervised information is effectively strengthened in a supervised way. The feature extraction module based on the pretraining of the shared SAE effectively shares the unsupervised information in the data, makes the unsupervised information flow to the supervision task, and provides an effective regularity for the network. In this framework, the input data of the three branches are calculated through three softmax layers to calculate the probability that each pixel belongs to a certain category. Cross-entropy is calculated for supervised data supervision features and their corresponding pseudo labels as regular term J_1. Similarly, J_2 is the cross-entropy for the unsupervised features of the supervised data and their corresponding pseudolabels as the regular term, and J_3 is the cross-entropy of unsupervised data and its corresponding pseudolabel as a regular term

$$J_1 = \sum_{i=1}^{l} y_{si} \log\left(\hat{y}_{si}\right) \tag{6}$$

$$J_2 = \sum_{i=1}^{l} y_{ui} \log\left(\hat{y}_{ui}\right) \tag{7}$$

$$J_3 = \sum_{i=l}^{n} y_{ui} \log\left(\hat{y}_{ui}\right). \tag{8}$$

In our classification model, a KL discrete stack AE is designed. Its function is to perform feature extraction on the labeled data and unlabeled data in the sample in the same way. In addition, the unsupervised pretraining method retains its weight, reduces the training parameters of the classifier, reduces the structural risk of the classification model, and improves the classification accuracy.

C. Classification Model Design Based on 3-D CNN

In order to extract the spatial and spectral features of the input raw hyperspectral data at the same time, 3-D CNN is used as the backbone network for sharing the unsupervised information classification model. 3-D CNN is usually used to process video files because it takes three-dimensional data as the input attribute so that it can capture two-dimensional pictures and one-dimensional time features in video files at the same time, so it has made achievements in dynamic target recognition and human behavior understanding. As far as this topic is concerned, HSIs are different from images in ordinary computer vision tasks. They are a collection of one-dimensional features that record the spectral response of an object and two-dimensional features that characterize the spatial distribution of the target. Therefore, the use of 3-D CNN to process HSIs directly obtains the spectrum space joint representation of the original data and then performs end-to-end training, which is easier and more accurate for the implementation of classification tasks.

Based on the network structure of 3-D CNN, this section presents the main processing units included in the proposed framework, namely the SAE, supervised and unsupervised feature extraction modules, the parameterization details of the classifier, and the regularization of the model. The implementation

TABLE I
SAE PARAMETERS

Layer	Input	Kernel size	Stride	Activation Function	Output
3D-Conv1	n×n×d	3×3×8×64	1×1×1	ReLU	(n-2)×(n-2)×(d-7)×64
3D-Conv2	(n-2)×(n-2)×(d-7)×64	3×3×3×64	1×1×1	ReLU	(n-4)×(n-4)×(d-9)×64
3D-Conv3	(n-4)×(n-4)×(d-9)×64	1×1×3×128	1×1×2	ReLU	(n-4)×(n-4)×((d-11)/2)×128

process of clustering operations. Finally, this section introduces the training process of this classification model.

1) Stack-type AE: The SAE designed in this section has two AEs based on KL divergence and a softmax classifier. The purpose of unsupervised pretraining is to reduce the hidden weight W and bias term of the network within the parameter space. Generate a better starting point than random initialization for the subsequent supervised training phase.
   Specifically, the two hidden AEs use the same structure, but their input parameters are different; the input to the second AE comes from the output of the first AE, and the parameters are set in Table I.
   After pretraining the SAE, the decoder is separated, the weight of the encoder is saved, and SAE is added to the 3-D CNN-based classification model. This main classification model uses the loss function generated by three cross-entropy for training.
2) Supervised and unsupervised feature extraction module: The main steps of the supervised feature extraction module include global average pooling (GAP), batch normalization (BN), and nonlinear activation. The use of GAP instead of full-connection operation here reduces the redundancy of full connection parameters. Set the BN operation to improve the training speed, and it is no longer sensitive to the weight scale. Use LeakyReLU for nonlinear activation to increase the convergence rate. The unsupervised feature extraction module uses the same settings as the supervised feature extraction module.
3) Classifier settings: This article sets up three classifiers to classify supervised features, unsupervised features, and fusion features. The classifiers are all implemented with the softmax layer, and the purpose is to map different input features to the real label space and the cluster label space.
4) Clustering algorithm: The important part of the information in the shared unsupervised information classification model in this article is the prior knowledge of the sample. The prior knowledge from unsupervised clustering is introduced into the supervised classifier to provide a regularity for the classification model, thereby reducing the model's sample dependence. This article uses the K-means algorithm to cluster all pixels and characterize the sample prototype. Set the value of k to the number of true categories in the sample. That is, the pseudolabels obtained by the clustering algorithm are used as input data.
5) Loss function: The loss function in this article is given by the cross-entropy combination obtained by the three classifiers. The specific formula is as follows:

$$l_{\text{total}} = \frac{\lambda_1}{l} J_1 + \frac{\lambda 2}{l} J_2 + \frac{\lambda_3}{n} J_3. \tag{9}$$

   Among them, λ_1, λ_2, and λ_3 are balance coefficients, J_1, J_2, and J_3 are cross-entropies between the output of the three classifiers and the real labels and the pseudolabels produced by clustering, respectively. Since the three losses are distinguishable, the backpropagation algorithm can be used to effectively train this framework in an end-to-end manner.

IV. EXPERIMENTS RESULTS AND ANALYSIS

This section conducts different experiments on four HSI classified datasets [Pavia University, Kennedy Space Center (KSC), Indian Pines, Salinas] to verify the effectiveness of the proposed method.

A. Datasets

This article uses four internationally popular public benchmark hyperspectral datasets to evaluate the experimental results of the proposed HSI classification algorithm, namely 1) Pavia University, 2) KSC, 3) Indian Pines, and 4) Salinas.

1) Pavia University is a scene captured by ROSIS sensors during a flight mission over Pavia in northern Italy. The size of the original data is 610×610×103, the geometric resolution is 1.3 m, and it contains nine types of ground objects, such as asphalt, gravel, grass, and trees.
2) KSC is the data collected by NASA's AVIRS Research Center at an altitude of approximately 20 km at the KSC in Florida. AVIRIS collected data in 224 10-nm-wide bands, the center wavelength of the data was 400–2500 nm, and the spatial resolution was 18 m.
3) Indian Pines. This scene was collected by the AVIRIS sensor at the Indian Pine test site in northwest Indiana. It consists of 145×145 pixels and 224 spectral reflection bands, with a wavelength range of 400–2500 nm. This scene is a subset of the larger scene. The Indian Pine scene includes two-thirds of agriculture and one-third of forests or other natural perennials.
4) Salinas. This scene was captured by the 224 band AVIRIS sensor over Salinas Valley, California, with a high spatial resolution (3.7 m pixels). The coverage area includes 512 lines by 217 samples. Like the Indian pines, Salinas discarded 20 water absorption bands, which in this case are [108–112], [154–167], 224. This image is only available as sensor radiance data. It includes vegetables, bare soil, and vineyards. Salinas ground truss contains 16 classes.

TABLE II
CLASSIFICATION RESULTS OF DIFFERENT METHODS ON PAVIA UNIVERSITY DATASET

Classes	SVM	DBN	3D CNN	Ours
1	96.74	92.74	96.90	100.00
2	82.09	87.58	96.39	95.81
3	48.88	64.87	66.93	99.94
4	76.47	94.43	82.02	96.99
5	94.86	98.93	100.00	99.44
6	39.77	71.59	71.17	100.00
7	65.33	68.27	62.80	100.00
8	84.60	64.26	81.62	91.48
9	99.79	99.88	100.00	99.39
AA(%)	84.10	90.01	92.28	95.51
OA(%)	85.13	89.50	92.50	94.47
Kappa	0.8211	0.8990	0.9242	0.9527

B. Comparison Methods

In order to verify the effectiveness of the algorithm proposed in this chapter, the four datasets are compared with the classic algorithms in the field of HSI classification. The comparison algorithms are SVM, DBN, and 3-D CNN [37]. SVM is a typical example of traditional algorithms for HSI classification. It can still play a better role in the case of limited training samples. DBN is a generative model used to represent the probability distribution between predicted data and labels. The fine-tuned and pretrained DBN has good performance in HIS classification tasks. 3-D CNN is a successful example used to extract the spatial–spectral features of HSIs synchronously in recent years. The backbone network of the algorithm in this chapter is 3-D CNN.

In addition, in order to further compare the performance of our algorithm, we compare our algorithm with some newer models, such as SSUN [43], SAGP [44], CAG [45], MCNN-CP [46], and BTA-Net [47].

In the experiment, the proportion of samples used for training on the four datasets is 5%, and the remaining samples are used for testing.

C. Evaluation Criterion

We use three different methods including overall accuracy (OA) [38], average accuracy (AA) [39], [21], and Kappa coefficient [41], [42] in this section to compare the performance of different measures.

D. Experiments Results

Under the premise of using the same experimental settings, we conducted a series of classification experiments with different methods of four hyperspectral datasets, and the results were summarized in Tables II–V. And it is worth mentioning that the test data did not participate in unsupervised learning.

We use supervised learning and unsupervised clustering processes to classify and cluster two kinds of data in the same dataset. Because unsupervised clustering is introduced, the information obtained from unsupervised learning is introduced into the supervised learning process in the way of classification

TABLE III
CLASSIFICATION RESULTS OF DIFFERENT METHODS ON KSC DATASET

Classes	SVM	DBN	3D CNN	Ours
1	99.40	92.74	96.90	100.00
2	95.17	99.48	99.06	100.00
3	94.78	84.87	86.93	99.94
4	76.47	94.43	82.02	94.99
5	94.86	98.93	100.00	99.44
6	39.77	71.59	91.17	91.50
7	45.33	68.27	82.80	90.00
8	84.60	64.26	81.62	95.48
9	99.79	99.88	100.00	96.39
10	82.44	95.19	93.09	97.53
11	72.02	96.52	99.50	98.50
12	97.10	97.85	100.00	97.31
13	99.65	87.58	96.39	98.81
AA(%)	85.96	90.89	94.75	96.94
OA(%)	86.26	90.08	93.46	97.62
Kappa	0.8603	0.9056	0.9422	0.9643

TABLE IV
CLASSIFICATION RESULTS OF DIFFERENT METHODS ON INDIAN PINES DATASET

Classes	SVM	DBN	3D CNN	Ours
1	45.01	33.33	79.17	100.00
2	99.65	87.58	96.39	99.81
3	94.78	64.87	76.93	90.94
4	76.47	94.43	82.02	94.99
5	94.86	98.93	100.00	99.44
6	39.77	71.59	71.17	100.00
7	45.33	68.27	62.80	100.00
8	84.60	64.26	81.62	91.48
9	99.79	99.88	100.00	99.39
10	82.44	95.19	93.09	99.53
11	72.02	96.52	99.50	93.50
12	97.10	97.85	100.00	97.31
13	95.17	99.48	99.06	100.00
14	93.67	98.50	99.03	100.00
15	68.64	70.35	89.90	99.15
16	92.78	99.52	98.97	99.66
AA(%)	71.93	86.62	89.55	95.94
OA(%)	72.17	88.33	89.30	94.62
Kappa	0.7364	0.8392	0.9132	0.9601

TABLE V
CLASSIFICATION RESULTS OF DIFFERENT METHODS ON SALINAS DATASET

Classes	SVM	DBN	3D CNN	Ours
1	98.31	87.49	100.00	100.00
2	89.63	87.32	99.97	100.00
3	98.78	83.87	97.21	97.86
4	99.22	95.53	99.64	98.52
5	96.83	99.23	99.78	99.21
6	89.87	91.34	99.87	100.00
7	83.33	93.22	95.63	96.44
8	77.85	78.56	77.14	96.48
9	98.22	91.91	99.44	98.84
10	78.64	94.12	98.25	99.40
11	76.99	86.22	98.16	97.51
12	98.12	89.92	90.01	99.31
13	89.70	88.36	99.06	95.58
14	80.55	98.21	99.23	99.43
15	78.32	79.71	93.66	99.15
16	89.09	97.55	92.96	100.00
AA(%)	91.60	92.32	93.55	95.43
OA(%)	89.25	89.63	94.46	96.58
Kappa	0.8282	0.8617	0.9292	0.9551

loss. This process not only provides information of similarities and differences between classes but also can obtain more information from supervised learning.

1) Comparison Results on the Pavia University Dataset: For the classification of bitumen, self-blocking bricks, and painted metal sheets, the algorithm proposed in this chapter can achieve 100% accuracy, which shows the ability of this algorithm to distinguish man-made materials. It can be calculated from Table II that on the Pavia university dataset, compared with the traditional algorithm SVM, the average classification accuracy of the algorithm proposed in this chapter has increased by 11.14%, and the overall classification accuracy has increased by 11%. The Kappa coefficient increased by 0.1316. In addition, compared with the method based on deep learning, the average classification accuracy of the algorithm proposed in this chapter has increased by more than 5%, the overall classification accuracy has increased by 5%, and Kappa has increased by 0.0537.

2) Comparison Results on the KSC Dataset: Both traditional methods and deep-learning-based method classification results on the KSC dataset have merits from Table III. The method based on deep learning obtains better performance than the traditional method SVM due to its deep feature extraction ability. The classification results on the category shrubs (Scrub) and salt marsh (Salt Marsh) are significantly better than SVM. But the classification effect of SVM on category 13 graminoid marsh is better. Grass swamps account for a small proportion in the KSC dataset, because SVM is not sensitive to high-dimensional data, and methods based on deep learning, including the algorithm in this chapter, have poor performance in categories with a small number of samples because of too many parameters. Compared with the current classification algorithm, the AA value of the algorithm proposed in this article is increased by 2.2%, OA is increased by 4.2%, and Kappa is increased by 0.0221.

3) Comparison Results on the Indian Pines Dataset: The classification accuracy of 3-D CNN on the wheat category is higher than the algorithm in this chapter. However, in most categories, the algorithm in this chapter is better than 3-D CNN. In general, the algorithm proposed in this chapter improves AA by more than 6.3%, OA by 5.32%, and Kappa by more than 0.04 on the Indian Pines dataset from Table IV.

4) Comparison Results on the Salinas Dataset: It also can be seen from Table V, on the Salinas dataset, our experimental results are still better than SVM, DBN, and 3-D CNN. In general, the accuracy in many classes has reached 100%. Also, OA is increased by 2%–7% and AA is increased by 2%–4% than others.

5) Other Hyperspectral Classification Frameworks: From Tables VI and VII, we can see that our model is much better than other models in the dataset of Pavia University. We found the Grass-pasture-mowed and Oats categories of the Indian Pines dataset have only 28 and 20 samples, respectively. Even if 5% of the samples were selected, only one sample was available for training in this experiment. However, because our algorithm can regularize the supervised classification model by making full use of the unsupervised information learned from the unlabeled data, which can effectively alleviate the overfitting problem caused by the small sample phenomenon. Because the Pavia University dataset has a larger sample size than the Indian Pines dataset, the accuracy of the classification is relatively high, our algorithm also has obvious advantages over SAGP, MCNN-CP, and others.

TABLE VI
ACCURACY OF DIFFERENT METHODS ON INDIAN PINES DATASET

methods	AA(%)	OA(%)	Kappa
SSUN	73.79	73.59	0.7006
SAGP	73.49	76.58	0.7163
CAG	77.01	78.37	0.7372
MCNN-CP	77.69	77.78	0.7455
BTA-Net	81.62	83.22	0.7894
ours	95.94	94.62	0.9601

TABLE VII
ACCURACY OF DIFFERENT METHODS ON PAVIA UNIVERSITY DATASET

methods	OA(%)	AA(%)	Kappa
SSUN	93.59	92.47	0.9151
SAGP	91.09	89.75	0.8814
CAG	95.28	94.05	0.9340
MCNN-CP	93.95	92.51	0.9197
BTA-Net	95.85	94.95	0.9377
ours	95.51	94.47	0.9527

TABLE VIII
ACCURACY OF DIFFERENT METHODS ON SALINAS DATASET

methods	OA(%)	AA(%)	Kappa
SSUN	90.33	88.23	0.8922
SAGP	92.94	95.69	0.9181
CAG	93.06	95.84	0.9357
MCNN-CP	92.11	94.62	0.9121
BTA-Net	94.34	96.96	0.9369
ours	95.43	96.58	0.9551

TABLE IX
TEST TIMES OF OUR MODEL AND OTHER METHODS ON DIFFERENT DATASETS

methods	Indian Pines	Pavia University	KSC	Salinas
SSUN	15.55s	24.11s	6.61s	39.25s
SAGP	3.72s	10.77s	4.11s	13.29s
BTA-Net	1.03s	4.21s	1.12s	4.55s
ours	2.41s	5.36s	1.31s	6.837s

In Table VIII, the sample distribution of the Salinas dataset is more balanced than the Indian Pines and Pavia University datasets, and the classification difficulty is lower. Because our algorithm effectively strengthens the input unsupervised information in a supervised way, and the feature extraction module efficiently shares the unsupervised information in the data, the unsupervised information flows to the supervised task in the classification process, providing an effective regularity for the network, so the experimental results are more robust.

E. Runtime Analysis of Algorithm

Table IX shows calculation times for our algorithm and other comparison algorithms. As shown in the table, the proposed model is faster than SSUN and SAGP. Especially on dataset Salinas, our algorithm is two times and six times faster than

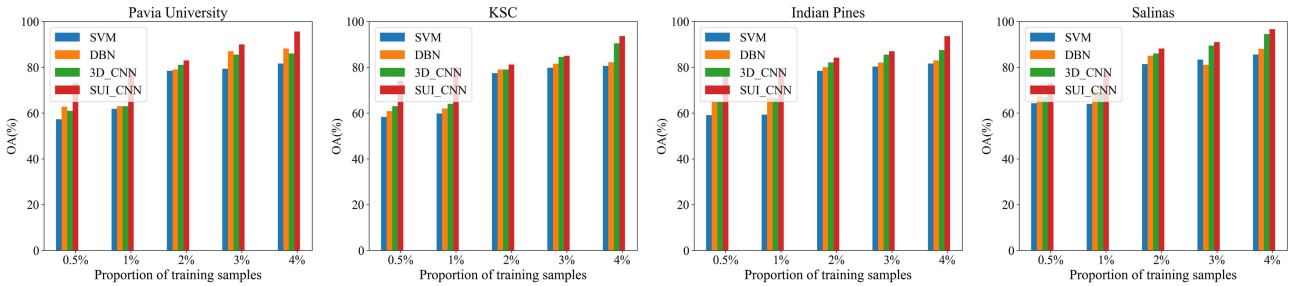

Fig. 4. Classification accuracy under different training sample ratios (%).

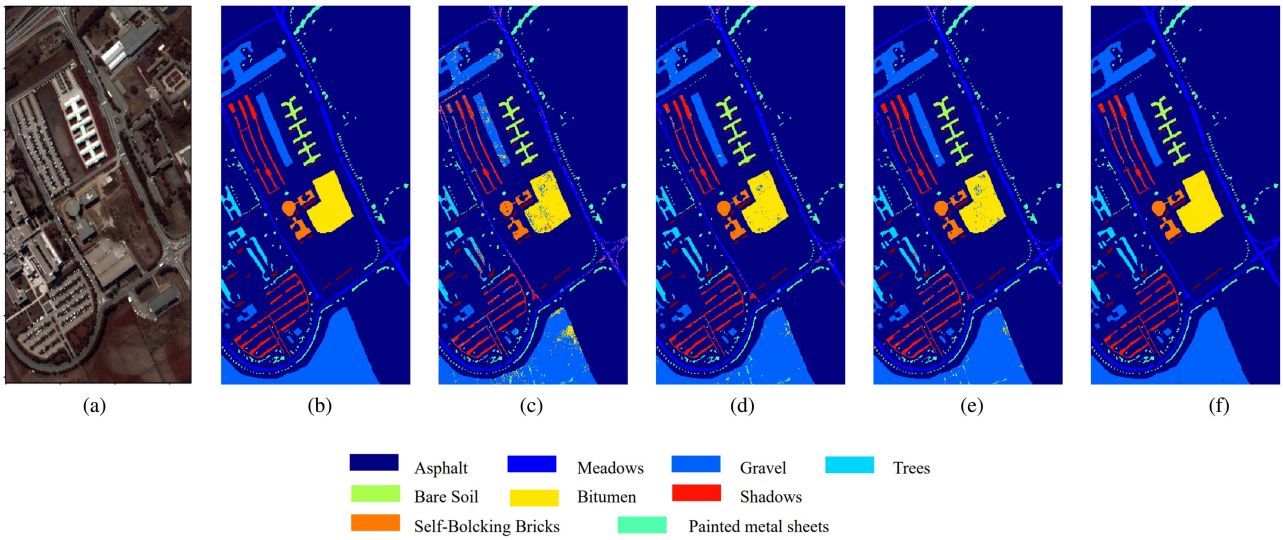

Fig. 5. Visual classification results on the Pavia University dataset. (a) PaviaU_RGB. (b) Groundtruth. (c) SVM. (d) DBCN. (e) 3-D CNN. (f) Ours.

SAGP and SSUN, respectively. Besides, compared with BTA-Net algorithms, although our algorithm is slightly lacking in computational complexity, in combination with OA, AA, and Kappa on Indian Pines, Pavia University, and Salinas datasets, our algorithm still has certain advantages. In general, our algorithm has greatly improved its accuracy while maintaining a relatively good computational complexity.

F. Classification Performance in the Case of Small Samples

In order to verify that the algorithm in this chapter alleviates the overfitting of the model by introducing regularization and improves the classification accuracy in the case of small samples. It alleviates the problem of insufficient classification accuracy of existing algorithms when the training samples are small and compares with other experiments under the condition of small samples. Specifically, the proportion of input training samples to the total number of pixels is set to 0.5%, 1%, 2%, 3%, and 4% on these four datasets and compare them with other experiments. The results are as follows in Fig. 4.

It can be seen that this algorithm proposed in the article has a small number of training samples, taking 1% and 3% as examples, and the classification accuracy on the four datasets is higher than the comparison algorithm. Especially in the case of 1% training samples, the algorithm OA proposed in this chapter can take more than 75%, which is a big improvement compared with the comparison algorithm. Especially the comparison with 3-D CNN shows that this chapter introduces the unsupervised information of the sample species into the training process, in the case of a very low sample size, the accuracy of the model did not decrease quickly, which can improve the effectiveness of the algorithm when the training samples is too few.

In the third type of object gravel (Gravel), no matter the traditional algorithm SVM or the method based on deep learning, there is a phenomenon of misclassification, as shown in Fig. 5. Part of the gravel is divided into trees (Trees), as shown in Fig. 6. Extracting the spectral information of gravel and trees in the Pavia University dataset, as shown in Fig. 6, it can be seen that the spectral curves of gravel and trees are relatively similar, so it is easy to misclassify. However, the accuracy of the algorithm proposed in this chapter is above 96% in these two categories, which can prove the effectiveness of the algorithm in the similar band.

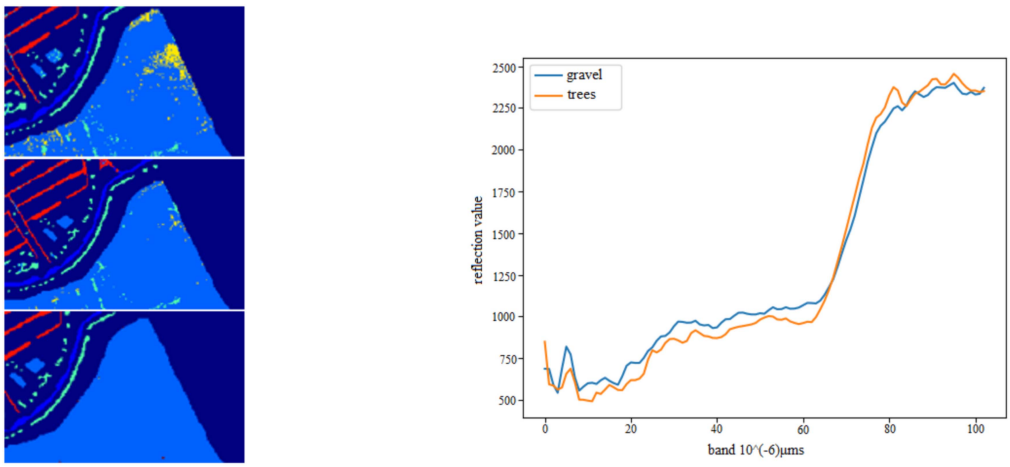

Fig. 6. Compare the classification of the algorithm in the gravel and trees categories. (a) SVM and DBN have many wrong marks on the level. (b) Spectral curves of gravel and trees.

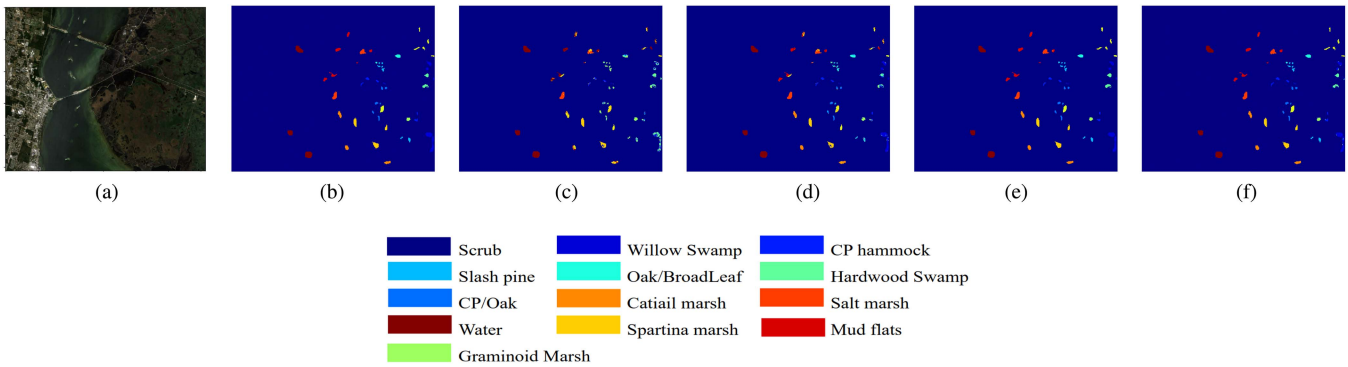

Fig. 7. Visual classification results on the KSC dataset. (a) KSC_RGB. (b) Groundtruth. (c) SVM. (d) DBCN. (e) 3-D CNN. (f) Ours.

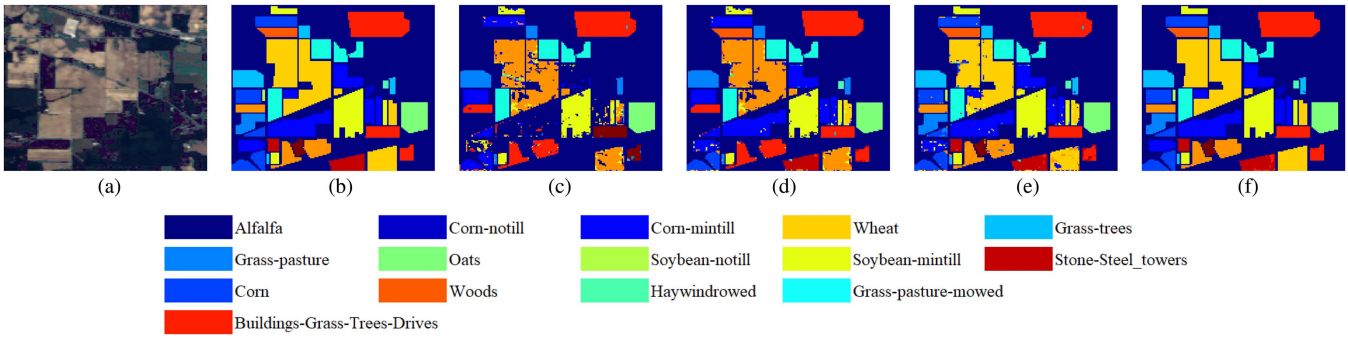

Fig. 8. Visual classification results on the Indian Pines dataset. (a) Indian Pines. (b) groundtruth. (c) SVM. (d) DBCN. (e) 3-D CNN. (f) Ours.

It can be seen from Fig. 7 that in the comparison experiments, some of the shrubs (Scrub) in the upper right corner of the image were mistakenly classified as salt marsh (Salt Marsh). But our algorithm avoids this, and the classification accuracy rate in the salt marsh category is 100%. In addition, the classification ability of mud flats is also more accurate in this chapter. As can be seen from Fig. 8, wheat on the Indian Pines dataset is easily classified as woods. SVM and DBN have low classification accuracy for these two categories. There are many misclassifications between stone–steel–towers and buildings–grass–trees–drives (buildings–grass–trees–drives).

It also can be seen from Fig. 9, on the Salinas dataset, our algorithm performed better than SVM, DBCN, and 3-D CNN on the class fallow_rough_plow.

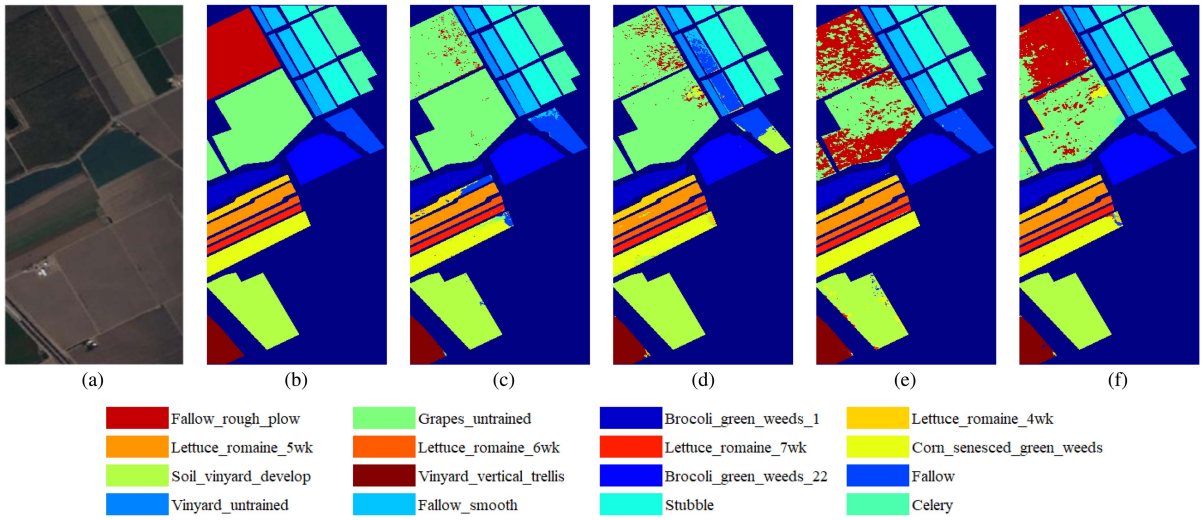

Fig. 9. Visual classification results on the Salinas dataset. (a) Salinas. (b) Groundtruth. (c) SVM. (d) DBCN. (e) 3-D CNN. (f) Ours.

V. Conclusion

Based on the idea of introducing regularization to alleviate overfitting, this article shares the unsupervised information of the complete set of samples into the training process of supervised classification and designs a 3-D CNN-based shared unsupervised information HSI classification model. Considering that HSIs have fewer training samples in practical applications, the classification method that introduces unsupervised information proposed in this chapter aims to alleviate the overfitting problem caused by small samples in the depth model. When compared with the traditional method SVM and the typical methods based on deep learning, such as DBN and 3-D CNN, this algorithm proposed in this chapter has higher classification accuracy in most categories. Also, in the case of reducing training samples, the algorithm proposed in this article is still advantageous.

References

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

[2] L. Liu, M. Li, Z. Zhao, and J. Qu, "Recent advances of hyperspectral imaging application in biomedicine," *Chin. J. Lasers*, vol. 45, no. 2, 2018, Art. no. 0207017.

[3] T. Alipourfard, H. Arefi, and S. Mahmoudi, "A novel deep learning framework by combination of subspace-based feature extraction and convolutional neural networks for hyperspectral images classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2018, pp. 4780–4783.

[4] T. Bahraini, P. Azimpour, and H. S. Yazdi, "Modified-mean-shift-based noisy label detection for hyperspectral image classification," *Comput. Geosci.*, vol. 155, no. 4, 2021, Art. no. 104843.

[5] K. Chatterjee, W. Dvorak, M. Henzinger, and A. Svozil, "Algorithms and conditional lower bounds for planning problems," *Artif. Intell.*, vol. 297, 2018, Art. no. 103499.

[6] Y. Chen, M. Crawford, and J. Ghosh, "Applying nonlinear manifold learning to hyperspectral data for land cover classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2005, pp. 4311–4314.

[7] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 6232–6251, Oct. 2016.

[8] T. Dong, C. Yang, and Y. Zhang, "Deep metric learning with online hard mining for hyperspectral classification," *Remote Sens.*, vol. 13, no. 7, 2021, Art. no. 1368.

[9] J. Duran, "Dissecting scientific explanation in AI (sXAI): A case for medicine and healthcare," *Artif. Intell.*, vol. 297, 2021, Art. no. 103498.

[10] H. Yu, X. Shang, X. Zhang, L. Gao, M. Song, and J. Hu, "Hyperspectral image classification based on adjacent constraint representation," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 4, pp. 707–711, Apr. 2021.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[12] H. Liang and Q. Li, "Hyperspectral imagery classification using sparse representations of convolutional neural network features," *Remote Sens.*, vol. 8, no. 2, 2016, Art. no. 99.

[13] G. Hirano, M. Nemoto, Y. Kimura, Y. Kiyohara, and T. Nagaoka, "Automatic diagnosis of melanoma using hyperspectral data and GoogleNet," *Skin Res. Technol.*, vol. 26, no. 6, pp. 891–897, 2020.

[14] M. Ivanovici, M. Marincas, and R. M. Coliban, "A vector median filter for hyperspectral images based on lexicographic ordering of estimated autocorrelation functions," in *Proc. 11th Workshop Hyperspectral Imag. Signal Process. Evol. Remote Sens.*, 2021, pp. 1–4.

[15] B. Jiang et al., "Fusion of machine vision technology and AlexNet-CNNs deep learning network for the detection of postharvest apple pesticide residues," *Artif. Intell. Agriculture*, vol. 1, pp. 1–8, 2019.

[16] R. G. Pontius, Jr and M. Millones, "Death to kappa: Birth of quantity disagreement and allocation disagreement for accuracy assessment," *Int. J. Remote Sens.*, vol. 32, no. 15, pp. 4407–4429, 2011.

[17] J. Lin, H. Chen, Z. J. Wang, and S. Li, "Structure preserving transfer learning for unsupervised hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 10, pp. 1656–1660, Oct. 2017.

[18] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.

[19] H. Shafri, A. Suhaili, and S. Manso, "The performance of maximum likelihood, spectral angle mapper, neural network and decision tree classifiers in hyperspectral image analysis," *J. Comput. Sci.*, vol. 3, no. 6, pp. 419–423, 2007.

[20] K. Y. Ma and C. I. Chang, "Iterative training sampling coupled with active learning for semisupervised spectral-spatial hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 10, pp. 8672–8692, Oct. 2021.

[21] 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.

[22] E. S. Olivas, J. D. M. Guerrero, M. Martinez-Sober, J. R. MagdalenaBenedito, and A. S. Lopez, *Handbook of Research on Machine Learning Applications and Trends (Algorithms, Methods, and Techniques)–Principal Graphs and Manifolds*. Hershey, PA, USA: IGI Global, 2010, pp. 457–481, doi: 10.4018/978-1-60566-766-9.

[23] M. E. Paoletti, J. M. Haut, N. S. Pereira, J. Plaza, and A. Plaza, "Ghostnet for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 12, pp. 10378–10393, Dec. 2021.

[24] Z. Liang and Z. Xingliang, "Spectral–spatial exploration for hyperspectral image classification via the fusion of fully convolutional networks," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 659–674, Oct. 2020, doi: 10.1109/JSTARS.2020.2968179.

[25] D. Qian and C. I. Chang, "A linear constrained distance-based discriminant analysis for hyperspectral image classification," *Pattern Recognit.*, vol. 34, no. 2, pp. 361–373, 2001.
[26] A. Qin, Z. Shang, J. Tian, and Y. Wang, "Spectral–spatial graph convolutional networks for semisupervised hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 15, pp. 241–245, Feb. 2019.
[27] Y. Quan, X. Zhong, W. Feng, C. W. Chan, and M. Xing, "Smote-based weighted deep rotation forest for the imbalanced hyperspectral data classification," *Remote Sens.*, vol. 13, no. 3, 2021, Art. no. 464.
[28] Z. Ren and W. U. Lingda, "Hyperspectral band selection based on affinity propagation," *Ship Electron. Eng.*, vol. 32, no. 15, pp. 1–4, 2018.
[29] B. Settles, "Active learning," *Synth. Lectures Artif. Intell. Mach. Learn.*, vol. 6, no. 1, pp. 1–114, 2012.
[30] B. Settles, "Weighted ensemble object detection with optimized coefficients for remote sensing images," *ISPRS Int. J. Geo-Inf.*, vol. 9, no. 6, 2020, Art. no. 370.
[31] Y. Tarabalka, M. Fauvel, J. Chanussot, and J. A. Benediktsson, "SVM and MRF-based method for accurate classification of hyperspectral images," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 4, pp. 736–740, Oct. 2010.
[32] L. Tong, J. Zhang, and Z. Ye, "Classification of hyperspectral image based on deep belief networks," in *Proc. IEEE Int. Conf. Image Process.*, 2014, pp. 5132–5136.
[33] S. Liu, B. Ding, J. Bai, and Z. Xiao, "Hyperspectral image classification based on extended morphological profile features and ghost module," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2021, pp. 3617–3620.
[34] X. Shen, H. Yu, C. Yu, Y. Wang, and M. Song, "Global spatial and local spectral similarity based sample augment and extended subspace projection for hyperspectral image classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2021, pp. 3637–3640.
[35] H. Yu, H. Zhang, Y. Liu, K. Zheng, Z. Xu, and C. Xiao, "Dual-channel convolution network with image-based global learning framework for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, Apr. 2022, Art no. 6005705.
[36] W. Wei, L. Zhang, Y. Li, C. Wang, and Y. Zhang, "Intra-class similarity structure representation based hyperspectral imagery classification with few samples," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, no. 99, pp. 1045–1054, Oct. 2020.
[37] R. Xin, Z. Jiang, and Y. Shao, "Complex network classification with convolutional neural network," *Tsinghua Sci. Technol.*, vol. 25, no. 4, pp. 447–457, 2020.
[38] Z. Xu, H. Yu, K. Zheng, L. Gao, and M. Song, "A novel classification framework for hyperspectral image classification based on multiscale spectral-spatial convolutional network," in *Proc. 11th Workshop Hyperspectral Imag. Signal Process. Evol. Remote Sens.*, 2021, pp. 1–5.
[39] Z. Xue, X. Yu, B. Liu, X. Tan, and X. Wei, "HResNetAM: Hierarchical residual network with attention mechanism for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 3566–3580, Oct. 2021, doi: 10.1109/JSTARS.2021.3065987.
[40] C. Yu, R. Han, M. Song, C. Liu, and C. I. Chang, "Feedback attention-based dense CNN for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Dec. 2021, Art. no. 5501916, doi: 10.1109/TGRS.2021.3058549.
[41] X. Zhao and X. Zhang, "Multi-frame super-resolution reconstruction algorithm of optical remote sensing images based on double regularization terms and unsupervised learning," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 35, no. 3, 2020, Art. no. 2154002.
[42] Y. Zhao, Y. Yuan, and Q. Wang, "Fast spectral clustering for unsupervised hyperspectral image classification," *Remote Sens.*, vol. 11, no. 4, 2019, Art. no. 399.
[43] Y. Xu, L. Zhang, B. Du, and F. Zhang, "Spectral–spatial unified networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 10, pp. 5893–5909, Oct. 2018.
[44] H. You, S. Tian, L. Yu, and Y. Lv, "Pixel-level remote sensing image recognition based on bidirectional word vectors," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 2, pp. 1281–1293, Feb. 2020.
[45] W. Cai and Z. Wei, "Remote sensing image classification based on a cross-attention mechanism and graph convolution," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, Jan. 2022, Art. no. 8002005, doi: 10.1109/LGRS.2020.3026587.
[46] J. Zheng, Y. Feng, C. Bai, and J. Zhang, "Hyperspectral image classification using mixed convolutions and covariance pooling," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 522–534, Jan. 2021.
[47] W. Cai et al., "A novel hyperspectral image classification model using bole convolution with three-directions attention mechanism: Small sample and unbalanced learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, Jan. 2023, Art. no. 5500917, doi: 10.1109/TGRS.2022.3201056.
[48] C. Deng, Y. Xue, X. Liu, C. Li, and D. Tao, "Active transfer learning network: A unified deep joint spectral-spatial feature learning model for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 3, pp. 1741–1754, Mar. 2019, doi: 10.1109/TGRS.2018.2868851.

Jian Ji was born in Xi'an, China, in 1971. She received the B.S. degree in computational mathematics from Northwest University, Xi'an, in 1993, and the Ph.D. degree in computer science and technology from Northwestern Polytechnical University, Xi'an, in 2007.

She is currently a Professor with the School of Computer Science and Technology, Xidian University, Xi'an. Her research interests include computational intelligence, pattern recognition, and image analysis.

Shuiqiao Liu was born in Shaanxi, China, in 1995. She received the B.S. degree in mathematics and applied mathematics from Northwestern University, Xi'an, China, in 2017, and the master's degree in computer science and technology from Xidian University, Xi'an, in 2021.

She is currently with AVIC Xi'an Aeronautical Computing Technology Research Institute, Shaanxi. Her research interest includes pattern recognition.

Fangrong Zhang was born in Gansu, China, in 1999. She received the B.S. degree in computer science and technology in 2021 from Xidian University, Xi'an, China, where she is currently working toward the master's degree in computer science and technology.

Her research interests include artificial intelligence and image analysis.

Xianfu Liao was born in Henan, China, in 1997. He received the B.S. degree in network engineering major from Henan University, Kaifeng, China, in 2020. He is currently working toward the master's degree in computer science and technology with Xidian University, Xi'an, China.

His research interest includes pattern recognition.

Shuzhen Wang was born in 1978. He received the Ph.D. degree in engineering from the School of Electro-Mechanical Engineering, Xidian University, Xi'an, China, in 2005.

He is currently a Professor with the School of Computer Science and Technology, Xidian University, and works with the Xi'an Innovation Academy of Industrial Internet. His research interests include radar imaging, machine learning, and computer vision.

Junru Liao was born in Sichuan, China, in 1999. She received the B.S. degree in information management and information system from Three Gorges University, Yichang, China, in 2021. She is currently working toward the master's degree in computer science and technology with Xidian University, Xi'an, China.

Her research interests include artificial intelligence and image analysis.