Semisupervised Hyperspectral Image Classification Using Spatial-Spectral Information and Landscape Features

XIAOWEI JI, (Student Member, IEEE), YING CUI, HENG WANG, LONG TENG, LINGXIU WANG, AND LIGUO WANG, (Member, IEEE)

College of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China

Corresponding author: Ying Cui (cuiying@hrbeu.edu.cn)

ABSTRACT In hyperspectral image classification, the foremost task is that: how can we apply limited labeled samples to achieve good classification results? Spatial–spectral classification methods, which assign a label to each pixel regarding both spatial and spectral information, are effective to improve classification performance. Moreover, semisupervised learning (SSL) focuses on the scenario that the number of labeled data is rather small while a large number of unlabeled data are available. To complement spatial–spectral classification methods and semisupervised learning for each other, we propose a novel learning landscape features semisupervised framework (LLFSF) based on M-training algorithm and weighted spatial-spectral double layer SVM classifiers module (WSS-DSVM). In this novel framework, we first propose a SLIC (simple linear iterative clustering) based non-local superpixel segmentation algorithm to initially learn landscape feature and spatial composition. Then, we apply WSS-DSVM module to obtain initial classification maps. To better characterize complex scenes of hyperspectral images, we quantizes both the landscape diversity and separability from the initial classification map, which increase availability of spatial details and structural information of objects. Finally, we put some patches with lower accuracy into Multiple-training algorithm for further classification. In order to achieve an unbiased evaluation, we have evaluated the performance of LLFSF on three different scene hyperspectral data sets and compare it with that of three state-of-the-art hyperspectral image classification methods. The experimental results confirm the efficacy of the proposed framework.

INDEX TERMS Hyperspectral image classification, landscape features, spatial-spectral information, semisupervised learning.

I. INTRODUCTION

Recently, in pace with the rapid development of imaging technology, hyperspectral imagery can obtain a large amount of information about an object via hundreds of contiguous and narrow spectral bands. Hyperspectral imagery (HSI) has emerged as a significant data in a variety of scientific fields, including medical imaging [1], chemical analysis [2], and remote sensing [3], agricultural monitoring [4], ecosystem monitoring [5] and endmember extraction [6]. The crucial component in these applications is classification.

The classification techniques are divided into supervised classification algorithms and unsupervised classification algorithms based on whether a prior knowledge is needed. Some conventional supervised classifiers can obtain satisfactory classification results, such as support vector machines [7], [8], neural networks [9], [10] and regression methods [11]. Recently, as the supervised models, deep learning networks have attracted much attention, due to the fact that the advantages of deep learning models. Firstly, the fundamental philosophy of deep learning is that let the trained model itself select more important features with fewer constraints imposed by human experts. They simultaneously learn feature representation and corresponding classifiers in

The associate editor coordinating the review of this manuscript and approving it for publication was Yongqiang Zhao.

the training process. Furthermore, multilayer neural networks can capture representative and discriminative information of large data. Multiple researches have demonstrated that CNNs can deliver the state-of-the-art results using spatialized input for HSI classification. Zhong *et al.* proposed an end-to-end spectral–spatial residual network (SSRN), which can take raw 3-D cubes as input data [12]. The residual blocks connect every other 3-D convolutional layer through identity mapping, which promotes the back propagation of gradients. Moreover, the batch normalization on every convolutional layer can regularize the learning process and improve the classification performance of trained models. Liu *et al.* proposed a novel supervised deep feature extraction technique based on siamese convolutional neural network (S-CNN), which contributes to enhance the performance of hyperspectral image classification [13]. This method applies a CNN with five layers to directly extract deep features from hyperspectral cube. Then, the siamese network composed by two CNNs learns the features that show a low intra-class and high inter-class variability.

Taking into account the aforementioned methods, there are two serious problems for the classification of remote sensing imagery: 1) They have satisfactory performance especially when the number of labeled training data is large. 2) Their performance highly relies on the quality of the training samples. However, the time consumption and the cost of data collection are very high and the available training samples are usually not enough. Compared with supervised classification algorithms, unsupervised classification doesn't need labeled training data. However, unsupervised classification algorithms are considerably more challenging and can be defined as the identification of natural distribution or structures within the data. K-means algorithm [14], nearest neighbor clustering [15] and fast density peak-based clustering (FDPC) [16] are commonly applied in unsupervised classification.

Can we label only limited samples to train an effective classifier and obtain accurate classification maps? The answer is yes. Semisupervised learning provide a promising way to solve this problem [17]–[19]. In general, semisupervised learning algorithms usually include multi-view learning algorithms [20], co-training algorithms [21], self-learning algorithms [22], tri-training algorithms [23], and the graph-based approaches [24], transductive support vector machines (TSVMs) generative models [25]. Semisupervised learning pays more attention to the unlabeled data in unsupervised approach. It improves the supervised model by increasing the quantity of the training samples, and enhances the generalization ability of the classifiers by applying the a small number of labeled data and a large number of unlabeled data.

The main problems of semisupervised learning approach are that how to select the most valuable unlabeled samples to train classifiers and how to determine the label of these new selected samples. Multiple classifier systems have been widely applied in semisupervised learning. Two common semisupervised learning methods, namely, co-training and self-training, was investigated by experiments. Co-training is a semi-supervised learning paradigm which trains two learners respectively from two different views and lets the learners label some unlabeled examples for each other. In [21], Munkhdalai *et al.* presented a semisupervised integration of different classifiers to cover knowledge from unlabeled data to recognize bio named entities in text. Moreover, many semisupervised learning frameworks based on the tri-training scheme are proposed for the classification of hyperspectral data. In [5], a semisupervised learning framework based on tri-training and self-learning were proposed. The framework involved two steps: First, the multiple classifiers learned by the improved tri-training and then integrated the outputs of all classifiers to the final hypothesis. Next, in order to solve the ill-posed classification problem, the confidence of each learner was measured by the improved estimation of classifier error. Simultaneously, self-training also expanded the labeled set by adding unlabeled samples with correct labels assigned by classifiers.

In [26], Li *et al.* proposed a weakly supervised deep learning (WSDL) method for multiple class geospatial object detection using scene-level tags only. Differing from existing WSDL methods, This novel method exploits both the separate scene class information and mutual cues between scene pairs, which aims at sufficiently training deep networks for pursuing the superior object detection performance. In detail, it first to leverage pair-wise scene-level similarity to learn discriminative convolutional weights by learning the mutual information between scene pairs. Then, it utilizes point-wise scene-level tags to learn class-specific activation weights. In [27], Kellenberger *et al.* presented an Active Learning (AL) strategy for re-utilizing a deep convolutional Neural Network (CNN) based object detector on a new dataset. Specifically, this method first bridges the gap by applying AL method and introduces a new criterion called Transfer Sampling (TS) to search corresponding regions between the source and target data in the space of CNN activations. And the CNN scores in the source data set are applied to rank the samples according to their likelihood of being animals, and this ranking is transferred to the target data set.

In the published context [28], learning features via landscape metrics, which are derived from the initial classification results can quantify both the landscape composition and spatial configuration and also refine the classification model. Hence, in this paper, we propose a novel framework, which aims at adequately exploiting spatial configuration and spectral information by learning landscape features from the whole data set. We first investigate SLIC (simple linear iterative clustering) based non-local superpixel segmentation algorithm to initially analyze landscape features and results. Then, we implement spectral-spatial double SVM classifiers module (WSS-DSVM) to obtain initial classification map. We feed the patches with lower certainty into Multiple-training (M-training) algorithm, aiming to select informative unlabeled samples and enhancing the discriminative ability of the corresponding classification model.

Meanwhile, we relearn the landscape features based on initial classification map. Finally, LLFSF flexibly combines the strengths of learning landscape features, spatial-spectral SVM classifiers and M-training via the classification certainties calculated by the probabilistic output of the respective classifiers.

The rest of this paper is organized as follows. Section II presents details of the related works on hyperspectral remote sensing image classification. Section III describes the proposed LLFSF framework in detail. Section IV specifically describes the effectiveness of LLFSF framework in different data sets. Section V summarizes this paper.

II. RELATED WORK

A. SPATIAL-SPECTRAL CLASSIFICATION ALGORITHM FOR HYPERSPECTRAL IMAGE

Recent studies show that sufficiently exploiting the spatial and spectral information can achieve a better performance in the classification of hyperspectral images. Spectral and spatial classifiers, which can assign more precise label to each pixel by considering its spatial and spectral information, also can increase overall classification accuracy. David Landgrebe and his team at Purdue University firstly introduced spatial context into a multiband image classification with the Extraction and Classification of Homogeneous Objects (ECHO) classifier [29]. According to the existing literatures, spectral-spatial classification algorithms can be roughly divided into four types i.e., spatial feature extraction-based methods [30], [31], probabilistic model-based methods [32], [33], segmentation-based methods [34], and fixed window-based methods [35]. Morphological profiles is a representative spatial feature extraction-based method. This method removes additional structural information and closes operator. However, opening operator is a dual operator, which applies image dilation and recovers abundant structural information. Hence, Morphological profiles (MPs) is effective at providing spatial features with only one band. Markov random fields (MRFs) is another recommendable technique in exploiting the spatial relatedness between adjacent pixels and it has been proven to be much more accurate than the pixelwise methods to classify hyperspectral images. [36]. As segmentation-based method, Cui *et al.* proposed a robust strategy to enforce the local collaborative property in homogeneous areas. This method first divides the denoised image into superpixels by considering the quarternion theory, then it implements collaborative sparse unmixing in each superpixel [37]. With respect to fixed window-based methods, Baassou *et al.* proposed an integrated spatial-spectral information algorithm for hyperspectral image classification [38]. This algorithm utilizes spatial pixel association (SPA) by exploiting spectral information divergence (SID), and applying spectral clustering techniques to reduce regions. The Weighted Spatial-Spectral Distance (WSSD) was proposed by Huang *et al.* in [39]. This proposed algorithm applies spatial window and spectral factor to obtain the spatial information and spectral information, respectively. Then it applies the spatial nearest points to reconstruct the center point and focuses on the local spatial structure. By increasing both the consistency of the same class pixels and the difference of the different class pixels, this algorithm effectively reduces the redundant spatial-spectral information and also obtains discriminating and representative features. Deep learning as popular techniques has drawn much attention in hyperspectral images classification. In [40], Yang *et al.* proposed a deep convolutional neural network with two-branch architecture to capture the joint spectral-spatial features from HSIs. Initially, this framework learns spectral features and spatial features, then learned spectral features and spatial features are concatenated and imported to fully connected layers to extract the joint spectral-spatial features. Furthermore, this framework implements transfer learning, when the training samples are limited. In detail, Low-layers and mid-layers of the network are pre-trained and transferred from other data sources and only top layers are trained with limited training samples extracted from the target scene images.

Taking into account the aforementioned spectral-spatial classification algorithms, although they enhance the classification performance, most of them neglect the separability and difference of the used bands. The equal employment of each band will lead to the underuse or overuse of the bands, which suggests that the time consumption of data collection may be greatly increased. Therefore, both the complex spectral information and dependent contribution of each band should be investigated.

B. MULTIPLE TRAINING ALGORITHM

Tri-training is an effective semisupervised learning algorithm [41], [42]. According to this integration strategy, more different classifiers will feedback the final classifier a better result. In tri-training algorithm, three classifiers are initially trained from the original training set, and then are iteratively trained by applying informative unlabeled samples in the process. In [41], Zhu *et al.* proposed a novel tri-training based on spatial neighborhood information to settle the problem in the traditional tri-training algorithm. Firstly, they select three better classifiers from MLR (Multinomial Logistic Regression), KNN (K Nearest Neighbor), ELM (Extreme Learning Machine) and RF (Random Forest) classifier based on disagreement measure and accuracy. All the classifiers are redefined by utilizing unlabeled samples in the training process. Then, the unlabeled samples are labeled for each classifier by the following two steps. Step 1: selecting unlabeled samples that receiving same labels from two classifiers. Step 2: spatial neighborhood information of initial training samples is applied in this proposed approach to construct the secondary screening of unlabeled samples.

In [23], Cui *et al.* has proposed a novel semisupervised classification method for hyperspectral data based on tri-training. This method combines different classifiers and stratified sampling based on labeled samples, which can increase the diversity of classifiers and avoid classifiers performance deterioration. In order to yield complementary and reliable

Landscape metrics	Calculation	Description	Performance
Mean patch size(MPS)	$\dfrac{\sum_{i=1}^{n} a_i}{n}$	a_i is the area (m^2) of patch i n is the number of patches for class i	The relative size of the patches in the landscape
Standard deviation of area (AREA_SD)	$Std\left(\bigcup_{i=1}^{n} a_i\right)$	a_i is the area (m^2) of patch i	Area distribution of different land-cover classes
Largest patch index (LPI)	$\dfrac{\max_{i=1}^{n}(a_i)}{A}$	a_i is the area (m^2) of patch i A is the total landscape area (m^2)	The percentage of total landscape area comprised by the largest land-cover patch, highlighting the dominant type in an urban scene
Edge density (ED)	$\dfrac{E}{A}$	E is the total length of edges in the landscape, and A is the total landscape area (m^2)	Total edges of a land-cover class relative to the total landscape area, quantifying the landscape structure from the edge aspect
Mean shape index (SHAPE_MN)	$\dfrac{\sum_{i=1}^{n} \dfrac{p_i}{2\sqrt{\pi a_i}}}{n}$	p_i is the perimeter of land-cover patch i, a_i is the area of the land-cover patch, and n is the number of patches within the landscape	Average measure of the shape complexity of each land-cover class
Standard deviation of shape index (SHAPE_SD)	$Std\left(\bigcup_{i=1}^{n} \dfrac{p_i}{2\sqrt{\pi a_i}}\right)$	p_i is the perimeter of land-cover patch i, a_i is the area of the land-cover patch, and n is the number of patches within the landscape	Distribution characteristics of shape complexity
Number of patches (NP)	n_i	n_i is the number of patches for class i	Spatial fragmentation of land-cover patches
Splitting index (SPLIT)	$\dfrac{A^2}{\sum_{i=1}^{n} a_i^2}$	a_i is the area (m^2) of patch i A is the total landscape area (m^2)	Spatial fragmentation of land-cover patches, but with different sensitivities to NP

results, three different classifiers are implemented as the base classifiers, respectively, which aims at taking advantage of different classifiers and making full use of the labeled samples. In [43], Wang *et al.* first employed an improved M-training techniques in hyperspectral image classification. In this paper, three different classifiers are applied to optimize the performance of classifier by complementing each other, which can increase the final classification accuracy.

C. MULTIPLE FACTOR AUTHENTICATION AND LANDSCAPE METRICS

Landscape metrics can effectively differentiate the distribution of ground objects and comprehensively capture the overall patterns of landscapes. And landscape metrics can be defined as quantitative indices that describe the structures and spatial characteristics of a landscape. Most of the previous studies in landscape metrics have focused on monitoring urban landscape changes. Landscape metrics can be viewed as an effective method for land source planners who need to better understand. Policy makers make appropriate decisions about sustainable development by learning landscape metrics [44]. Many spatial landscape properties can be quantified by applying a series of metrics and eight commonly used landscape metrics are be introduced in table 1. Nowadays, landscape metrics captures much interests in remote sensing (RS) and geographic information systems (GIS). In this study, we focus on the analysis of the spatial characteristics of land cover patches, land cover classes and highlight

the value of spatial metrics in the study of hyperspectral image classification mages, both the dimensional features and redundancy between frequency bands are increased, which results in misclassification between spectrally similar classes. Moreover, the unlabeled samples are selected for refining the classifiers may result in low certainties. And per-pixel classification using spectral information alone is generally subject to the salt-and-pepper effect. Under this circumstance, multiple factor authentication has an effective ability to improve the classification accuracy. It has been demonstrated that multiple factor authentication can guarantee the reliability of the unlabeled samples in many respect and exploit the strengths of the individual algorithm and classifier.

This idea is derived from a collaborative manner framework for hyperspectral image classification [45]. In detail, an pseudolabel verification procedure based on active learning is performed for gradually enhancing the pseudolabeling accuracy and facilitating semisupervised learning. Simultaneously, the unlabeled samples that cannot be assigned with pseudolabels with high confidence are regarded as candidates in iteration process.

By assigning pseudolabel and multiple authentication, we can complement different algorithms and different classifiers for each other. Multiple authentication enables a collaborative labeling procedure by capture advantages of different modules to acquire more confident labeled pixels and thus improves the final classification performance.

III. PROPOSED METHODOLOGY

In this paper, we propose a new framework for hyperspectral image classification using spectral and spatial features. This novel learning landscape features semisupervised framework (LLFSF) combines different semisupervised methods and learn landscape feature in a collaborative process.

A. STEP 1: SUPERPIXEL-BASED NON-LOCAL SEGMENTATION ALGORITHM

If the spatial information of hyperspectral image is complex, we apply SLIC-based non-local superpixel segmentation to presort the original hyperspectral map, and the landscape features can be derived from the segmentation map. The simple linear iterative clustering (SLIC) algorithm not only considers the spectral distance but also calculates spatial distance in the iterative clustering process. The SLIC algorithm has been widely applied for superpixel segmentation of color images, because it has simple and effective characters. Specifically, this algorithm has only two tunable parameters P and ω_S, P is utilized to control the number of pre-segmented super-pixels, ω_S is applied to measure the relative weight of the spectral and spatial distances of each pixel to the center pixel. In our proposed segmentation algorithm, we apply the hexagonal grid described in [46] instead of the square grid used in the standard SLIC algorithm to generate the initial clusters We can see from figure 1, every side and every corner of the hexagon are shared by two and three hexagons, respectively. Compared with original SLIC with square, selecting a hexagonal gird for image segmentation has evident advantages is that it can sufficiently learn the surrounding spatial information. Each hexagon has more non-diagonal neighbors than a square, and hexagonal grids provide less distance distortion of boundary pixels [46]. The size of improved SLIC algorithm can be described with the width ω and height h, the spacing of adjacent hexagons can be represented by the horizontal distance *horiz* and the vertical distance *vert*. m^i and n^i are the central coordinates of the superpixel. row_i and col_i are the index of superpixel. Meanwhile, calculating the center of a hexagon is not very complicated than a square and it can be expressed as a simple matrix multiplication:

$$\begin{bmatrix} m^i \\ n^i \end{bmatrix} = \begin{bmatrix} vert & 0 \\ 0 & horiz \end{bmatrix} \begin{bmatrix} 2row_i - 1 \\ col_i \end{bmatrix} \quad (1)$$

Note that the expected spatial extent of the superpixel is an approximate hexagonal region, but in the allocation step, we search a similar pixel in the window with size $2\omega \times 2\omega$ around the central superpixel.

As the 3D hyperspectral image changing, the corresponding spectral distance measurement is updated to the spectral angle distance, which is an effective similarity measure of the spectral features. As for calculating the spectral distance, we also need to calculate the distance from each pixel to the cluster center (Euclidean distance) in the clustering process.
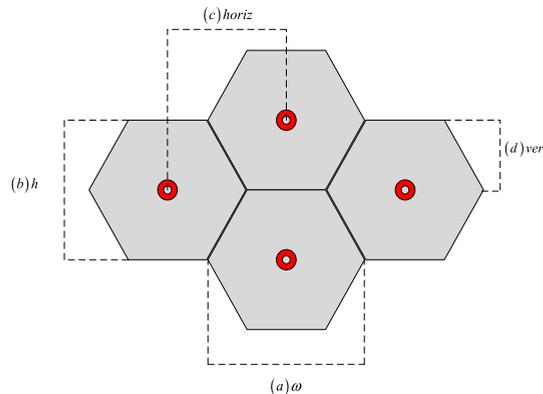
FIGURE 1. SLIC initial superpixel model.

We summarize the distance metrics as follows:

$$d_x = \arccos \left(\frac{x_j^T x^i}{\|x_j\|_2 \|x^i\|_2} \right) \quad (2)$$

$$d_{min} = \sqrt{(m^i - m_j)^2 + (n^i - n_j)^2} \quad (3)$$

$$D_j = \sqrt{d_x^2 + \left(\frac{d_{mn}}{\omega} \right)^2 \omega_s^2} \quad (4)$$

where x_i and x_j represent the average spectrum of all pixels in the ith and jth superpixel, respectively. (m_j, n_j) represents the coordinates of the pixel j, d_x represents the spectral distance of the pixel j to the cluster center, and d_{mn} represents the spatial distance of pixel j to the cluster center. Since the spatial distance d_{mn} changes more significantly than spectral distance d_x, the spatial distance is normalized by applying the hexagonal mesh width ω when calculating the total distance D_j. The ω_S as parameter can control the relative weight between spatial and spectral similarities, usually taken values in the range [0.1, 1]. With the increasing of parameter value, spatial similarity becomes more important. Inversely, When ω_S is tuned small value, spectral similarity is more significant. Since the spectral proximity is assumed to be more vital in measuring the similarity, ω_S has been set to 0.3 in our experiments. When the spatial-spectral information of the hexagons is similar, the proposed non-local segmentation algorithm will merge these hexagons into a bigger non-local area. We have set threshold values to integrate similar blocks into a bigger syncretic block. If the threshold value is bigger, the number of syncretic block is larger and syncretic block is smaller. On the contrary, when the threshold value is tuned to be small, it indicates that a syncretic block will contain more hexagons. The flowchart of the simple linear iterative clustering (SLIC) algorithm is shown in Figure 2.

The spatial feature calculation strategies can be regarded as a form of preprocessing before the classification, which improve the class separability through the addition of spatial features. We can make pre-judgment on the complex landscape composition and spatial configuration. From the figure 9, we can observe that SLIC-based non-local superpixel segmentation algorithm splits the original image into

FIGURE 2. SLIC algorithm flow chart.

Algorithm 1 Weighted Spatial-Spectral Double Layer SVM Algorithm

Input:

Initial training set: $L = \left\{ (x_i, y_i)_{i=1}^{l} \right\}$

Initial unlabeled data set: $U = \left(x_j \right)_{j=1}^{u}$

1. Calculate the average spatial-spectral value of K neighbor samples to replace the center sample.
2. Calculate the distance between center sample and neighbor samples and assign weighted to each sample.
3. Update the hyperspectral data.
4. When the confidence value of unlabeled samples is higher than confidence threshold value (CT), add these unlabeled samples into training set.
5. Update training set and test set.
6. Classify U by double layer SVM classifiers.

Output: Initial classification map

structure, which also indicates that it has more abundant spatial information. In term of these, we will pay more attention on learning spatial information in the following steps.

B. STEP 2: WEIGHTED SPATIAL-SPECTRAL DOUBLE LAYER SVM MODULE

To better characterize the complex scenes of hyperspectral images, we apply weighted spectral information strategy to preprocess data. To calculate the weight of a pixel located at x in the classified map, we extract the spatial-spectral information in a window of size=w', whose central pixel is x. We implement spectral similarity to calculate the difference that between the central sample and the surrounding K neighbor samples. Samples are more similar to the central sample will be given greater weight value. Specifically, we assign the weight by calculating Euclidean distance from adjacent pixels to the center pixel. The weight for pixel x_i can be expressed as ω_i. Then, we calculate confidence level of a pixel located at x in the classified map.

Confidence threshold can be applied to assess the reliability of samples. If this value is high, greater than threshold, the sample x_i is assigned to training set. Simultaneously, the training data set and the testing data set are updated by adding the samples with high confidence level. With Algorithm 1, we will obtain initial classification map by double layer spatial-spectral SVM classifiers (spectral feature classifier and spatial feature classifier). From the initial classification map, we can effectively quantify the spatial structures in terms of both composition and distribution, such as area, edge, shape, and aggregation.

C. STEP 3: MULTIPLE FACTOR AUTHENTICATION AND LEARNING LANDSCAPE FEATURE

Recently, landscape feature draws great interests in remote. Learning landscape feature aims to extract a new spatial feature that specifically describe the composition and configuration of the land-cover classes based on the contemporary

many non-local similar blocks. The number of non-local similar blocks can effectively represents the landscape density and complexity. Specifically, if a part of image has many non-local similar blocks, that indicates the spatial configuration is crowded and the spatial information is more complex. Furthermore, from the location of non-local similar blocks, we can precisely acquire the configuration information of land-cover. Pre-judgment based on original image can primarily realize the land-cover spatial composition and distribution, which contributes to the improvement of final accuracy. If this original image has complex spatial

classification map. In order to better comprehend the practical scenes of remote sensing images, learning landscape feature has been an indispensable link in our framework, which also contributes to enhance the efficacy of the overall framework.

Taking into account the initial classification map, we calculate the current five landscape features. They are Mean patch size (MPS), Standard deviation of area (AREA_SD), Largest patch index (LPI), Mean shape index (SHAPE_MN) and Splitting index (SPLIT). SPLIT represents the spatial fragmentation of land-cover patches, and SHAPE_MN indicates that the average measure of the shape complexity of each land-cover class. Take into account the SPLIT and SHAPE_MN, we select specific strategy for further classification based on current result. LPI can quantify the percentage of total landscape area comprised by the largest land-cover class, indicating the dominant land-cover patch and fragmentation of the image scene. And MPS are applied to describe the spatial structure from table 1, AREA_SD reflects the distribution and composition of the land-cover classes. Hence, according to the feedback provided by the metabolic landscape features, the classification results and model can be sustainably optimized. Generally speaking, different images have different performances in same feature. In step 2, the accuracy of each patch is divided into relatively high or low certainty values. In this step, the confidence threshold (CT) is used to determine which pixels should be further processed. A smaller value represents that only a small number of pixels are reliable. In this study, The patches with low threshold are considered for further processed by M-training modules. In addition, the patch with a high threshold means that more details and information should be preserved. And we also calculate the landscape features which are derived from initial classification map obtained by two layer SVM classifiers. Calculating the landscape features of a pixel located at the classification map, we extract the contextual geography and spatial structure information in a patch (of size $= p'$). It should be mentioned that, in our framework, the size of patch is not fixed, but according to the results of initial classification map. The landscape feature with metric m of class i for pixel x can then be expressed as $h_i(x, p', m)$. Therefore, all metrics for class i are denoted as:

$$h_i(x, p') = \left[h_i(x, p', 1), \ldots, h_i(x, p', m), \ldots, h_i(x, p', M) \right]$$

(5)

where M is the number of landscape metrics [28]. In this framework, for each certain hyperspectral data set, we can obtain initial classification map, temporary classification map and final classification map. The initial classification map is achieved by two layer SVM classifiers and each pixel obtain the initial class label.

Therefore, we select totally different hyperspectral image scene as the input in our experiment. For instance, Indian pines image has lots of grass, woods and many different kinds of crops. This data has complex and diversity classes. Moreover, these patches to be classified are close to each other in

space. Pavia university image contains fewer classes than last one. Beside it has lots of Meadows and trees, it also has Bare Soil and Bitumen. And the arrangement of the land-cover is not very crowded. Compared with Pavia university image, Pavia centre image has many pixels and these pixels are close to each other. It is a typical fragmented spatial pattern in our experiments.

It should be mentioned that the idea of multiple factor authentication is adopted in this step. In the overall framework, we have obtained two classification maps and a split image and assigned pseudolabels for unlabeled samples by two methods, respectively. Moreover, taking into account the arrangement and contextual information of the land-cover classes, we learn landscape features from the labeling space, which can reduce the salt-and-pepper noise in the classification map. From aforementioned procedure, we have implemented different methods to classify each unlabeled sample and the landscape features are updated for based on continuously updating classification map. From the perspective of multiple factor authentication, we investigate the landscape features to classify controversial patches that have distinguishing classification results in two algorithms. Landscape features effectively quantify the spatial structures in terms of both composition and configuration. Taking into the neighborhood information with respect to other land-cover classes can reduce the salt-and-pepper noise in the classification map.

D. STEP 4: M-TRAINING ALGORITHM

We have proposed a novel multiple-training (M-training) method for alleviating classification issues by automatically selecting informative training samples [43]. In M-training algorithm, the classification error rate is composed by the unlabeled sample error rate and the labeled sample error rate. To complement the characteristic of classifiers, classifiers system of the M-training method is diverse. We implement support vector machine (SVM), random forest (RF), and K-Nearest Neighbor (KNN) as base classifiers.

SVM is very suitable for the classification of remote-sensing images when the number of training samples is small. SVM is a supervised nonparametric statistical learning technique, which is not constrained to prior assumptions on the distribution of the input data. However, SVM has a limited ability to improve the classification accuracy, in terms of the edge areas. The K-Nearest Neighbor (KNN) algorithm is a simple classification algorithms in machine learning field. It often generates competitive results and has significant advantages over several other data mining methods. It chooses k training data nearest to the test data. The final class labels of test data are decided by majority voting. KNN is usually utilized as a pixel-wise classifier in these researches which relies on the optimal distance metric and feature space [28]. Random Forests (RF), as an ensemble learning technique, is increasingly applied in land-cover classification by applying multispectral and hyperspectral satellite sensor imagery and radar data [47], [48].

Algorithm 2 M-Training Algorithm

Input:

Initial training set: $L = \left\{ (x_i, y_i)_{i=1}^l \right\}$

Initial unlabeled data set: $U = (x_j)_{j=1}^u$

Initial iteration times: $t = 0$

the number of iterations: T

While $t \le T$:

Repeat:

1. Train classifiers C1, C2, C3, C4 by using initial training set L.

2. Choose a classifier as main classifier and others are assistant classifiers.

3. Use U as the test set. When the unlabled samples recieve same classifiaction results from one main classifier and three assistant classifiers, this unlabeled sample will be labled by classifiers and put it into new labeled data set $L_1(t)$, and $L_1(t) = \{x | x \in U, C_1(x) = C_2(x) = C_3(x) = C_4(x)\}$ (Similarly, when C2, C3, C4 as main classifier, respectively, we denote new labeled data set as $L_2(t)$, $L_3(t)$, $L_4(t)$, respectively.

4. If C1 as main classifier and $e_1(t)|L_1(t)| < e_1(t-1)|L_1(t-1)|$, update the labeled dataset and unlabeled data set. $L_1(t) = L_1(t-1) \cup L_1(t)$ If $e_1(t)|L_1(t)| \ge e_1(t-1)|L_1(t-1)|$, we randomly select samples as $L_1(t)$ to ensure $e_1(t)|L_1(t)| < e_1(t-1)|L_1(t-1)|$.

5. Update iteration times $t = t + 1$.

6. Until T rounds are reached.

Output: Trained classifiers

7. Input test samples into the trained classifiers, using majority vote strategy to obtain classification results.

It applies the majority vote to predict the class of a given observation. It is insensitive to noise points in training samples. With the limited training samples, the generalization ability of RF is substantially improved. Therefore, it will not be over-fitting when the size of the data set increasing. A RF consists of a combination of classifiers where each classifier contributes with a single vote for the assignation of the most frequent class to the input vector (x), $\hat{C}_{rf}^B = majorityvote \left\{ \hat{C_b}(x) \right\}_1^B$, where $\hat{C}_b(x)$ is the class prediction of the bth random forest tree. We can regard RF as a new concept of classifiers. Furthermore, when the RF makes a tree grow, it uses the best segmentation of a random subset of input features or predictive variables in the division of every node, instead of using the best split variables.

In our experiment, the four classifiers (denoted as C1, C2, C3, and C4) are initially trained by the labeled samples. Each classifier can be main classifier and the rest of classifiers are as assistant classifiers, respectively. In M-training, suppose we have a hyperspectral image with m samples $D = \{x_1, x_2, \ldots, x_m\}$ of d dimensions. In the experiment, we calculate the classification error rate of both the unlabeled

samples and the labeled samples, respectively. The error rate of unlabeled samples $e_i(U)$ are obtained by vote strategy. The $e_i(t)$ represents the upper limit of the classification error rate for auxiliary classifier at tth iteration. We denote $L(t)$ as newly labeled data set at tth iteration.

$$e_i(U) = \frac{(n_i(u) - k_i)}{n_i(u)} \quad (6)$$

$$e_i(t) = \lambda * e_i(L) + (1 - \lambda) * e_i(U) \quad (7)$$

where $n_i(u)$ is the total number of unlabeled samples. k_i denotes the number of samples, and these samples are labeled as same class from assistant classifiers. $\lambda (\lambda = 0, 0.1, 0.2 \ldots 0.9, 1)$ is weighting parameter that tunes the tradeoff between $e_i(t)$ and $e_i(U)$. When a sample in U obtains the same result from the assistant classifiers and main classifier, it can be assigned with confident pseudolabels to improve the training model in the next time. Moreover, when the C1 as main classifier, and $e_1(t)|L_1(t)| < e_1(t-1)|L_1(t-1)|$, the original labeled data set is expanded as $L \cup L(t)$ according to iteration. When $e_1(t)|L_1(t)| \ge e_1(t-1)|L_1(t-1)|$, we will randomly select samples from $L(t)$ and these selected samples are formed into $S(S = L_1(t))$. Simultaneously, we apply $L_1(t) = L_1(t-1) \cup L_1(t)$ to retrain classifiers C1. Then, beginning the iteration loop, we label only limited samples to train effective classifiers. When the iteration is finished, we apply the majority vote to predict the class of a given observation.

Why we choose the weighted spatial-spectral double layer SVM module and M-training algorithm as the main classification methods in this framework? The main reason as following, first, the Weighted spatial-spectral double layer SVM module obtain excellent classification performance when given a certain number of labeled samples. If the initial labeled pixels is too few, which will inevitably lead to many classification errors in some classes. Next, concerning the edge regions, tri-training outperforms than other methods are introduced in published literature [28]. M-training is an improved algorithm based on tri-training. M-training is also superior to others on the edge regions and has better classification performance on land- cover with similar spectral. Hence, we choose these algorithms to complement with each other. After this step, we integrate the patches with the optimized results and obtain a final classification map. The flowchart of the overall framework (LLFSF) is shown in figure 3.

IV. DATASET DESCRIPTION AND DESIGN OF EXPERIMENT

A. DATASET DESCRIPTION

In the experiments, three hyperspectral remote-sensing images were employed as benchmark data sets. The details of the three data sets are as follows. The first hyperspectral data set is Indian pines data set. In June 1992, the NASA AVIRIS image was acquired over the Indian pines agricultural site in northwestern Indiana. AVIRIS is a sophisticated
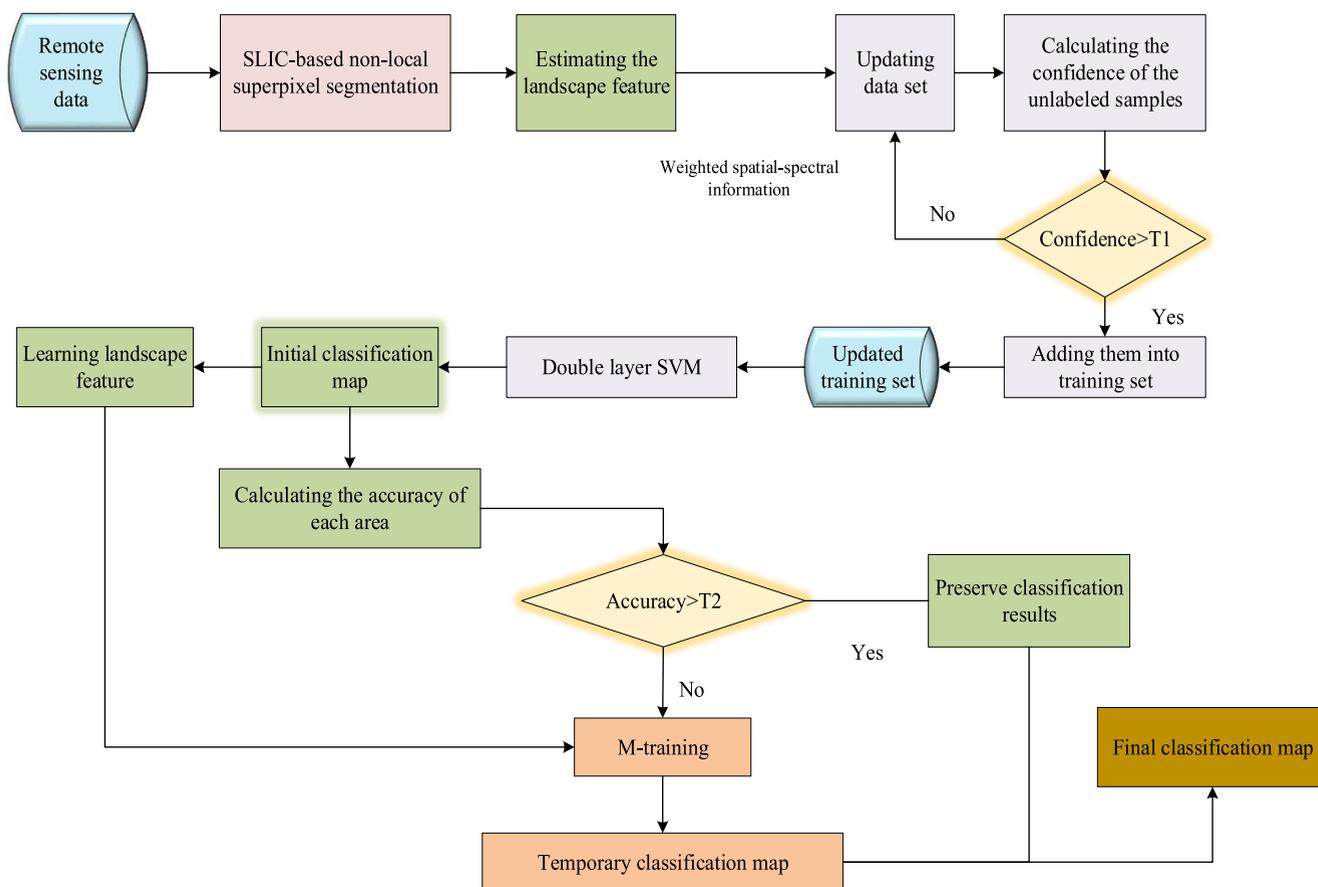
FIGURE 3. Flowchart of the learning landscape features semisupervised framework (LLFSF).
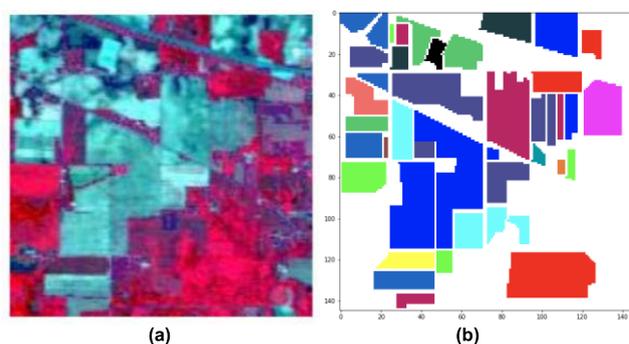
FIGURE 4. False-color composite image of Indian Pines data set and color map of ground truth. (a) False-color image. (b) Ground truth.

TABLE 2. Numbers of samples for the corresponding classes of the Indian Pines data set.

Class Label	Class Name	#samples
C1	Alfalfa	46
C2	Corn-notill	1428
C3	Corn-mintill	830
C4	Corn	237
C5	Grass-Pasture	483
C6	Grass-Trees	730
C7	Grass-Pasture-mowed	28
C8	Hay-windrowed	478
C9	Oats	20
C10	Soybeans-notill	972
C11	Soybeans-mintill	2455
C12	Soybean-cleantill	593
C13	Wheat	205
C14	Woods	1265
C15	Building-Grass-Tress	386
C16	Stone-Steel-Towers	93

optical sensor system including a number of major subsystems, components, and characteristics. This data set contains 145×145 pixels, with 220 spectral bands covering the range of 375–2200 μm [49]. The corresponding spatial resolution is approximately 20 m. The data set contains 16 classes representing the different classes of land cover, and 10249 available samples.

Pavia university data set and Pavia Centre data set were acquired by the Reflective Optics System Imaging Spectrometer (ROSIS) instrument in 2001, covering the city

of Pavia, Italy. Pavia university data set is centered at the University of Pavia, with a size of 610×340 pixels. It comprises 115 spectral channels in the wave-length range from 0.43 to 0.68 um with a spatial resolution of 1.3 m, and 103 channels are used in the experiment after removing noise and water absorption bands. This data set contains nine classes

TABLE 3. Numbers of samples for the corresponding classes of the Pavia university data set.

Class Label	Class Name	#sample
C1	Asphalt	6631
C2	Meadows	18649
C3	Gravel	2099
C4	Trees	3064
C5	Painted metal sheets	1345
C6	Bare Soil	5029
C7	Bitumen	1330
C8	Self-Blocking Bricks	3682
C9	Shadows	947

TABLE 4. Numbers of samples for the corresponding classes of the Pavia centre data set.

Class Label	Class Name	#sample
C1	Water	65278
C2	Trees	6508
C3	Meadows	2905
C4	Self-Blocking Bricks	2140
C5	Bare soil	6549
C6	Asphalt	7585
C7	Gravel	7287
C8	Tiles	3122
C9	Shadows	2165
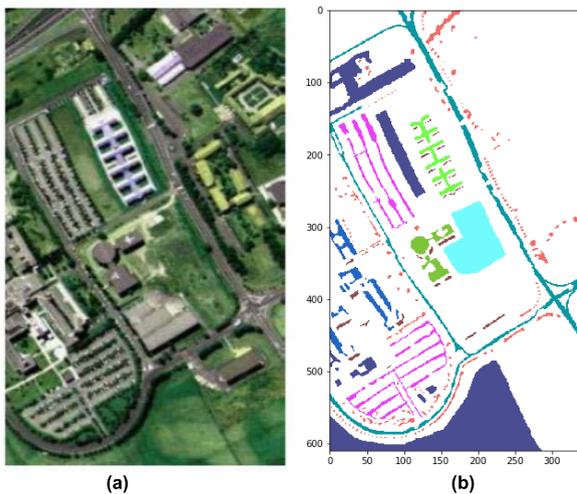
FIGURE 5. False-color composite image of Pavia University data set and color map of ground truth. (a) False-color image. (b) Ground truth.
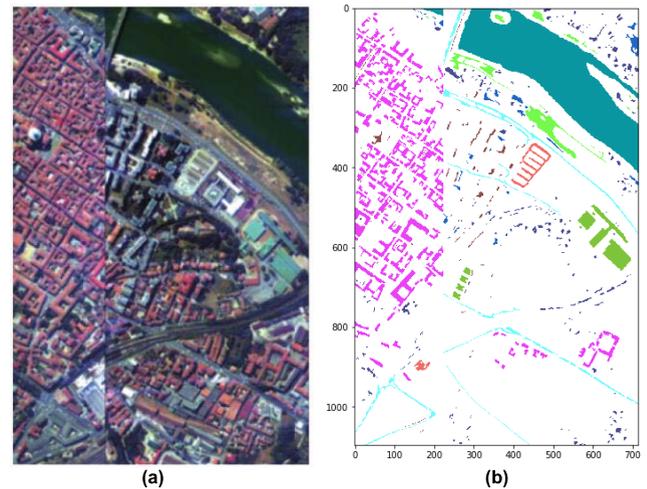
FIGURE 6. False-color composite image of Pavia Centre data set and color map of ground truth. (a) False-color image. (b) Ground truth.

representing the different types of land-cover, and there are 42776 available samples [50].

Pavia Centre data set is also applied in our paper. After noise and water absorption bands are removed, the number of spectral bands is 102 for Pavia Centre. The geometric resolution is 1.3 meters. Pavia Centre is a 1096×1096 pixels image, but some of the samples in images contain no information and have to be discarded before the analysis. Because the Pavia Centre data set is too large, we select a part of the whole image as input data. The selected image is a 548×715 pixels image. This data set contains nine classes representing the different types of land cover.

B. DESIGN OF EXPERIMENTS

In the experiments, for every algorithm, ten runs were executed on each image with different initial labeled data. Support vector machine (SVM) is a supervised nonparametric statistical learning technique. The parameters of SVM were set as kernel = radial basis function (RBF). The LibSVM and RF toolbox were adopted. There are two parameters for the SVM classifier, i.e., the regularization parameter P and the Gaussian kernel parameter G, which are usually selected via cross validation. But in this paper, in order to be efficient

and ensure the precision, we adopted the same parameters settings. As for KNN in M-training algorithm, we set $K = 3$.

As for weighted spatial-spectral double layer SVM module. In Indian pines data set, we first randomly select 5% samples from each class as initial labeled training data set. Then, we will add about 1% unlabeled samples with high confidence threshold into training set. In this section, we donate confidence threshold as T1 and (T1 $\geq$ 0.95). As for the subsequent M-training algorithm, we randomly divide the total available data into two parts: 25% for training and 75% for testing. The training data contains the unlabeled samples and the labeled samples. We randomly select 5% samples in each class as the initial labeled data. The unlabeled samples receive the same labels form four classifiers will be added into training set.

In Pavia university data set, as for weighted spatial-spectral double layer SVM module, we first randomly select 1% labeled samples from each class as initial training set. Then we add unlabeled samples with high confidence into training set. (T1 $\geq$ 0.95) As for the subsequent M- algorithm, we randomly divide the total available data into two parts: 25% for training and 75% for testing. We randomly selected 1% samples in each class as the initial labeled data. The unlabeled
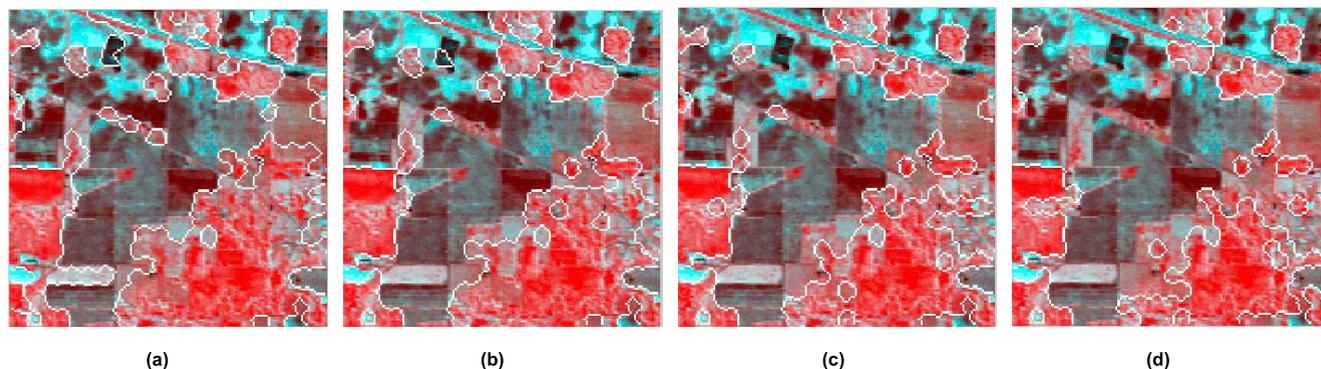
FIGURE 7. SLIC-based non-local superpixel segmentation results based on Indian pines data set with different threshold. (a) threshold=0.150, (b) threshold=0.175 (c) threshold=0.200 (d) threshold=0.225.
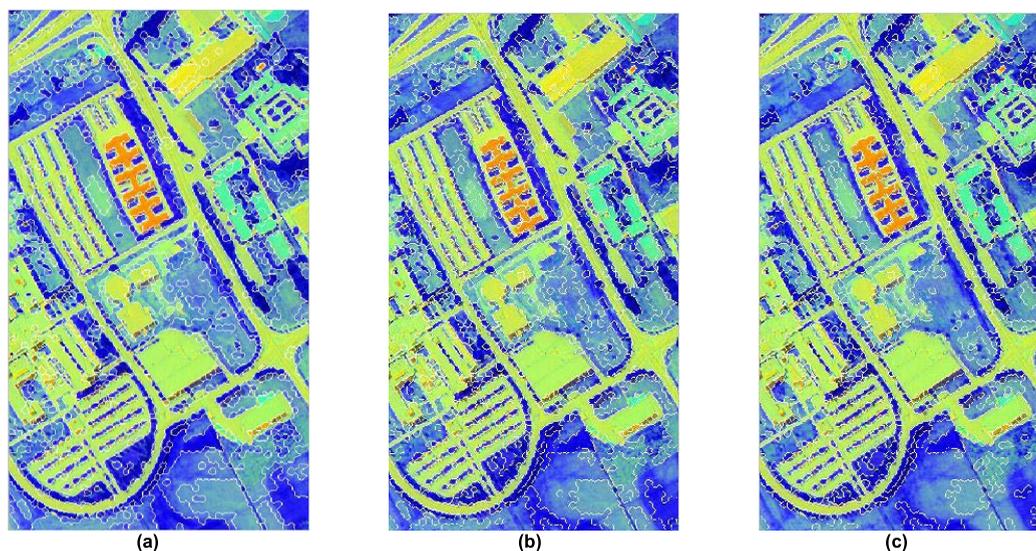
FIGURE 8. SLIC-based non-local superpixel segmentation results based on Pavia university data set with different threshold. (a) threshold=0.18, (b) threshold=0.21 (c) threshold=0.24.

samples receive the same labels form four classifiers will be added into training set.

In Pavia centre data set, as for weighted spatial-spectral double layer SVM module, we first randomly select 0.1% labeled samples from each class as initial training set. Then we add the unlabeled samples with high confidence (T1 $\geq$ 0.995) into training set. Because the number of Pavia centre data set is very large, we try to apply a small sized number of labeled samples to obtain better performance. In M-training algorithm, we also randomly divide the total available data into two parts: 25% for training and 75% for testing. And for training set, we randomly select 0.1% samples in each class as the initial labeled data. The unlabeled samples receive the same labels form four classifiers will be added into training set. All experiments were conducted on an LENOVO Z50 laptop with the NVDIA GeForce 840M graphical processing unit.

V. RESULTS AND ANALYSIS
We apply a very small sized number of labeled samples and limited training samples to validate the effectiveness of

proposed framework. We firstly utilize the SLIC-based non-local superpixel segmentation (SLIC-NL) algorithm to preprocess the image. SLIC-NL can initially assess the complexity of each land-cover class and area distribution of different land-cover classes, which can attribute to the learning landscape features. We can see the segmentation results with different threshold in figures 7-9. In the experiments, weighted spatial-spectral double layer SVM algorithm (WSS-DSVM), tri-training algorithm and M-training algorithm were considered for a comparative analysis. The general comments regarding the results are summarized as follows: From table 7, the classification accuracy of LLFSF is superior to other algorithms, which indicates that our proposed framework has the potential to improve the classification performance. Furthermore, the results can also be confirmed by figure 10, where learning-landscape achieves the highest overall accuracies in the Indian pines data set. We can also observe that tri-training algorithm and M-algorithm obtain worse classification accuracy when they classify 16 classes land-covers. They also have many noises and misclassifications on the classification maps.
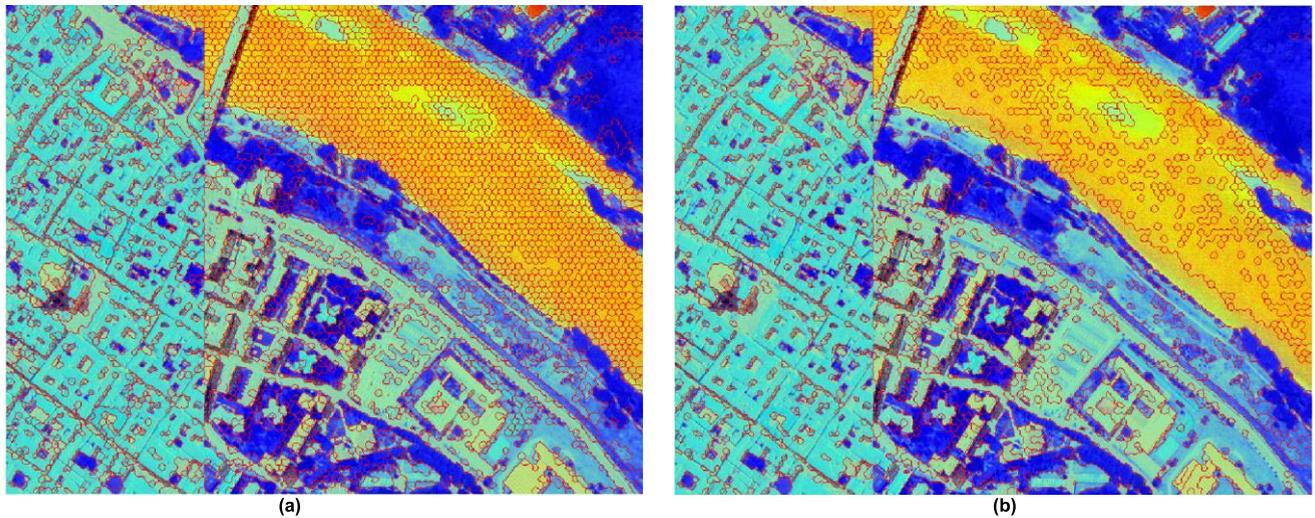
FIGURE 9. SLIC-based non-local superpixel segmentation results based on Pavia centre data set with different threshold. (a) threshold=0.16, (b) threshold=0.24.

TABLE 5. The landscape features of the Indian pines data set.

Landscape feature	Calculated value							
Mean patch size(MPS)	Patch 1 232.00	Patch 2 483.60	Patch 3 136.83	Patch 4 230.00	Patch 5 112.00	Patch 6 118.80	Patch 7 99.00	Patch 8 56.00
	Patch 9 20.00	Patch 10 242.00	Patch 11 480.00	Patch 12 115.75	Patch 13 198.00	Patch 14 112.33	Patch 15 165.00	Patch 16 95.00
Standard deviation of area (AREA_SD)	246.4274							
Largest patch index (LPI)	0.1130							
Mean shape index SHAPE_MN	2.0048							
Splitting index (SPLIT)	20.2840							

TABLE 6. The landscape features of the Pavia centre data set.

Landscape feature	Calculated value								
Mean patch size(MPS)	Patch 1	Patch 2	Patch 3	Patch 4	Patch 5	Patch 6	Patch 7	Patch 8	Patch 9
	3.2986	0.0029	0.0100	0.2140	0.0329	0.0956	0.0594	0.0215	0.0049
Standard deviation of area(AREA_SD)	$3.3523e+03$								
Largest patch index (LPI)	0.5528								
Mean shape index SHAPE_MN	1.5124								
Splitting index (SPLIT)	3.2378								

From the classification map and confusion matrix, we can find that Soybean-clean, Woods, Grass-trees and corn-notill are seriously misclassified. Although WSS-DSVM has better classification performance than M-training and tri-training algorithms, especially in Grass-trees and Soybean-mentill. It also has many salt-and-pepper noise in homogenous regions because incorrect labels may be added during the semisupervised learning.

TABLE 7. Numbers of samples for the corresponding classes of the Indian Pines data set.

Performance	Tri-training	M-training	WSS-DSVM	LLFSF
OA (%)	67.64	71.80	92.04	95.12
AA (%)	53.19	61.42	85.91	90.27
Kappa	0.6258	0.6756	0.9093	0.9444
Computational time (s)	223.21	128.47	60.68	139.75

TABLE 8. Numbers of samples for the corresponding classes of the PAVIA university data set.

Performance	Tri-training	M-training	WSS-DSVM	LLFSF
OA (%)	81.22	78.74	95.52	98.46
AA (%)	75.71	64.89	94.70	96.39
Kappa	0.7412	0.7025	0.9404	0.9796
Computational time (s)	962.61	540.75	341.87	372.03

TABLE 9. Numbers of samples for the corresponding classes of the PAVIA centre data set.

Performance	Tri-training	M-training	WSS-DSVM	LLFSF
OA (%)	94.07	93.19	97.98	99.09
AA (%)	79.81	75.52	93.75	94.88
Kappa	0.9050	0.8910	0.9557	0.9732
Computational time (s)	4053.54	2312.87	1347.51	1899.57

(a)   (b)   (c)   (d)
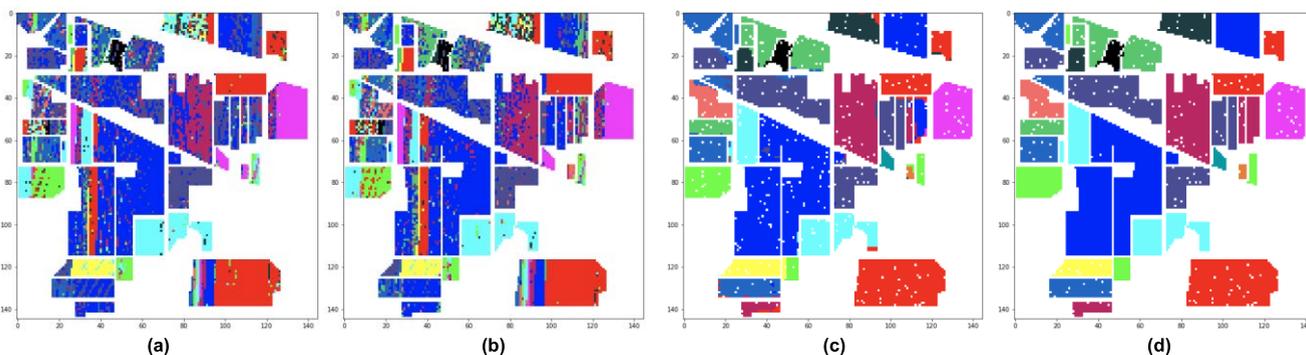
FIGURE 10. Comparison of the final classification map of different frameworks on the Indian Pines dataset. (a)Tri-training; (b)M-training; (c) WSS-DSVM; (d) LLFSF.

Table 8 shows the classification results of the comparative methods on the Pavia university data set. Compared to M-training, tri-training and WSS-DSVM, the average accuracy improvements achieved by (LLFSF) are 19.72%, 17.74%, 2.94%, respectively. From the figure 11, in terms of the overall accuracy, we can find that M-training has worse classification performance than WSS-DSVM.

All the Bare soil are almost misclassified as Meadows. But we can also observe that M-training algorithm has more excellent classification ability on some specific classes, such as Meadows and Asphalt. Our proposed framework has the best performance among the compared algorithms. LLFSF applies less computation time to obtain best classification results. Pavia centre data set has a large amount of samples,

and the structural information of geospatial objects is complex and redundant.

Moreover, the separability of land-cover is higher than other data sets utilized in this paper. We apply a very small sized number of labeled samples and limited training samples to validate the effectiveness of proposed framework. We firstly utilize the SLIC-NL algorithm to preprocess the image, which can attribute to the learn landscape features.

After we obtain the initial classification map from WSS-DSVM algorithm, we begin to re-learn the complexity of each land-cover class and area distribution of different land-cover classes, which can contribute to the final classification results. We can also see form table 9, our proposed framework can obtain best classification results with a very small number
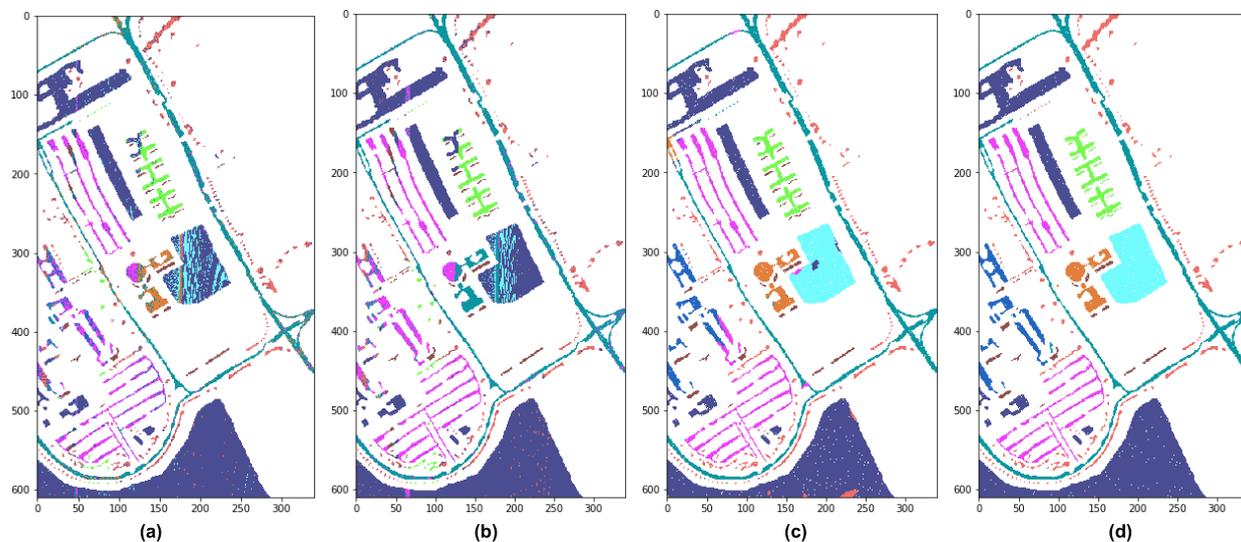
FIGURE 11. Comparison of the final classification map of different frameworks on the Pavia university data set. (a)Tri-training; (b)M-training; (c) WSS-DSVM; (d) LLFSF.
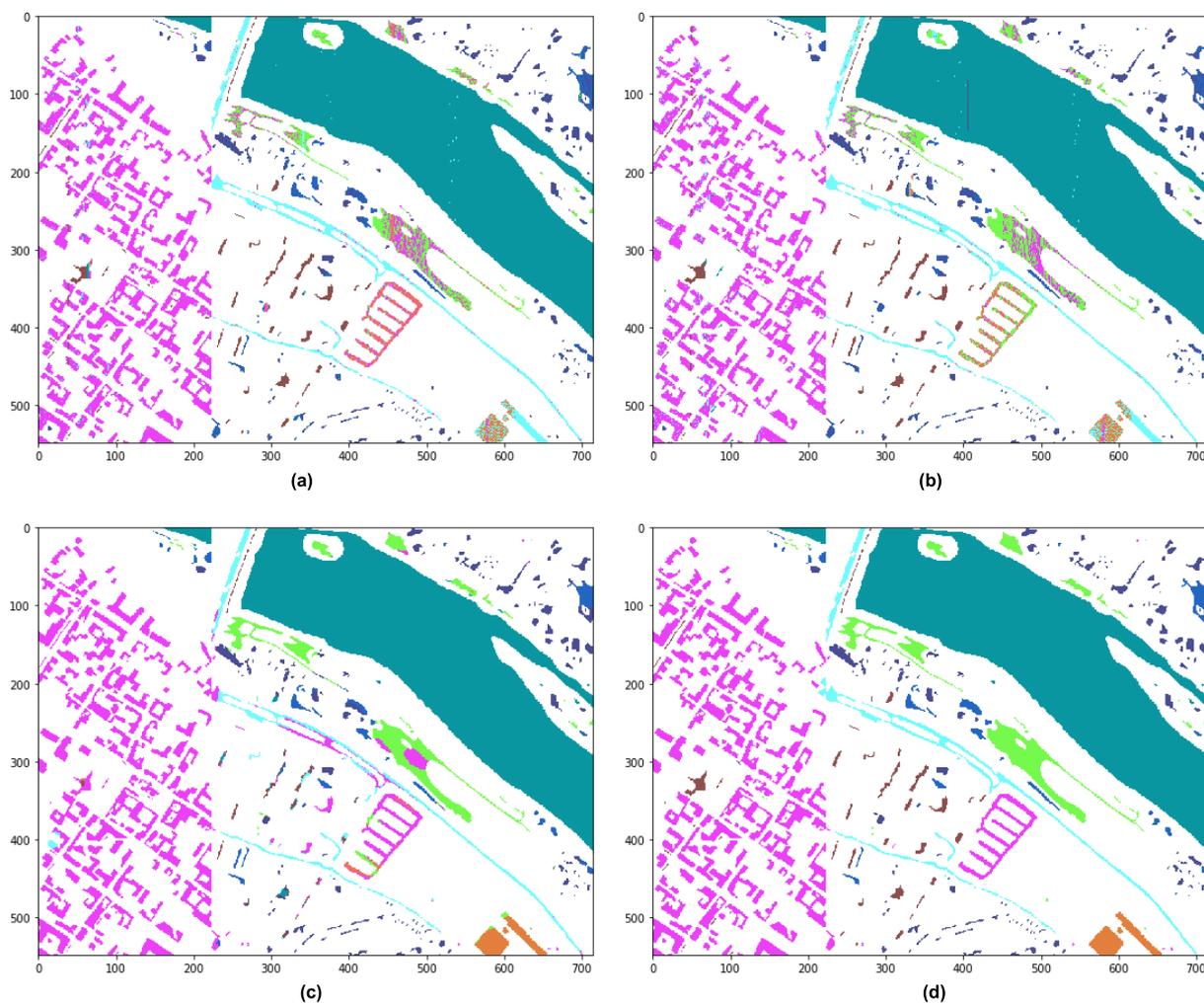
FIGURE 12. Comparison of the final classification map of different frameworks on the Pavia centre dataset. (a)Tri-training; (b)M-training; (c) WSS-DSVM; (d) LLFSF.
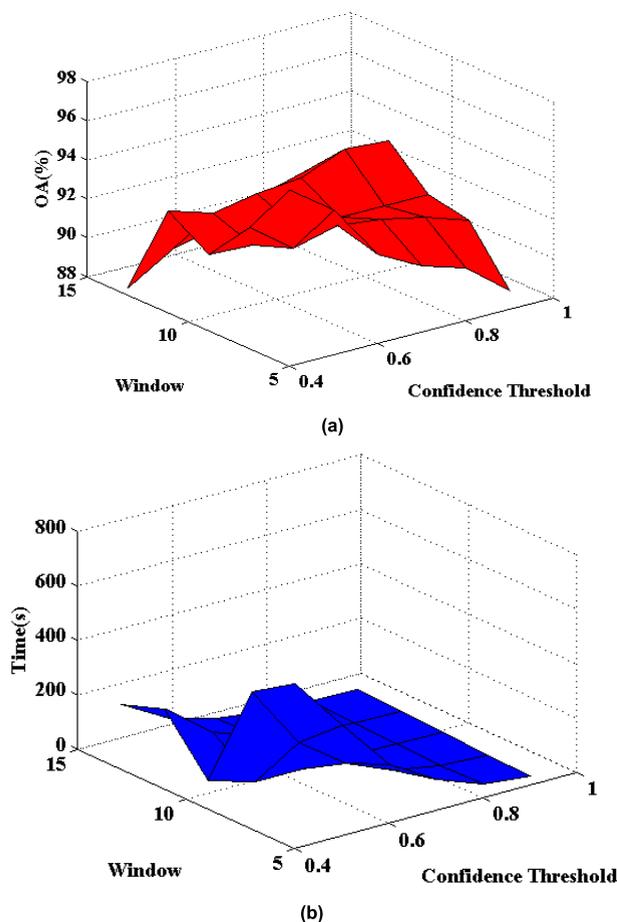
FIGURE 13. Analysis of the parameter sensitivity with the Indian pines data set for the weighted spatial-spectral double layer SVM module. (a) Overall accuracy (OA); (b) Running time.

FIGURE 14. Analysis of the parameter sensitivity with the Pavia university data set for the weighted spatial-spectral double layer SVM module. (a) Overall accuracy (OA); (b) Running time.

of labeled samples. Compared to M-training and tri-training, the average accuracy improvements achieved by LLFSF are 5.02–5.9%, respectively. Figure 12 has shown the classification maps obtained with the different algorithms. LLFSF is regarded as statistically better than the other compared algorithms.

The experiments of three data sets can confirm the efficacy of the proposed LLFSF. The classification map of LLFSF appears clearly, showing the efficiency of reducing the salt-and-pepper noise. It has been demonstrated that the proposed framework can learn the spatial configuration from the current classification map. Therefore, it is effective at exploiting the representative information and capturing the overall patterns from the unlabeled data.

PARAMETER ANALYSIS

The threshold parameter sensitivity analysis for two layer SVM system with the Indian pines data set are displayed in figure 13. In double layer SVM method, the size of windows and the value of confidence threshold are vital to the classification results. From figure 13, we can observe that the size of windows in a range of 5-13 and the value of confidence threshold in a range of 0.9-0.4 can result different

performances. For instance, when the size of windows is 5 and the value of confidence is 0.4, we can obtain the maximum OA value, that is 97.00%. On the contrary, the minimum OA value is 88.98%, it can be obtained, when the size of windows is 5, and the value of confidence threshold is 0.9.

Actually, when we obtain the maximum OA value, the time consumption is very high and the size of training set is very large. According to double two layer SVM method, we choose candidate samples with high confidence threshold as training set. If we set the value of confidence threshold is small, which will add a large number of unlabeled samples into training set. This can greatly increase the cost of data collection. So we should select the high confidence threshold to decrease time consumption. And we can also find that when the confidence threshold is the same, when the window size is equal to 11, the overall accuracy is higher than other parameter setting. But when the window size is 9, the overall accuracy is more stable and time consumption is more shorter. Considering the tradeoff between computational burden and performance of the classification model, in our experiment, we set the confidence threshold is 0.95 and window size is 9.

(a)

(b)
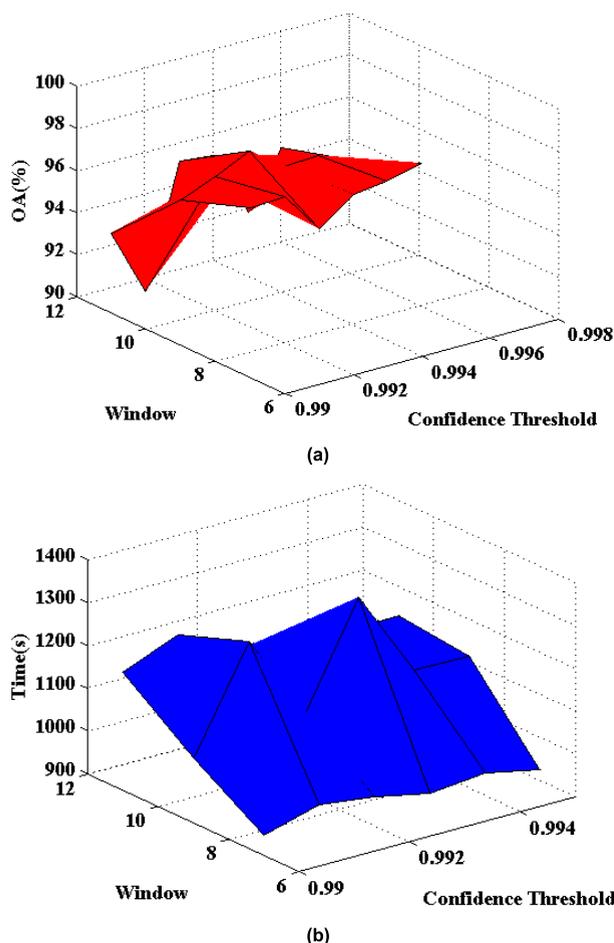
FIGURE 15. Analysis of the parameter sensitivity with the Pavia centre data set for the weighted spatial-spectral double layer SVM module. (a) Overall accuracy (OA); (b) Running time.

In Pavia university data set, from figure 14, we can observe that the size of windows in a range of 7-11 and the value of confidence threshold in a range of 0.85-0.95. It should be also mentioned that when the size of window is bigger than 11, the computational costs become very huge. So the biggest window size is 11 in our experiment. The maximum OA is 97.98%, and it can be obtained when the size of windows and the value of confidence threshold are set as 7 and 0.9, respectively. When the size of windows is 11 and the value of confidence threshold is 0.95, we can obtain the minimum OA, which is 95.52%.

The interesting phenomenon is that when the window size is 9, with the increasing of the confidence threshold, the overall accuracy doesn't change. And the running time also maintains same value. In order to reveal the superiority of the our LLFSF, we choose the "worst parameters" in the experiment. We set the confidence threshold is 0.95 and window size is 11. This is a understandable setting, since we want to implement the least the training samples and time consumption to obtain better classification. Although these parameters make the accuracy is a little worse than other parameters, we can utilize the following steps to enhance

the classification performance with a small sized number of training samples.

In Pavia centre data set, from figure 15, we can observe that the size of windows in a range of 7-11 and the value of confidence threshold in a range of 0.990-0.995 can generate different results. For instance, when the size of windows and the value of confidence threshold are set as 9 and 0.992, respectively, the maximum OA 98.36% can be obtained. Conversely, the minimum OA is 90.60%, when the size of windows and the value of confidence threshold are set as 11 and 0.991, respectively. We can also observe that when the size of window is 7, the classification performance is more excellent than other parameters. Hence, we select the window size is 7. In order to use the less training samples, we set the the confidence threshold is 0.995. In these parameters, we can obtain more approving results in less computational costs and training samples.

VI. CONCLUSION

This paper applies limited labeled samples, informative unlabeled samples and landscape features to achieve the better classification results. We investigate a novel framework that integrates two semisupervised learning algorithms and landscape feature learning in a collaborative manner for hyperspectral image classification. First, we implement a novel SLIC-based non-local superpixel segmentation to initially learn the landscape feature. Then, we apply a weighted spatial-spectral double layer SVM module to initially classify the image. Simultaneously, we apply this initial classification map to relearn landscape features, which can better quantize land-cover composition and spatial information. In next step, we put the data of patches with lower accuracy into M-training algorithm for further classification. The proposed framework (LLFSF) flexibly combines the strengths of weighted spatial-spectral double layer SVM module and M-training via the classification certainties. Moreover, LLFSF learns the landscape features via landscape metrics to quantify both the land-cover composition and spatial configuration and improve the classification model. It has been shown that the effectiveness of the LLFSF in the different characteristics remote sensing images. Through Multi-classifier system (M-training) and weighted spatial-spectral double layer SVM module, LLFSF can also avoid many classification errors obtained by inaccurate classifiers when given too few initial labeled samples and limited training samples. We also learn the spatial features which are derived from the initial classification map, and the classification result is gradually optimized and updated according to the feedback provided by the landscape features and semisupervised learning algorithms.

The experimental results obtained with Indian pines farmland hyperspectral data, Pavia university hyperspectral data and the center of Pavia city. The different and representative data sets can sufficiently demonstrate that LLFSF is able to significantly increase the classification accuracy and substantially reduce labeling cost and computational cost.

REFERENCES

[1] G. Lu and B. Fei, "Medical hyperspectral imaging: A review," *J. Biomed. Opt.*, vol. 19, no. 1, p. 010901, 2014.

[2] Y. Wang, G. Chen, and M. Maggioni, "High-dimensional data modeling techniques for detection of chemical plumes and anomalies in hyperspectral images and movies," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 9, pp. 4316–4324, Sep. 2016.

[3] L. Zhang, L. Zhang, D. Tao, and X. Huang, "On combining multiple features for hyperspectral remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 3, pp. 879–893, Mar. 2012.

[4] J. Plaza, R. Pérez, A. Plaza, P. Martínez, and D. Valencia, "Mapping oil spills on sea water using spectral mixture analysis of hyperspectral image data," *Proc. SPIE*, vol. 5995, pp. 79–86, Nov. 2005.

[5] H. Shen, X. Meng, and L. Zhang, "An integrated framework for the spatio–temporal–spectral fusion of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7135–7148, Dec. 2016.

[6] W. Sun, J. Ma, G. Yang, B. Du, and L. Zhang, "A Poisson nonnegative matrix factorization method with parameter subspace clustering constraint for endmember extraction in hyperspectral imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 128, pp. 27–39, Jun. 2017.

[7] B. Demir, C. Persello, and L. Bruzzone, "Batch-mode active-learning methods for the interactive classification of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 3, pp. 1014–1031, Mar. 2011.

[8] L. Yang, S. Yang, P. Jin, and R. Zhang, "Semi-supervised hyperspectral image classification using spatio-spectral Laplacian support vector machine," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 3, pp. 651–655, Mar. 2014.

[9] 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.

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

[11] T. V. N. Prabhakar, G. Xavier, P. Geetha, and K. P. Soman, "Spatial preprocessing based multinomial logistic regression for hyperspectral image classification," *Procedia Comput. Sci.*, vol. 46, pp. 1817–1826, 2015.

[12] Z. Zhong, J. Li, Z. Luo, and M. Chapman, "Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 847–858, Feb. 2018.

[13] B. Liu, X. Yu, P. Zhang, A. Yu, Q. Fu, and X. Wei, "Supervised deep feature extraction for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 1909–1921, Apr. 2018.

[14] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient K-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.

[15] C. Cariou and K. Chehdi, "Unsupervised nearest neighbors clustering with application to hyperspectral images," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 6, pp. 1105–1116, Sep. 2015.

[16] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, Jun. 2014.

[17] D. Hong, N. Yokoya, N. Ge, J. Chanussot, and X. X. Zhu, "Learnable manifold alignment (LeMA): A semi-supervised cross-modality learning framework for land cover and land use classification," *ISPRS J. Photogram. Remote Sens.*, vol. 147, pp. 193–205, Jan. 2019.

[18] H. Wu and S. Prasad, "Semi-supervised deep learning using pseudo labels for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 27, no. 3, pp. 1259–1270, Mar. 2018.

[19] Y. Shao, N. Sang, and C. Gao, "Spatial and class structure regularized sparse representation graph for semi-supervised hyperspectral image classification," *Pattern Recognit.*, vol. 81, pp. 81–94, Sep. 2018.

[20] W. Di and M. M. Crawford, "Active learning via multi-view and local proximity co-regularization for hyperspectral image classification," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 3, pp. 618–628, Jun. 2011.

[21] T. Munkhdalai, M. Li, T. Kim, O.-E. Namsrai, S.-P. Jeong, J. Shin, and K. H. Ryu, "Bio named entity recognition based on co-training algorithm," in *Proc. Int. Conf. Adv. Inf. Netw. Appl. Workshops*, Mar. 2012, pp. 857–862.

[22] B. Fichtelmann, E. Borg, and K. P. Guenther, "Adaption of a self-learning algorithm for dynamic classification of water bodies to MERIS data," *Lect. Notes Comput. Sci.*, vol. 9158, no. 1905, pp. 177–192, 2015.

[23] Y. Cui, G. Song, X. Wang, Z. Lu, and L. Wang, "Semisupervised classification of hyperspectral images based on tri-training algorithm with enhanced diversity," *J. Appl. Remote Sens.*, vol. 11, no. 4, p. 1, 2017.

[24] L. Yu and L. He, "Spectral–spatial graph-based semi-supervised hyperspectral image classification," in *Proc. IEEE 4th Int. Conf. Comput. Commun. (ICCC)*, Chengdu, China, Dec. 2018, pp. 1736–1740.

[25] X. Li, X. Lyu, Y. Tong, S. Li, and D. Liu, "An object-based river extraction method via optimized transductive support vector machine for multi-spectral remote-sensing images," *IEEE Access*, vol. 7, pp. 46165–46175, 2019.

[26] Y. Li, Y. Zhang, X. Huang, and A. L. Yuille, "Deep networks under scene-level supervision for multi-class geospatial object detection from remote sensing images," *ISPRS J. Photogramm. Remote Sens.*, vol. 146, pp. 182–196, Dec. 2018.

[27] B. Kellenberger, D. Marcos, S. Lobry, and D. Tuia, "Half a percent of labels is enough: Efficient animal detection in UAV imagery using deep CNNs and active learning," *IEEE Trans. Geosci. Remote Sens.*, to be published. doi: 10.1109/TGRS.2019.2927393.

[28] X. Han, X. Huang, J. Li, Y. Li, M. Y. Yang, and J. Gong, "The edge-preservation multi-classifier relearning framework for the classification of high-resolution remotely sensed imagery," *ISPRS J. Photogram. Remote Sens.*, vol. 138, pp. 57–73, 2018.

[29] D. A. Landgrebe, *Signal Theory Methods in Multispectral Remote Sensing*. New York, NY, USA: Wiley, 2003.

[30] L. Zhang, Q. Zhang, B. Du, X. Huang, Y. Y. Tang, and D. Tao, "Simultaneous spectral-spatial feature selection and extraction for hyperspectral images," *IEEE Trans. Cybern.*, vol. 48, no. 1, pp. 16–28, Jan. 2018.

[31] H. Li, C. Li, C. Zhang, Z. Liu, and C. Liu, "Hyperspectral image classification with spatial filtering and $\ell_{2,1}$ norm," *Sensors*, vol. 17, no. 2, p. 314, 2017.

[32] L. Xu and J. Li, "Bayesian classification of hyperspectral imagery based on probabilistic sparse representation and Markov random field," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 4, pp. 823–827, Apr. 2014.

[33] M. Khodadadzadeh, J. Li, A. Plaza, H. Ghassemian, J. M. Bioucas-Dias, and X. Li, "Spectral–spatial classification of hyperspectral data using local and global probabilities for mixed pixel characterization," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 10, pp. 6298–6314, Oct. 2014.

[34] H. Song and Y. Wang, "A spectral-spatial classification of hyperspectral images based on the algebraic multigrid method and hierarchical segmentation algorithm," *Remote Sens.*, vol. 8, no. 4, p. 296, 2016.

[35] A. Soltani-Farani, H. R. Rabiee, and S. A. Hosseini, "Spatial-aware dictionary learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 1, pp. 527–541, Jan. 2015.

[36] G. Zhang and X. Jia, "Simplified conditional random fields with class boundary constraint for spectral-spatial based remote sensing image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 5, pp. 856–860, Sep. 2012.

[37] C. Ying, W. Heng, and Z. Haifeng, "Superpixel-based local collaborative sparse unmixing for hyperspectral image," *J. Appl. Remote Sens.*, vol. 13, no. 1, pp. 1–13, 2019.

[38] B. Baassou, M. He, S. Mei, and Y. Zhang, "Unsupervised hyperspectral image classification algorithm by integrating spatial-spectral information," *Proc. Int. Conf. Audio Lang. Image Process.*, 2012, pp. 610–615.

[39] H. Hong and Z. Xin-Lei, "Hyperspectral image classification with combination of weighted spatial-spectral and KNN," *Opt. Precis. Eng.*, vol. 24, no. 4, pp. 873–881, 2016.

[40] J. Yang, Y.-Q. Zhao, and J. C.-W. Chan, "Learning and transferring deep joint spectral-spatial features for hyperspectral classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 8, pp. 4729–4742, Aug. 2017.

[41] J. Zhu, K. Tan, and D. Qu, "Tri_training for remote sensing classification based on multi-scale homogeneity," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2016, pp. 3055–3058.

[42] L. G. Wang, Y. S. Yang, and T. T. Lu, "Semi-supervised classification for hyperspectral image based on tri-training," *Appl. Mech. Mater.*, vols. 687–691, pp. 3644–3647, Nov. 2014.

[43] C. Ying, W. Xueting, L. Zhongjun, and W. Liguo, "Hyperspectral image classification based on improved M-training algorithm," (in Chinese), *J. Harbin Eng. Univ.*, vol. 39, no. 10, pp. 104–110, 2018.

[44] F. Aguilera, L. M. Valenzuela, and A. Botequilha-Leitão, "Landscape metrics in the analysis of urban land use patterns: A case study in a Spanish metropolitan area," *Landscape Urban Planning*, vol. 99, no. 3, pp. 226–238, 2011.

[45] L. Wan, K. Tang, M. Li, Y. Zhong, and A. K. Qin, "Collaborative active and semisupervised learning for hyperspectral remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2384–2396, May 2015.

[46] X. Wang, Y. Zhong, L. Zhang, and Y. Xu, "Spatial group sparsity regularized nonnegative matrix factorization for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 11, pp. 6287–6304, Nov. 2017.

[47] L. Wei, X. Wan, G. Zhao, B. Cui, W. Liu, and B. Qi, "Spectral-spatial classification of hyperspectral imagery based on random forests," in *Proc. Int. Conf. Internet Multimedia Comput. Service*, 2013, pp. 47–63.

[48] T. Li, J. Leng, L. Kong, S. Guo, G. Bai, and K. Wang, "DCNR: Deep cube CNN with random forest for hyperspectral image classification," *Multimedia Tools Appl.*, vol. 78, no. 3, pp. 3411–3433, Feb. 2018.

[49] Q. Lu, X. Huang, J. Li, and L. Zhang, "A novel MRF-based multifeature fusion for classification of remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 4, pp. 515–519, Apr. 2016.

[50] L. Zhang, L. Zhang, D. Tao, and X. Huang, "Tensor discriminative locality alignment for hyperspectral image spectral-spatial feature extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 1, pp. 242–256, Jan. 2013.

HENG WANG received the master's degree in information and communication engineering from Harbin Engineering University, Harbin, China, in 2019. His research interests include hyperspectral image processing, computer science, and artificial intelligence.

LONG TENG received the B.E. degree in electronic information engineering from the Shaanxi University of Technology, Shaanxi, China, in 2016. He is currently pursuing the Ph.D. degree in information and communication engineering with Harbin Engineering University. His research interests include radar signals sorting, deep learning, and image processing.

XIAOWEI JI received the B.E. degree in electronic information engineering from Northeast Forestry University (NEFU), Harbin, China, in 2017. She is currently pursuing the M.S. degree in information and communication engineering with Harbin Engineering University (HEU). Her research interests include signal processing, remote sensing, and machine learning. She also participates in reviewing IEEE Access.

LINGXIU WANG received the bachelor's degree in electronic information engineering from Northeast Forestry University (NEFU), in 2018. She is currently pursuing the master's degree in information and communication engineering with Harbin Engineering University (HEU). Her research interests include remote sensing image processing and machine learning.

YING CUI was born in Harbin, Heilongjiang, China, in 1979. She received the Ph.D. degree in signal and information processing and the Ph.D. degree in computer science and technology from Harbin Engineering University (HEU), Harbin, in 2006 and 2009, respectively. She joined the Faculty, in 2005 and is currently an Associate Professor with the School of Information and Communication Engineering, HEU. Her research interests include intelligent signal processing, image processing, and wireless sensor network optimization. She is a Reviewer for many journals, including the *Journal of Wuhan University*, the *Journal of Harbin Engineering University*, the *Journal of Applied Remote Sensing*, and *Acta Optica Sinica and Laser & Optoelectronics Progress*.

LIGUO WANG received the M.S. degree in natural science and the Ph.D. degree in engineering from the Harbin Institute of Technology (HIT), Harbin, China. He is currently a Full Professor with the School of Information and Communication Engineering, Harbin Engineering University. He is also a member of the Chinese Society for Optical Engineering, a Senior Member of the Chinese Institute of Electronics and the China Institute of Communications (CIC), and a Fellow of the International Society for Digital Earth (ISDE). He is also a member of the Senior Committee of Photoelectric Technology of Astronautics Society. He is a Reviewer for many journals, including the IEEE Transactions on Geoscience and Remote Sensing, the IEEE Geoscience and Remote Sensing Letters, and the Journal of Electronics and Information Technology.

• • •