Learning Hierarchical Spectral–Spatial Features for Hyperspectral Image Classification

Yicong Zhou, *Senior Member, IEEE*, and Yantao Wei

***Abstract*—This paper proposes a spectral–spatial feature learning (SSFL) method to obtain robust features of hyperspectral images (HSIs). It combines the spectral feature learning and spatial feature learning in a hierarchical fashion. Stacking a set of SSFL units, a deep hierarchical model called the spectral–spatial networks (SSN) is further proposed for HSI classification. SSN can exploit both discriminative spectral and spatial information simultaneously. Specifically, SSN learns useful high-level features by alternating between spectral and spatial feature learning operations. Then, kernel-based extreme learning machine (KELM), a shallow neural network, is embedded in SSN to classify image pixels. Extensive experiments are performed on two benchmark HSI datasets to verify the effectiveness of SSN. Compared with state-of-the-art methods, SSN with a deep hierarchical architecture obtains higher classification accuracy in terms of the overall accuracy, average accuracy, and kappa (κ) coefficient of agreement, especially when the number of the training samples is small.**

***Index Terms*—Hierarchical learning, hyperspectral image classification, kernel-based extreme learning machine, spectral–spatial feature.**

I. Introduction

HYPERSPECTRAL sensors can capture hundreds of spectral channels for each image pixel from ultraviolet to infrared nowadays. This characteristic makes it possible to accurately discriminate different materials of interest [1], [2]. The classification of hyperspectral image (HSI) has become one of the most important tasks for many applications including both commercial and military domains. However, the high dimensionality of HSIs causes several theoretical and practical challenges to classification [1], [3].

In order to deal with the problems encountered in HSI classification, many methods have been proposed in the past few years. Dimensionality reduction is an effective way to improve the HSI classification accuracy [4], [5]. Many dimensionality reduction methods have been adopted for HSI classification. For instance, locally linear embedding has been utilized to reduce the dimensionality of HSIs [6]. To make use of the label information, some supervised dimensionality reduction methods have been proposed to extract HSI features [5], [7]. Recently, semi-supervised methods have also been proposed for HSI dimensionality reduction [4], [8]. Apart from dimensionality reduction algorithms, designing effective spectral classifiers is another possible way to promote the classification accuracy. Support vector machine (SVM), a commonly used classifier [9], has been used successfully for HSI classification [10]. However, choosing suitable kernel functions, kernel-specific parameters, and regularization parameters is the major concern in the design of SVM [11]. Recently, kernel-based extreme learning machine (KELM) has been proposed [12]. Compared with SVM, KELM is faster and has good generalization ability [12], [13]. It has been applied to HSI classification [14], and the results confirm that it is comparable in accuracy with SVM and has lower computational complexity. However, most of these existing methods make use of only the spectral information of HSIs.

To cope with this problem, spectral–spatial-based classification techniques have received considerable attentions [15]–[23]. Many techniques have been adopted to make use of spatial information of HSIs, such as morphological profiles [24], Markov fields [1], or Gabor filters [25]. These spectral–spatial-based methods assume that pixels within a local region usually represent the same material [25] and have achieved promising results [20]–[22], [26]. Quesada-Barriuso *et al.* created an extended morphological profile (EMP) from the wavelet features to obtain spectral–spatial features [27]. In [28], a spectral–spatial classification method for HSIs was proposed using morphological component analysis-based image separation rationale in the sparse representation. Maximizer of the posterior marginal by loopy belief propagation (MPM-LBP) was proposed by Li *et al.* [29]. It exploits the marginal probability distribution from both the spectral and spatial information. Zhong *et al.* [30] developed a discriminant tensor spectral–spatial feature extraction method for HSI classification. Kang *et al.* [31] proposed a spectral–spatial classification framework based on edge-preserving filtering (EPF), where the filtering operation achieves a local optimization of the probabilities. Gabor filters and local binary pattern (LBP) were introduced for extracting local spatial features of HSIs in [32] and [33], respectively.

Manuscript received January 31, 2015; revised May 9, 2015; accepted July 1, 2015. Date of publication July 28, 2015; date of current version June 14, 2016. This work was supported in part by the Macau Science and Technology Development Fund under Grant FDCT/106/2013/A3, in part by the Research Committee at the University of Macau under Grant MYRG2014-00003-FST, Grant MRG017/ZYC/2014/FST, Grant MYRG113(Y1-L3)-FST12-ZYC, and Grant MRG001/ZYC/2013/FST, in part by the Fundamental Research Funds for the Central Universities under Grant CCNU14A05023, and in part by the Wuhan Science and Technology Plan Project under Grant 20140601010100030 and Grant 2014010101010025. This paper was recommended by Associate Editor D. Tao. *(Corresponding author: Yantao Wei.)*

Y. Zhou is with the Department of Computer and Information Science, University of Macau, Macau 999078, China (e-mail: yicongzhou@umac.mo).

Y. Wei is with the School of Educational Information Technology, Central China Normal University, Wuhan 430079, China (e-mail: yantaowei@mail.ccnu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2015.2453359

Soltani-Farani *et al.* [34] presented spatial aware dictionary learning (SADL) method for HSI classification. It is a structured dictionary-based model incorporating both spectral and contextual characteristics of spectral samples. Recently, Li *et al.* developed a new framework for the classification of HSIs that pursues the combination of multiple features [18]. It can deal with linear and nonlinear class boundaries in HSI data.

However, most of these methods do not extract spectral–spatial features in a hierarchical fashion. Recent studies show that the hierarchical deep model can extract more abstract and invariant features of data. It has the ability of yielding higher classification accuracy than those traditional and shallower classifiers [35]–[39]. Consequently, designing a deep learning model with spectral and spatial information is a promising direction to be explored. Chen *et al.* [35] applied a deep learning method to HSI classification. However, they did not design the structure of the deep learning methods according to the characteristics of HSIs.

Based on above discussions, this paper presents a hierarchical HSI classification model called spectral–spatial networks (SSN). Generally, HSI classification systems have the delicate task of describing a smooth land cover using spectral information with a high within-class variability. It is crucial to exploit the nonlinear characteristics of HSIs. The proposed method intends to learn the discriminative features using the hierarchical deep architecture. It can extract spectral–spatial features by iteratively abstracting neighboring regions and recomputing representations for new regions. In this way, the within-class variability will be reduced and the classification maps become smoother. Thus, the hierarchical deep architecture, which extracts more abstract and effective features of the HSI data, can overcome the problems faced by the shallow architecture. The main differences between the proposed method and other deep learning methods are that the convolutional filters used in the proposed method are learned directly from the images rather than learned by the stochastic gradient descent method used in the traditional deep learning methods, and that the structure of the proposed method is specially designed based on the characteristics of the HSI. The major contributions of this paper can be summarized as follows.

1) A new spectral–spatial feature learning (SSFL) method is proposed. It combines the spectral feature learning and spatial feature learning in a hierarchical fashion.
2) Based on SSFL, SSN as a feedforword network is designed for HSI classification. It can learn discriminative spectral–spatial features of HSIs explicitly by embedding the simple supervised learning methods in the deep hierarchical architecture.
3) It provides a new way to learn spectral–spatial features in a hierarchical fashion. In the similar way, more hierarchical methods can also be designed.
4) SSN combines the simple subspace learning method and KELM in the framework of deep hierarchical learning. It achieves higher accuracy compared with state-of-the-art methods, especially when the number of the training samples is small.

The rest of this paper is organized as follows. Section II briefly reviews deep learning and KELM. The proposed SSFL method is given in Section III. It combines the spectral information and spatial context to promote classification performance. Section IV presents a detailed description of the proposed HSI classification method called SSN. Section V shows experimental results on two widely used HSI datasets and the performance comparisons with various methods. Finally, Section VI presents the conclusions and possible future research.

II. Related Work

The motivation of this paper is to design a deep hierarchical model for HSI classification. To make use of the spectral and spatial information of HSIs, the proposed method combines the deep hierarchical architecture, subspace learning, and KELM. Consequently, deep learning and KELM are reviewed firstly.

A. Deep Learning

Deep learning, inspired by the mechanism of human vision, recently attracted more and more attentions due to its good performance in many fields such as speech recognition, computer vision, and natural language processing [40]–[42]. The intention of deep learning is to discover more abstract representations in higher levels [43]. It involves a class of models to hierarchically learn high-level features of input data with a deep hierarchical architecture. Deep learning has the general formulation as

$$f(\mathbf{x}) \approx g_1(g_2(\ldots(g_n(\mathbf{x}))\ldots)) \tag{1}$$

where $\mathbf{x}$ is the input, $g_i(i = 1, \ldots, n)$ is the operation on the ith layer, and $f(\mathbf{x})$ is the new representation of $\mathbf{x}$. The input of a higher layer is the output of its previous layer in the deep learning models. In this way, it can progressively lead to more abstract and complex features at higher layers. More abstract features are generally invariant to most local changes of the input. Commonly used deep learning models include deep belief networks [36], deep Boltzmann machines [44], stacked auto-encoders (SAE) [45], and convolutional neural networks. Recent study also shows that deep models can give better approximations to nonlinear functions than shallow models [46], [47].

HSI classification is an important pattern recognition task. The obtained features are expected to have the ability to discriminate pixels from different classes while being invariant to intraclass variability. For these reasons, SSN as a feedforward network is proposed for HSI classification in this paper. It can fuse the spectral and spatial information on different scales in a hierarchical fashion.

B. Kernel-Based Extreme Learning Machine

ELM aims at training single hidden layer feedforward neural networks (SLFN) [48]–[50]. It uses an idea to train SLFN, which is the hidden-node parameters are randomly generated based on certain probability distributions. ELM is closely

related to some previous work, such as [51]–[53]. A similar idea randomly generating the node parameters based on sparse representation has also been investigated in the matching problem, such as in [54]–[57]. ELM provides not only the smaller training error but also the better performance. Stacked ELM (S-ELM) divides a single large ELM network into multiple stacked small ELMs for solving large and complex data problems [58], and Kernel-based ELM (KELM) uses a kernel function to improve the stability of ELM [12].

Let N training samples be $\{\mathbf{x}_i, \mathbf{y}_i\}(i = 1, \ldots, N)$, where $\mathbf{x}_i \in R^d$ and $\mathbf{y}_i = (y_{i,1}, \ldots, y_{i,\mathfrak{C}}) \in R^{\mathfrak{C}}$ indicate the class. Here the training samples belong to $\mathfrak{C}$ classes and

$$\mathbf{y}_{i,j} = \begin{cases} 1, & \mathbf{x}_i \text{ belongs to the } j\text{th class} \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

The output function with L hidden neurons is

$$f_L(\mathbf{x}_i) = \sum_{j=1}^{L} \beta_j h_j(\mathbf{x}_i) = \mathbf{h}(\mathbf{x}_i)\beta = \mathbf{y}_i \tag{3}$$

where $\beta = \left(\beta_1 \beta_2 \ldots \beta_L\right)^T$ is the weight vector between the hidden layer and the output layer, and $\mathbf{h}(\mathbf{x}_i) = (h_1(\mathbf{x}_i), \ldots, h_L(\mathbf{x}_i))$ is the hidden-layer output corresponding to the input $\mathbf{x}_i$. The training pixels are mapped onto the L-dimensional feature space by $\mathbf{h}(\mathbf{x}_i)$. N equations coming from (3) can be written in a compact form and represented by $\mathbf{H}\beta = \mathbf{Y}$, where $\mathbf{H}$ is the output matrix of the hidden layer

$$\mathbf{H} = \begin{pmatrix} \mathbf{h}(\mathbf{x}_1) \\ \vdots \\ \mathbf{h}(\mathbf{x}_N) \end{pmatrix} = \begin{pmatrix} \mathbf{h_1}(\mathbf{x}_1) \ldots \mathbf{h_L}(\mathbf{x}_1) \\ \vdots \\ \mathbf{h_1}(\mathbf{x}_N) \ldots \mathbf{h_L}(\mathbf{x}_N) \end{pmatrix} \tag{4}$$

and

$$\mathbf{Y} = \left(\mathbf{y}_1 \mathbf{y}_2 \ldots \mathbf{y}_N\right)^T \tag{5}$$

is the expected output matrix of samples. The minimal norm least square solution of ELM is

$$\beta = \mathbf{H}^* \mathbf{Y} \tag{6}$$

where $\mathbf{H}^*$ is the Moore–Penrose generalized inverse of $\mathbf{H}$. Here the orthogonal projection method can be used to obtain $\mathbf{H}^*$, that is

$$\mathbf{H}^* = \mathbf{H}^T \left(\mathbf{H}\mathbf{H}^T\right)^{-1}. \tag{7}$$

In order to obtain a stable solution and better generalization performance, one can regularize $\mathbf{H}\mathbf{H}^T$ by adding a positive value ρ, then we have

$$f(\mathbf{x}_i) = \mathbf{h}\beta^T = \mathbf{h}(\mathbf{x}_i)\mathbf{H}^T \left(\frac{\mathbf{I}}{\rho} + \mathbf{H}\mathbf{H}^T\right)^{-1} \mathbf{Y}. \tag{8}$$

When the feature mapping $\mathbf{h}(\mathbf{x}_i)$ is unknown, we can define a kernel matrix for ELM as follows:

$$\mathbf{K}(\mathbf{X}, \mathbf{X}) = \mathbf{H}\mathbf{H}^T : k\left(\mathbf{x}_i, \mathbf{x}_j\right) = \mathbf{h}(\mathbf{x}_i) * \mathbf{h}\left(\mathbf{x}_j\right)^T. \tag{9}$$

A lot of kernel functions can be used in this method, and they do not need to satisfy the Mercer's theorem. In this paper, the commonly used radial basis function kernel is selected. The output of the KELM classifier is

$$\begin{aligned} f(\mathbf{x}_i) &= \mathbf{h}(\mathbf{x}_i)\mathbf{H}^T \left(\frac{\mathbf{I}}{\rho} + \mathbf{H}\mathbf{H}^T\right)^{-1} \mathbf{Y} \\ &= \begin{pmatrix} \mathbf{K}(\mathbf{x}_i, \mathbf{x}_1) \\ \vdots \\ \mathbf{K}(\mathbf{x}_i, \mathbf{x}_N) \end{pmatrix}^T \left(\frac{\mathbf{I}}{\rho} + \mathbf{K}(\mathbf{X}, \mathbf{X})\right)^{-1} \mathbf{Y}. \end{aligned} \tag{10}$$

The label of the input data is determined by the index of the output node with the highest output value [48].

KELM has been widely used for many applications. In this paper, this effective shallow learning machine is embedded into the hierarchical architecture to obtain a deep learning model.

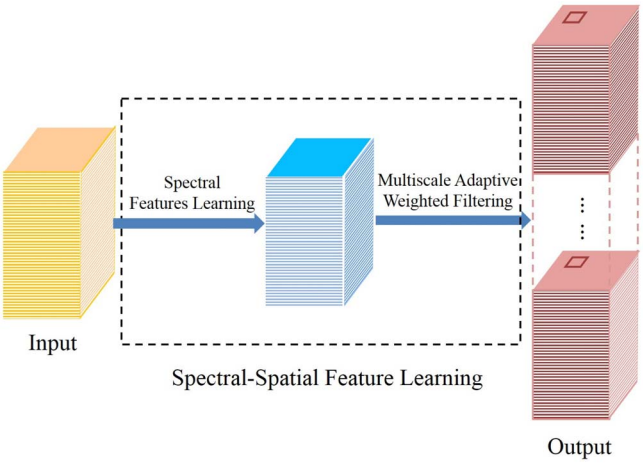

Fig. 1. Architecture of SSFL.

III. Spectral–Spatial Feature Learning Method

In order to design the deep hierarchical learning method, SSFL is proposed firstly in this paper. SSFL consists of the spectral feature learning and spatial feature learning (see Fig. 1). Next, we will describe them in detail.

A. Spectral Feature Learning

The training set is denoted by $\boldsymbol{I}_{\text{tr}} = \{\boldsymbol{I}_1, \boldsymbol{I}_2, \ldots, \boldsymbol{I}_N\}$, where $\boldsymbol{I}_j \in R^{d'}$ $(j = 1, 2, \ldots, N)$ is the jth training pixel. These labeled samples can be divided into $\mathfrak{C}$ classes. To extract discriminative spectral features from the training pixels, a supervised subspace learning method, linear discriminant analysis (LDA) [59] is used. LDA aims at maximizing the between-class scatter while minimizing the within-class scatter [59]. Assuming that K_{spe} is the number of filters in this stage and $\boldsymbol{W}_{\text{spe}} \in R^{d' \times K_{\text{spe}}}$ is the filter set. For the cth class, the mean of the new representation is given by

$$\boldsymbol{m}_c = \frac{1}{N_c} \sum_{n=1}^{N_c} \boldsymbol{W}_{\text{spe}}^T \tilde{\boldsymbol{I}}_n^c \tag{11}$$

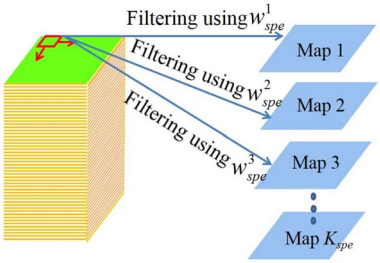

Fig. 2. Filtering operations on the spectral leaning stage.

where N_c is cardinality of the cth class set and $\tilde{\boldsymbol{I}}_n^c$ is the nth training pixel belonging to the cth class. Let $\boldsymbol{S}_w$ be the within-class scatter, that is

$$\boldsymbol{S}_w = \sum_{c=1}^{C} p_c \frac{1}{N_c} \sum_{i=1}^{N_c} \left(\boldsymbol{W}_{\text{spe}}^T \tilde{\boldsymbol{I}}_i^c - \boldsymbol{m}_c\right)\left(\boldsymbol{W}_{\text{spe}}^T \tilde{\boldsymbol{I}}_i^c - \boldsymbol{m}_c\right)^T \quad (12)$$

where $p_c = N_c/N$. The total mean of the new representation of training pixels is given by

$$\boldsymbol{m} = \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{W}_{\text{spe}}^T \tilde{\boldsymbol{I}}_n. \quad (13)$$

Therefore, the between-class scatter of the training pixels is given by

$$\boldsymbol{S}_b = \sum_{c=1}^{C} p_c (\boldsymbol{m}_c - \boldsymbol{m})(\boldsymbol{m}_c - \boldsymbol{m})^T. \quad (14)$$

LDA maximizes the ratio of between-class scatter to within-class scatter using a family of filters, that is

$$\max_{\boldsymbol{W}_{\text{spe}} \in R^{d' \times K_{\text{spe}}}} = \frac{\boldsymbol{W}_{\text{spe}}^T \boldsymbol{S}_b \boldsymbol{W}_{\text{spe}}}{\boldsymbol{W}_{\text{spe}}^T \boldsymbol{S}_w \boldsymbol{W}_{\text{spe}}}. \quad (15)$$

The filters are known as the K_{spe} largest eigenvectors (eigenvectors correspond to a number of largest eigenvalues) of

$$\boldsymbol{S}_b \boldsymbol{w}_{\text{spe}} = \lambda \boldsymbol{S}_w \boldsymbol{w}_{\text{spe}}. \quad (16)$$

Once the filters are obtained, the normalized HSI is filtered pixel by pixel (see Fig. 2). Note that the filters are learned from the data, and other subspace learning methods can also be used to learn filters [60], [61]. We can find that the output of this stage or layer has K_{spe} maps.

B. Spatial Feature Learning

For the output of the spectral feature learning, the spatial information will be exploited using adaptive weighted filters (AWFs). AWF is a spatial filter within a block region, where the central pixel (vector) is replaced with the generated feature according to the weights assigned to its neighbors. The adaptive weights can be defined by

$$w_{\text{spa}}^{i,j} = \frac{s_{i,j}}{\sum_1^{m \times m} s_{i,j}} \quad (17)$$

Fig. 3. AWF whose size is 3×3.

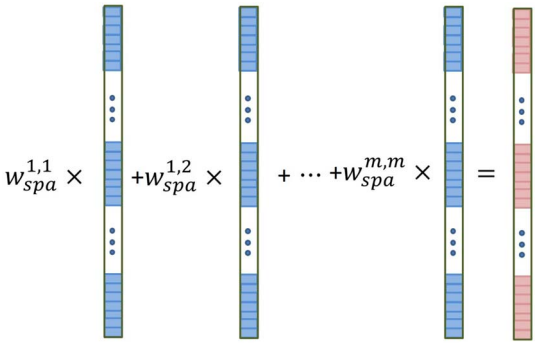

Fig. 4. Weighted filtering at the spatial feature learning stage.

where $m \times m$ is the size of the filter and

$$s_{i,j} = \exp\left(-\frac{\|p_0 - p_{i,j}\|^2}{\sigma}\right) \quad (18)$$

is the similarity measure. In (18), p_0 and $p_{i,j}$ are the central pixel and the pixel located in the ith row and jth column, respectively. σ can be adaptively determined by

$$\sigma = \frac{1}{\text{std}(\boldsymbol{d})} \quad (19)$$

where

$$d_{(i-1)\times m+j} = \| p_0 - p_{i,j} \|^2 . \quad (20)$$

In Fig. 3, an AWF whose size is 3×3 is given. Note that AWF has different weights on different positions.

Once the weights are obtained, we can generate a new representation of the central pixel by weightedly summing the neighboring pixels. From the definition of AWF, we can find that the defined filters are smoothing filters. The weights are defined by the similarity. In this case, the weights between the central pixel and its neighboring pixels within the same class are larger. On the other hand, the pixels within a local region usually represent the same material in the HSI. Consequently, the obtained pixels within the same class may become more similar. For this reason, we can say that AWF ensures that pixels in the same class have similar features. The process is illustrated in Fig. 4, where the output is our obtained spectral–spatial feature.

In practice, HSIs may contain homogenous regions with different sizes. Consequently, the multiscale AWFs are used to capture different spatial structures of HSIs. In this way, the output of the previous stage can be filtered by the AWFs on K_{spa} scales. For a given pixel, the spectral–spatial features on different scales can be concatenated into a new "pixel." Finally, the output of SSFL can be taken as a new "HSI,"

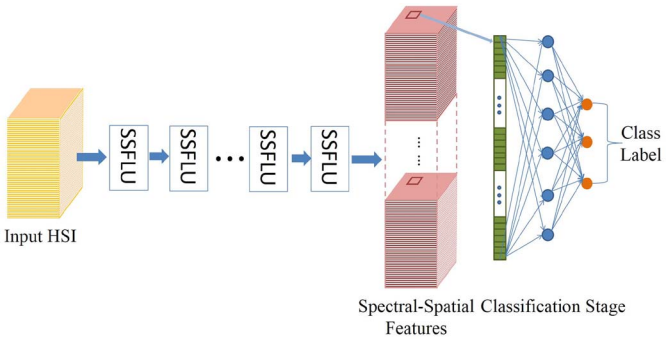

Fig. 5. Flowchart of the proposed SSN.

where the number of the bands is $K_{\text{spe}} \times K_{\text{spa}}$. When performing SSFL in an iterative way, we are able to form a deep hierarchical architecture.

IV. Spectral–Spatial Networks

This section proposes a deep hierarchical SSFL model called SSN. It contains a set of stacked SSFL units (SSSFLUs). The flowchart of SSN is given in Fig. 5. First, discriminative spectral–spatial features are learned by SSSFLUs which lead to a deep learning architecture. Then, the classification stage assigns a label to each pixel. SSN is composed of three stages: 1) image preprocessing; 2) SSSFLUs; and 3) classification.

A. Image Preprocessing

In this stage, an HSI is normalized. Suppose that the maximum and minimum of the input image are Max and Min, respectively. The radiance values are normalized to $[0, 1]$ in the following way:

$$\tilde{\boldsymbol{I}}_{ij}(n) = \frac{\boldsymbol{I}_{ij}(n) - \text{Min}}{\text{Max} - \text{Min}} \tag{21}$$

where $\tilde{\boldsymbol{I}}_{ij}(n)$ is the nth component of the pixel in the ith row and jth column in the HSI. In this way, a normalized HSI can be obtained.

B. SSSFLUs

In this stage, a normalized HSI is fed into the SSSFLU. Several SSSFLUs form a deep hierarchy with multiple layers. Each layer receives its input from the output of the previous layer. Note that the projection directions in the spectral feature learning are obtained in the training stage. AWFs are obtained according to the data to be processed.

C. Classification

In order to perform classification utilizing the learned features, we feed the learned spectral–spatial features to a KELM classifier. The output-layer size is set to be the same as the total number of classes, and the input has the same size as the dimension of output features of SSSFLUs (see Section II). Because KELM is implemented as a single-layer neural network, it can be integrated with the former layers of networks to obtain a deep model. This stage consists of two layers, where the input layer is the output layer of the previous SSSFLUs.

Algorithm 1 SSN: Training Procedure

Input: $\mathbb{I} \in R^{m' \times n' \times d'}$, $\boldsymbol{I}_{tr} = [\boldsymbol{I}_1 \ldots \boldsymbol{I}_N]$.
Output: the SSN model.

```
Image Normalization.
for l = 1 : L do
    Perform LDA to obtain K^l_spe projection directions.
    for i = 1 : m' do
        for j = 1 : n' do
            Project the pixel in the ith row and jth column to K^l_spe projection directions.
        end for
    end for
    for q = 1 : K^l_spa do
        Perform adaptive weighted filtering on the qth scale.
    end for
    Concatenate the filtered images on different scales.
end for
Feed spectral-spatial features to KELM to train a classifier.
```

Algorithm 2 SSN: Test Procedure

Input: $\mathbb{I} \in R^{m' \times n' \times d'}$.
Output: Labels of the test pixels.

```
Image Normalization.
for l = 1 : L do
    for i = 1 : m' do
        for j = 1 : n' do
            Project the pixel in the ith row and jth column to K^l_spe projection directions.
        end for
    end for
    for q = 1 : K^l_spa do
        Perform adaptive weighted filtering on the qth scale.
    end for
    Concatenate the filtered images on different scales.
end for
Feed spectral-spatial features to the trained KELM classifier.
Output the labels of test pixels.
```

In summary, the training procedure of the proposed SSN is shown in Algorithm 1, where $\mathbb{I} \in R^{m' \times n' \times d'}$ is the HSI to be processed, L is the number of SSSFLUs, K^l_{spa} and K^l_{spe} are the numbers of the spatial filters and spectral filters in the lth units, respectively. In the training stage, the projection directions on different layers and the KELM classifier are learned. Similarly, the testing procedure can be found in Algorithm 2.

As we will see through extensive experiments, the SSN model is simple but effective to make full use of the spectral–spatial information. In SSFLUs, LDA can ensure the pixels within the same class have the similar features no matter how far away in the spatial space they are, and AWF is to ensure the neighbor pixels within the same class have similar features.

TABLE I
NUMBERS OF SAMPLES IN EACH GROUND-TRUTH CLASS IN THE INDIAN PINES DATASET

Class No	Class Name	Samples Subtotal
1	Alfalfa	54
2	Corn-notill	1434
3	Corn-mintill	834
4	Corn	234
5	Grass-pasture	497
6	Grass-trees	747
7	Grass-pasture-mowed	26
8	Hay-windrowed	489
9	Oats	20
10	Soybean-notill	968
11	Soybean-mintill	2468
12	Soybean-clean	614
13	Wheat	212
14	Woods	1294
15	Buildings-Grass-Trees-Drives	380
16	Stone-Steel-Towers	95
Total		**10366**

TABLE II
NUMBERS OF SAMPLES IN EACH GROUND-TRUTH CLASS IN THE UNIVERSITY OF PAVIA DATASET

Class No	Class Name	Samples Total
1	Asphalt	6852
2	Meadows	18686
3	Gravel	2207
4	Trees	3436
5	Painted metal sheets	1378
6	Bare Soil	5104
7	Bitumen	1356
8	Self-Blocking Bricks	3878
9	Shadows	1026
Total		**43923**

TABLE III
OAS OF DIFFERENT METHODS ON THE AVIRIS INDIAN PINES DATASET

Method	1%	2%	3%	4%	5%
SVM	57.86±2.86	65.14±1.27	69.85±0.88	72.79±1.07	74.58±0.91
KELM	58.52±3.30	66.08±1.03	71.35±1.15	74.18±0.87	75.87±0.89
PCA-KELM	58.22±3.30	65.94±0.99	71.27±1.20	74.22±0.89	75.85±0.82
MH-KELM	78.04±2.86	88.77±0.73	93.19±1.29	94.73±1.00	96.56±0.70
EPF	71.75±2.58	76.34±7.38	83.51±2.67	86.22±1.61	88.46±1.16
MPM-LBP	77.16±3.46	84.29±1.84	88.19±1.34	90.82±1.47	91.60±0.98
SADL	78.95±1.09	88.19±1.49	90.74±0.97	92.95±0.66	94.47±1.12
SSN	**84.70±2.28**	**91.33±0.97**	**93.96±1.33**	**95.59±0.61**	**97.02±0.36**

V. EXPERIMENTAL RESULTS AND DISCUSSION

A. Datasets and Experimental Setups

In our experiments, two benchmark HSIs are used to verify the effectiveness of SSN. The first image is called Indian Pines. It was gathered by the AVIRIS sensor over the Indian Pines test site in North-western Indiana. This image scene has the size of 145×145 pixels. The ground truth available is designated into 16 classes. The name and quantity of each class are reported in Table I. The number of bands has been reduced to 200 by removing bands covering the region of water absorption. This scene constitutes a challenging classification problem due to the significant presence of mixed pixels in all available classes and of the unbalanced number of available labeled pixels per class [29].

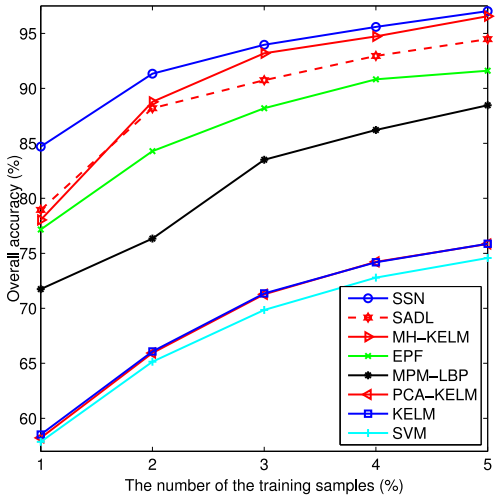

Fig. 6. OAs of different methods with different numbers of training samples on the Aviris Indian Pines dataset.

TABLE IV
SUMMARY OF PARAMETERS ON THE AVIRIS INDIAN PINES DATASET

l	K^l_{spe}	K^l_{spa}	$m \times m$
1	15	5	3×3, 5×5, 7×7, 9×9, 11×11
2	15	5	3×3, 5×5, 7×7, 9×9, 11×11
3	15	5	3×3, 5×5, 7×7, 9×9, 11×11

The second image used in experiments is the University of Pavia image, which was acquired by the ROSIS sensor during a flight campaign over Pavia, northern Italy. This scene has the size of 610 × 340 pixels (covering the wavelength range from 0.4 to 0.9 μm). Nine ground-truth classes are used in our experiments. In our experiments, 12 most noisy bands are removed and finally 103 out of the 115 bands are used. The class descriptions and sample distributions for this image are given in Table II. The total number of labeled samples in this image is 43 923.

For both images, $r\%$ ($r = 1, 2, 3, 4, 5$) of labeled samples from each class are randomly chosen for training, and the remaining samples are used for testing. The window sizes of the AWFs are $m = 3, 5, \ldots, 11$. In order to deal with the pixels on the border of the HSI, we pad the image with mirror reflections of itself. In this paper, different methods are compared based on the overall accuracy (OA), average accuracy (AA) and κ coefficient, where OA is the percentage of correctly classified pixels in the testing set, AA is the mean of class-specific accuracy values, and κ is the percentage of agreement corrected by the number of agreements that would be expected purely by chance [62]. All experiments are carried out using MATLAB on an Intel i7-4790 3.60 GHz machine with 12 GB RAM.

B. Experimental Results on the Indian Pines Dataset

Table III and Fig. 6 give OAs of different methods, where MH-KELM is the multihypothesis-based KELM [26], [63].

TABLE V
CLASS-SPECIFIC CLASSIFICATION ACCURACIES (IN PERCENTAGE), OA (IN PERCENTAGE), AA (IN PERCENTAGE), AND κ COEFFICIENTS FOR THE AVIRIS INDIAN PINES DATASET

Method	SVM	KELM	PCA-KELM	MH-KELM	EPF	MPM-LBP	SADL	SSN
Alfalfa	27.74	10.57	11.13	77.17	28.04	36.04	69.81	**81.32**
Corn-notill	44.54	44.88	44.28	74.62	64.71	71.42	68.73	**80.51**
Corn-mintill	36.97	39.94	41.71	70.56	75.69	54.16	65.58	**79.75**
Corn	21.21	15.50	15.45	55.11	49.84	42.03	62.81	**68.57**
Grass-pasture	59.67	57.62	58.23	68.11	**96.28**	78.23	82.34	79.25
Grass-trees	84.37	82.95	81.84	83.94	73.47	**96.81**	93.84	95.18
Grass-pasture-mowed	76.80	59.20	57.20	88.00	47.86	52.40	100.0	**95.20**
Hay-windrowed	85.93	79.36	77.02	78.93	89.39	**95.56**	97.48	93.08
Oats	32.63	39.47	37.37	87.37	0	58.42	81.05	**89.47**
Soybean-mintill	49.32	45.22	46.71	77.64	77.40	64.21	74.11	**77.85**
Soybean-notill	62.03	67.26	65.99	85.62	66.69	**86.99**	81.34	85.62
Soybean-clean	31.91	32.19	32.87	63.43	52.19	63.38	57.71	**69.95**
Wheat	88.52	90.81	90.33	94.16	97.26	**99.23**	98.66	96.36
Woods	84.45	85.98	85.69	89.75	86.60	**97.42**	93.71	96.28
Buildings-Grass -Trees-Drives	21.49	25.27	25.29	60.82	72.26	40.08	68.70	**83.16**
Stone-Steel-Towers	48.62	34.57	32.45	23.94	20.00	19.15	79.57	**96.06**
OA	57.86±2.86	58.52±3.30	58.22±3.30	78.04±2.86	71.75±2.58	77.16±3.46	78.95±1.09	**84.70±2.28**
AA	53.51±1.58	50.67±2.46	50.22±2.57	73.70±4.04	62.35±7.46	65.97±5.81	79.72±2.78	**85.48±2.10**
κ	0.5200±0.0309	0.5235±0.0368	0.5207±0.0368	0.7482±0.0331	0.6709±0.0311	0.7364±0.0405	0.7598±0.0127	**0.8257±0.0255**

The experimental results reported here are averaged results over ten random runs, where the parameters are given in Table IV. We can find that the proposed SSN performs the best, especially when the number of the training samples is small. The experimental results of SVM, KELM, and PCA-KELM given in Table III and Fig. 6 are obtained by using spectral information only. We can find that these spectral–spatial-based methods perform much better than spectral-based methods. Consequently, our experiments also show the advantage of exploiting spatial information in HSI classification. Table III also shows that the accuracies of SSN have small standard deviations. This indicates that SSN has a stably excellent classification performance.

The classification accuracy for each class, OA, AA, and the κ coefficient are reported in Table V, where 1% of labeled samples from each class are randomly chosen for training. SSN shows a significant gain in accuracy, where the improvement in AA is about 6%. In addition, we perform the paired t-test between SSN and other methods. Fig. 7 shows the detailed statistics of the κ coefficients of different methods, where the central mark is the median value of κ. The edges of boxes are the 25th and 75th percentiles. Lines extending vertically from the boxes indicate variability outside the upper and lower quartiles. Abnormal outliers are shown as red "+"s [35]. Paired t-test results show that improvements on κ are statistically significant (at the level of 95%). This is due to the discriminative spectral information learning as well as the spatial dependence modeling in the proposed SSN. The full classification maps given in Fig. 8 also demonstrate the effectiveness of SSN.

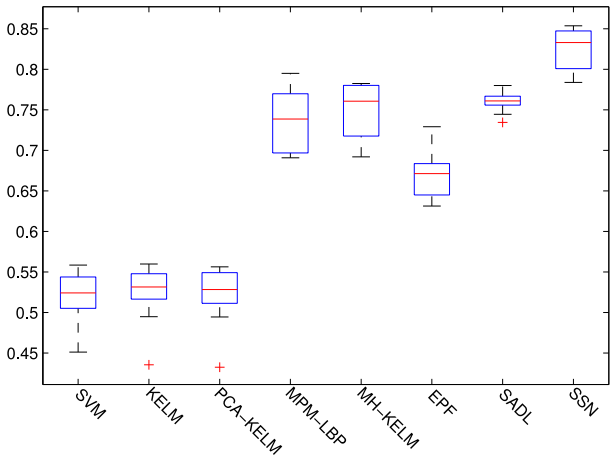

Fig. 7. Box plot of the κ coefficients of different methods on the Aviris Indian Pines dataset.

C. Experimental Results on the University of Pavia Dataset

The experimental results are given in Table VI and Fig. 9, where the parameters are given in Table VII. As can be observed, SSN performs the best, especially when the training data are limited (as low as 1%). We can find that the standard deviations of the SSN's OAs are smaller than those of other methods. This also indicates that SSN is more robust. SSN performs better than other deep learning models, and Chen's method achieves accuracy of 98.52% with 60% of the tagged samples used as the training set [35]. It also shows that SSN has a low sample complexity. This may be due to the fact that SSN is designed to directly incorporate operations to learn discriminative features. The same setup with that in [35] is also used, and the experimental results are given in Table VIII, where SAE-LR-S and SAE-LR-J are SAE with a logistic regression on spatial information and joint spectral–spatial information, respectively. The results of SAE-LR-S and SAE-LR-J are directly cited from the Chen's paper. We can find that SSN yields higher accuracy.

Table IX shows the OA, AA, κ, and individual class accuracies obtained in our comparisons, where 1% of labeled

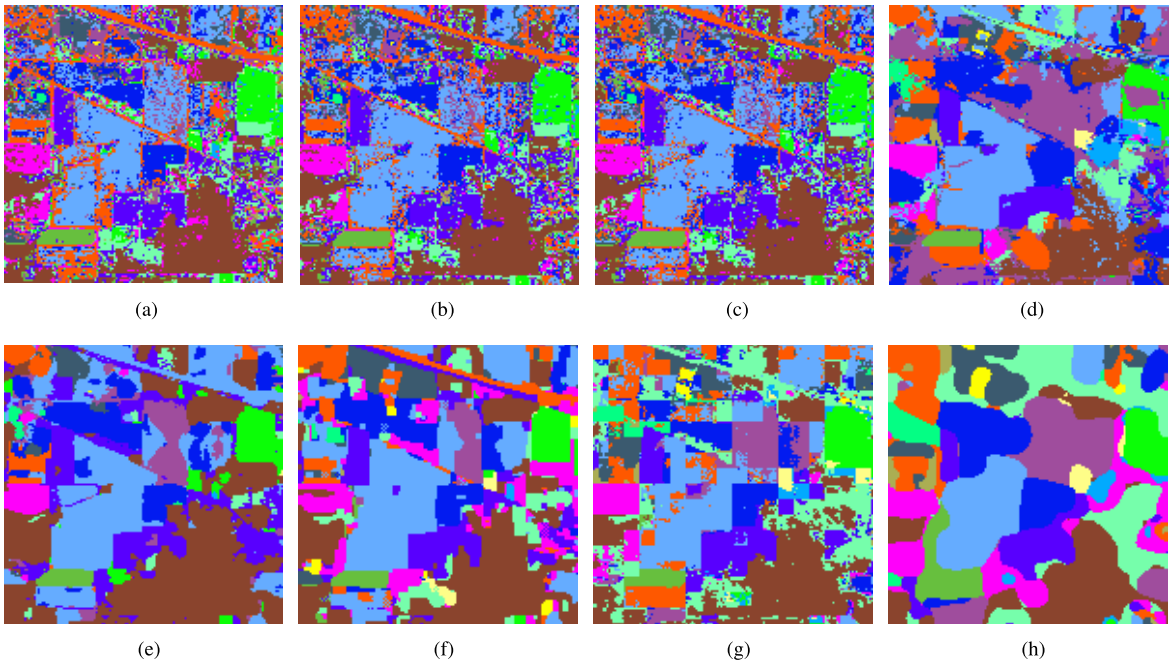

Fig. 8. Classification maps of the Indian Pines dataset using different methods. (a) SVM. (b) KELM. (c) PCA-KELM. (d) MH-KELM. (e) EPF. (f) MPM-LBP. (g) SADL. (h) SSN.

TABLE VI
OAs OF DIFFERENT METHODS ON THE UNIVERSITY OF PAVIA DATASET

Method	1%	2%	3%	4%	5%
SVM	89.19±0.68	91.30±0.36	92.39±0.32	92.89±0.30	93.26±0.18
KELM	89.08±0.72	90.76±0.24	91.92±0.33	92.43±0.18	92.74±0.16
PCA-KELM	89.16±0.76	90.94±0.30	92.03±0.31	92.48±0.15	92.82±0.16
MH-KELM	95.57±0.57	97.69±0.27	98.58±0.23	99.04±0.13	99.16±0.08
EPF	95.06±1.16	96.72±0.47	97.39±0.34	97.57±0.19	97.74±0.27
MPM-LBP	95.42±0.62	96.95±0.38	97.48±0.39	97.74±0.30	97.96±0.30
SADL	93.28±0.50	96.03±0.31	97.32±0.35	98.39±0.13	98.76±0.13
SSN	**97.53±0.20**	**98.33±0.26**	**98.95±0.11**	**99.22±0.12**	**99.36±0.11**

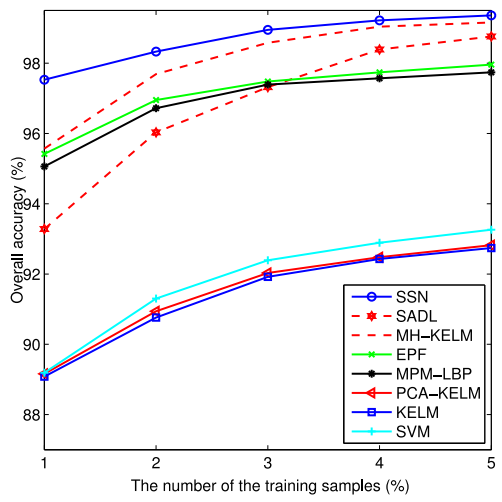

Fig. 9. OAs of different methods with various numbers of training samples on the University of Pavia dataset.

TABLE VII
SUMMARY OF PARAMETERS ON THE UNIVERSITY OF PAVIA DATASET

l	K^l_{spe}	K^l_{spa}	$m \times m$
1	30	5	3×3, 5×5, 7×7, 9×9, 11×11
2	30	5	3×3, 5×5, 7×7, 9×9, 11×11
3	30	5	3×3, 5×5, 7×7, 9×9, 11×11

TABLE VIII
COMPARISON WITH DEEP LEARNING-BASED METHODS ON THE UNIVERSITY OF PAVIA DATASET

Method	SAE-LR-S	SAE-LR-J	SSN
OA	98.12 [35]	98.52 [35]	99.57

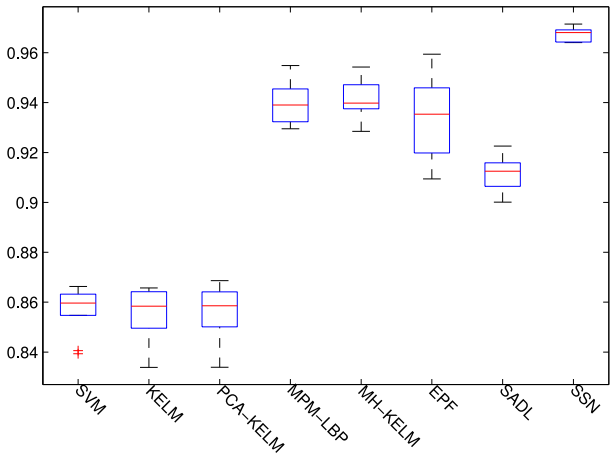

Fig. 10. Box plot of κ of different methods on University of Pavia dataset.

samples from each class are randomly chosen for training. The experimental results presented here are the averaged results over ten random runs. As can be seen, SVM has poor results because it uses only spectral information. Table IX also shows that the proposed SSN significantly outperforms other state-of-arts methods. Thus, our SSN can effectively exploit the spatial information. For more details, Fig. 10 shows statistical

TABLE IX
CLASS-SPECIFIC CLASSIFICATION ACCURACIES (IN PERCENTAGE), OA (IN PERCENTAGE), AA (IN PERCENTAGE), AND κ COEFFICIENTS FOR THE UNIVERSITY OF PAVIA DATASET

Method	SVM	KELM	PCA-KELM	MH-KELM	EPF	MPM-LBP	SADL	SSN
Asphalt	86.99	85.67	85.70	96.74	93.05	97.38	92.66	**98.79**
Meadows	97.33	97.50	97.57	99.79	95.37	99.45	98.92	**99.85**
Gravel	65.38	66.83	66.75	90.84	**96.87**	79.02	74.65	87.10
Trees	86.22	85.34	85.02	89.99	**99.46**	90.62	93.50	93.99
Painted metal sheets	98.39	97.69	97.70	51.46	98.09	97.52	99.38	**99.61**
Bare Soil	76.48	78.14	78.53	97.06	**98.52**	93.37	94.57	98.34
Bitumen	78.58	76.19	76.62	98.76	**99.54**	82.95	77.26	92.62
Self-Blocking Bricks	85.33	84.25	84.45	93.24	87.63	91.52	77.73	**93.97**
Shadows	96.09	96.96	96.90	96.03	96.88	98.94	**99.14**	94.20
OA	89.19±0.43	89.08±0.72	89.16±0.76	95.57±0.57	95.06±1.16	95.42±0.62	93.28±0.50	**97.53±0.20**
AA	85.64 ±0.65	85.40±1.75	85.47±1.76	90.43±1.14	**96.16± 0.75**	92.31±1.06	89.76±0.92	95.38±0.42
κ	0.8566±0.0094	0.8553±0.0101	0.8563±0.0106	0.9416±0.0075	0.9345±0.0157	0.9395±0.0082	0.9116±0.0067	**0.9675±0.0026**

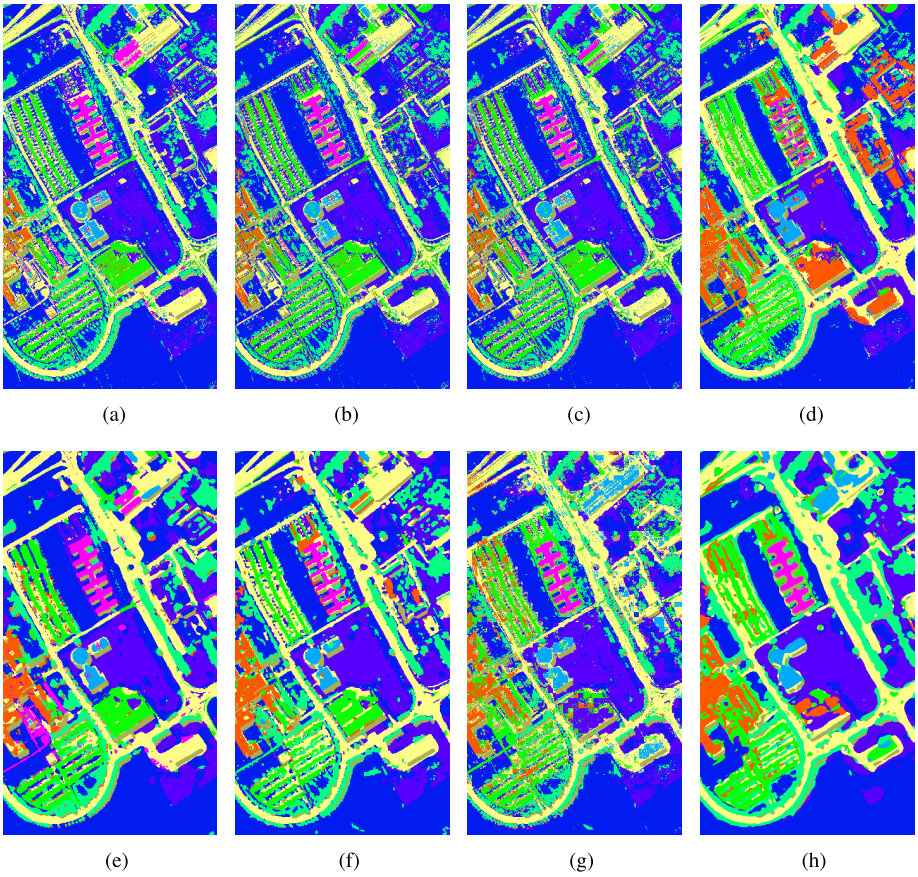

Fig. 11. Classification maps of the University of Pavia dataset using different methods. (a) SVM. (b) KELM. (c) PCA-KELM. (d) MH-KELM. (e) EPF. (f) MPM-LBP. (g) SADL. (h) SSN.

evaluations of κ coefficients. SSN has small standard deviation of κ. It also demonstrates the advantages of SSN. The full classification maps of different methods are given in Fig. 11. We can find that the maps of the spectral–spatial-based methods are smoother.

D. Discussion

In order to further analyze the proposed SSN, we test its performance on more experiments. The experimental results on the Indian Pines dataset are reported and the same conclusions can be made on the other dataset.

First, in order to demonstrate the effectiveness of the spectral–spatial strategy adopted by SSN, we compare SSN with KELM and LDA+KELM, where KELM and LDA+KELM make use of only the raw pixels. Table X shows the experimental results, where 1% of all labeled data for training. It is observed that our SSN can improve the classification accuracy significantly. The experimental results show that the proposed hierarchal SSFL module is important to SSN.

Second, we show effects of the depth on the classification accuracy. The depth plays a key role in SSN. We train a

TABLE X
CLASSIFICATION ACCURACIES ON THE INDIAN PINES DATASET

Method	KELM	LDA+KELM	SSN
1%	58.52	58.71	**84.70**

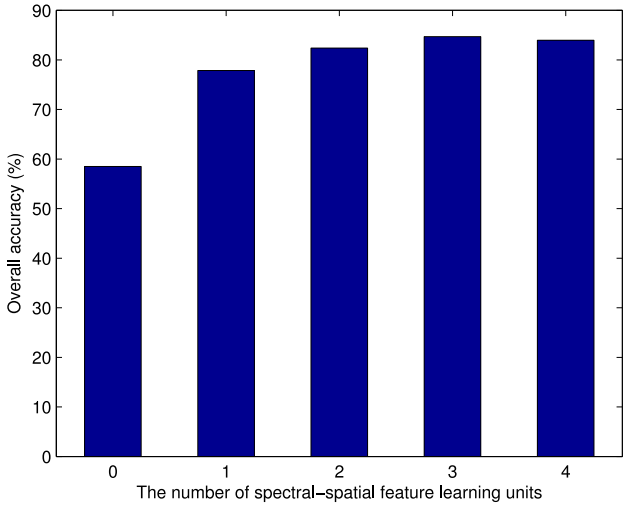

Fig. 12. Effect of different depths on OAs on the Indian Pines dataset.

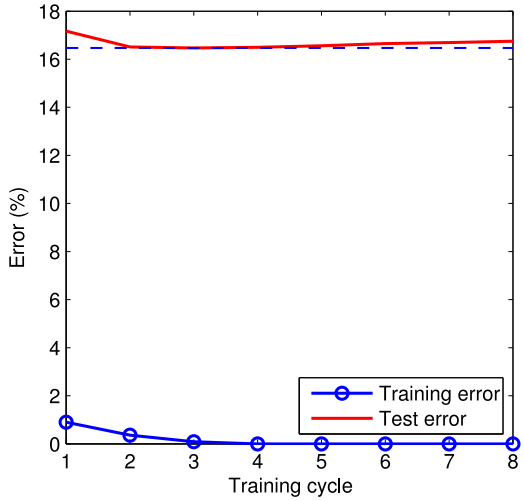

Fig. 13. Verify the overfitting on the Indian Pines dataset.

series of SSN with different depths, but with fixed projection direction numbers (15) and scale numbers (5) to see how the depth of the features effects classification accuracies. As shown in Fig. 12, more layers usually lead to a higher classification accuracy, where the depth is indicated by the number of SSFLUs (each SSFLU contains two layers). This helps us to determine how many layers are needed to obtain higher classification accuracy. Furthermore, we also find that too many layers and a small number of labeled training samples can result in overfitting to the training data. Fig. 13 shows that the proposed method with more layers has a slightly overfitting, where the number of SSSFLUs is set to 4.

Third, we compare the proposed features with common features (LBP, scale-invariant feature transform (SIFT), and Gabor) used in image classification. The experimental results

TABLE XI
COMPARISONS WITH COMMON FEATURES USED IN IMAGE CLASSIFICATION

Method	Accuracy
SIFT	84.32
Gabor-SVM [33]	86.82
LBP-SVM [33]	90.63
Gabor-Spec-ELM [33]	90.61
LBP-Spec-ELM [33]	92.76
SSN	**93.62**

TABLE XII
COMPARISONS OF DIFFERENT CLASSIFIERS

Method	Soft-Max	SVM	KELM
Accuracy	86.40	90.75	**93.62**
Training time	8.5060S	0.0292S	**0.0233S**

are given in Table XI, where the experiment setup is the same with that in [33]. These results show that the proposed method performs better than these benchmark features. Consequently, we can conclude that the proposed method can learn spectral–spatial features efficiently.

Finally, in order to explain why KELM is used in the final stage, the comparisons with multiple-class SVM and soft-max classifier are given in Table XII (the experiment setup is also the same with that in [33]). We can find that KELM leads to a higher classification accuracy and is fast. Consequently, KELM is used in the final stage for classification.

VI. CONCLUSION

In this paper, a hierarchical spectral–spatial-based HSI classification method called SSN has been proposed. It consists of SSSFLUs and KELM. In each SSFLU, the spectral feature learning stage learns discriminative spectral features while the spatial feature learning stage catches the spatial information using the multiscale AWFs. KELM is embedded to the hierarchical architecture to obtain classification results. Extensive experiments have been performed on the Indian Pines and University of Pavia datasets to verify the effectiveness of SSN. Experimental results have demonstrated good robustness and accuracy of the proposed SSN. It has been shown that hierarchical spectral–spatial features are useful for HSI classification. SSN is also attractive for advanced classification of the HSI datasets with limited training samples.

Although the experiments confirm that SSN is suitable for HSI classification, several questions remain to be investigated in our future work.

1) In SSN, the spectral-spatial features are learned through the simple feature learning and image processing methods. It is interesting to mathematically analyze and justify its effectiveness and investigate new methods based on other advanced feature learning methods.
2) We have demonstrated that a deeper hierarchical architecture always leads to a higher classification accuracy, however, a too deep architecture will act in an opposite way. It is interesting to determine the optimal depth for a given HSI.

ACKNOWLEDGMENT

The authors would like to thank the University of Pavia and Prof. P. Gamba for kindly providing the ROSIS images of the University of Pavia, Prof. D. Landgrebe for making the AVIRIS Indian Pines hyperspectral dataset available to the community, Prof. G. B. Huang for sharing KELM source code, Prof. J. Li for sharing MPM-LBP source code, Dr. X. Kang for sharing EPF source code, Dr. C. Chen for sharing MH source code, and Dr. A. Soltani-Farani for sharing SADL source code. The authors would also like to thank the editor and anonymous reviewers for their insightful comments and suggestions that helped improve the quality of this paper.

REFERENCES

[1] P. Ghamisi, J. A. Benediktsson, and M. O. Ulfarsson, "Spectral-spatial classification of hyperspectral images based on hidden Markov random fields," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 5, pp. 2565–2574, May 2014.

[2] J. Li, P. R. Marpu, A. Plaza, J. M. Bioucas-Dias, and J. A. Benediktsson, "Generalized composite kernel framework for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 9, pp. 4816–4829, Sep. 2013.

[3] H. Li, G. Xiao, T. Xia, Y. Y. Tang, and L. Li, "Hyperspectral image classification using functional data analysis," *IEEE Trans. Cybern.*, vol. 44, no. 9, pp. 1544–1555, Sep. 2014.

[4] Q. Shi, L. Zhang, and B. Du, "Semisupervised discriminative locally enhanced alignment for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 9, pp. 4800–4815, Sep. 2013.

[5] Y. L. Chang, J. N. Liu, C. C. Han, and Y. N. Chen, "Hyperspectral image classification using nearest feature line embedding approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 1, pp. 278–287, Jan. 2013.

[6] T. Han and D. G. Goodenough, "Nonlinear feature extraction of hyperspectral data based on locally linear embedding (LLE)," in *Proc. Geosci. Remote Sens. Symp.*, vol. 2. Seoul, Korea, Jul. 2005, pp. 1237–1240.

[7] W. Li, S. Prasad, J. E. Fowler, and L. M. Bruce, "Locality-preserving dimensionality reduction and classification for hyperspectral image analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 4, pp. 1185–1198, Apr. 2012.

[8] C. Gong, D. Tao, K. Fu, and J. Yang, "Fick's law assisted propagation for semisupervised learning," *IEEE Trans. Neural Netw. Learn. Syst.*, DOI: 10.1109/TNNLS.2014.2376963.

[9] D. Tao, X. Tang, X. Li, and X. Wu, "Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 7, pp. 1088–1099, Jul. 2006.

[10] Y. Bazi and F. Melgani, "Toward an optimal SVM classification system for hyperspectral remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 11, pp. 3374–3385, Nov. 2006.

[11] G. Mountrakis, J. Im, and C. Ogole, "Support vector machines in remote sensing: A review," *ISPRS J. Photogramm. Remote Sens.*, vol. 66, no. 3, pp. 247–259, May 2011.

[12] G. B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 513–529, Apr. 2012.

[13] R. Zhang, Y. Lan, G.-B. Huang, Z.-B. Xu, and Y. C. Soh, "Dynamic extreme learning machine and its approximation capability," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 2054–2065, Dec. 2013.

[14] M. Pal, A. E. Maxwell, and T. A. Warner, "Kernel-based extreme learning machine for remote-sensing image classification," *Remote Sens. Lett.*, vol. 4, no. 9, pp. 853–862, Sep. 2013.

[15] Y. Zhou, J. Peng, and C. L. P. Chen, "Dimension reduction using spatial and spectral regularized local discriminant embedding for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 2, pp. 1082–1095, Feb. 2015.

[16] L. Sun, Z. Wu, J. Liu, L. Xiao, and Z. Wei, "Supervised spectral-spatial hyperspectral image classification with weighted Markov random fields," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 3, pp. 1490–1503, Mar. 2015.

[17] J. Xia, J. Chanussot, P. Du, and X. He, "Spectral–spatial classification for hyperspectral data using rotation forests with local feature extraction and Markov random fields," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2532–2546, May 2015.

[18] J. Li, X. Huang, P. Gamba, J. M. Bioucas-Dias, L. Zhang, J. A. Benediktsson, and A. Plaza, "Multiple feature learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 3, pp. 1592–1606, Mar. 2015.

[19] P. Ghamisi, M. D. Mura, and J. A. Benediktsson, "A survey on spectral–spatial classification techniques based on attribute profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2335–2353, May 2015.

[20] S. Bernabe, P. R. Marpu, A. Plaza, M. D. Mura, and J. A. Benediktsson, "Spectral–spatial classification of multispectral images using kernel feature space representation," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 1, pp. 288–292, Jan. 2014.

[21] R. Ji *et al.*, "Spectral–spatial constraint hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 3, pp. 1811–1824, Mar. 2014.

[22] L. Fang, S. Li, X. Kang, and J. A. Benediktsson, "Spectral–spatial hyperspectral image classification via multiscale adaptive sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 12, pp. 7738–7749, Dec. 2014.

[23] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, "Advances in spectral-spatial classification of hyperspectral images," *Proc. IEEE*, vol. 101, no. 3, pp. 652–675, Mar. 2013.

[24] M. Fauvel, J. Benediktsson, J. Chanussot, and J. Sveinsson, "Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 11, pp. 3804–3814, Nov. 2008.

[25] O. Rajadell, P. Garcia-Sevilla, and F. Pla, "Spectral–spatial pixel characterization using Gabor filters for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 4, pp. 860–864, Jul. 2013.

[26] C. Chen, W. Li, H. Su, and K. Liu, "Spectral–spatial classification of hyperspectral image based on kernel extreme learning machine," *Remote Sens.*, vol. 6, no. 6, pp. 5795–5814, May 2014.

[27] P. Quesada-Barriuso, F. Arguello, and D. B. Heras, "Spectral–spatial classification of hyperspectral images using wavelets and extended morphological profiles," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 4, pp. 1177–1185, Apr. 2014.

[28] Z. Xue, J. Li, L. Cheng, and P. Du, "Spectral–spatial classification of hyperspectral data via morphological component analysis-based image separation," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 1, pp. 70–84, Jan. 2015.

[29] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Spectral–spatial classification of hyperspectral data using loopy belief propagation and active learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 2, pp. 844–856, Feb. 2013.

[30] Z. Zhong *et al.*, "Discriminant tensor spectral-spatial feature extraction for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens. Lett.*, vol. 12, no. 5, pp. 1028–1032, May 2015.

[31] X. Kang, S. Li, and J. A. Benediktsson, "Spectral–spatial hyperspectral image classification with edge-preserving filtering," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 5, pp. 2666–2677, May 2014.

[32] W. Li and Q. Du, "Gabor-filtering-based nearest regularized subspace for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 4, pp. 1012–1022, Apr. 2014.

[33] W. Li, C. Chen, H. Su, and Q. Du, "Local binary patterns and extreme learning machine for hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 7, pp. 3681–3693, Jul. 2015.

[34] 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.

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

[36] G. E. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.

[37] Y. Tang, T. Xia, Y. Wei, H. Li, and L. Li, "Hierarchical kernel-based rotation and scale invariant similarity," *Pattern Recognit.*, vol. 47, no. 4, pp. 1674–1688, Apr. 2014.

[38] H. Li, Y. Wei, L. Li, and C. P. Chen, "Hierarchical feature extraction with local neural response for image recognition," *IEEE Trans. Cybern.*, vol. 43, no. 2, pp. 412–424, Apr. 2013.

[39] R. Salakhutdinov, J. B. Tenenbaum, and A. Torralba, "Learning with hierarchical-deep models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1958–1971, Aug. 2013.

[40] G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.
[41] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. 25*, Red Hook, NY, USA, 2012, pp. 1097–1105.
[42] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1915–1929, Aug. 2013.
[43] Y. Bengio and O. Delalleau, "On the expressive power of deep architectures," in *Algorithmic Learning Theory*. Berlin, Germany: Springer, 2011, pp. 18–36.
[44] R. Salakhutdinov and G. E. Hinton, "Deep Boltzmann machines," in *Proc. Int. Conf. AI Statist.*, Clearwater, FL, USA, 2009, pp. 448–455.
[45] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Proc. Adv. Neural Inf. Process. Syst. 19*. Cambridge, MA, USA, 2007, pp. 153–160.
[46] I. Sutskever and G. E. Hinton, "Deep, narrow sigmoid belief networks are universal approximators," *Neural Comput.*, vol. 20, no. 11, pp. 2629–2636, Nov. 2008.
[47] N. Le Roux and Y. Bengio, "Deep belief networks are compact universal approximators," *Neural Comput.*, vol. 22, no. 8, pp. 2192–2207, Aug. 2010.
[48] G. B. Huang, D. H. Wang, and Y. Lan, "Extreme learning machines: A survey," *Int. J. Mach. Learn. Cybern.*, vol. 2, no. 2, pp. 107–122, Jun. 2011.
[49] H. Chen, J. Peng, Y. Zhou, L. Li, and Z. Pan, "Extreme learning machine for ranking: Generalization analysis and applications," *Neural Netw.*, vol. 53, pp. 119–126, May 2014.
[50] Y. Zhou, J. Peng, and C. Chen, "Extreme learning machine with composite kernels for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, DOI: 10.1109/JSTARS.2014.2359965.
[51] Y.-H. Pao, G.-H. Park, and D. J. Sobajic, "Learning and generalization characteristics of the random vector functional-link net," *Neurocomputing*, vol. 6, no. 2, pp. 163–180, Apr. 1994.
[52] C. L. P. Chen, S. R. LeClair, and Y. H. Pao, "An incremental adaptive implementation of functional-link processing for function approximation, time-series prediction, and system identification," *Neurocomputing*, vol. 18, nos. 1–3, pp. 11–31, Jan. 1998.
[53] C. L. P. Chen and J. Z. Wan, "A rapid learning and dynamic stepwise updating algorithm for flat neural networks and the application to time-series prediction," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 29, no. 1, pp. 62–72, Mar. 1999.
[54] J. Ma, J. Zhao, J. Tian, X. Bai, and Z. Tu, "Regularized vector field learning with sparse approximation for mismatch removal," *Pattern Recognit.*, vol. 46, no. 12, pp. 3519–3532, Dec. 2013.
[55] J. Ma, J. Zhao, J. Tian, A. L. Yuille, and Z. Tu, "Robust point matching via vector field consensus," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1706–1721, Apr. 2014.
[56] J. Ma, J. Zhao, Y. Ma, and J. Tian, "Non-rigid visible and infrared face registration via regularized Gaussian fields criterion," *Pattern Recognit.*, vol. 48, no. 3, pp. 772–784, Mar. 2015.
[57] J. Ma *et al*., "Robust L2E estimation of transformation for non-rigid registration," *IEEE Trans. Signal Process.*, vol. 63, no. 5, pp. 1115–1129, Mar. 2015.
[58] H. Zhou, G.-B. Huang, Z. Lin, H. Wang, and Y. C. Soh, "Stacked extreme learning machines," *IEEE Trans. Cybern.*, DOI: 10.1109/TCYB.2014.2363492.
[59] P. N. Belhumeur, J. P. Hespanha, and D. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.
[60] D. Tao, X. Li, X. Wu, and S. J. Maybank, "Geometric mean for subspace selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 260–274, Feb. 2009.
[61] D. Xu, S. Yan, D. Tao, S. Lin, and H.-J. Zhang, "Marginal Fisher analysis and its variants for human gait recognition and content-based image retrieval," *IEEE Trans. Image Process.*, vol. 16, no. 11, pp. 2811–2821, Nov. 2007.
[62] H. Pu, Z. Chen, B. Wang, and G.-M. Jiang, "A novel spatial–spectral similarity measure for dimensionality reduction and classification of hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 11, pp. 7008–7022, Nov. 2014.
[63] C. Chen *et al*., "Spectral–spatial preprocessing using multihypothesis prediction for noise-robust hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 4, pp. 1047–1059, Apr. 2014.

Yicong Zhou (M'07–SM'14) received the B.S. degree from Hunan University, Changsha, China, in 1992, and the M.S. and Ph.D. degrees from Tufts University, Medford, MA, USA, in 2008 and 2010, all in electrical engineering.

He is currently an Assistant Professor with the Department of Computer and Information Science, University of Macau, Macau, China. His current research interests include multimedia security, image/signal processing, pattern recognition, and medical imaging.

Dr. Zhou was a recipient of the Third Prize of the Macau Natural Science Award in 2014. He is a member of the International Society for Photo-Optical Instrumentations Engineers and Association for Computing Machinery.

Yantao Wei received the B.Sc. degree in information and computing science from the Qingdao University of Science and Technology, Qingdao, China, in 2006, and the M.S. degree in computational mathematics and the Ph.D. degree in control science and engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2008 and 2012, respectively.

He is currently with the School of Educational Information Technology, Central China Normal University, Wuhan. His current research interests include computer vision and pattern recognition.