CNN-Based Multilayer Spatial–Spectral Feature Fusion and Sample Augmentation With Local and Nonlocal Constraints for Hyperspectral Image Classification

Jie Feng, *Member, IEEE*, Jiantong Chen, Liguo Liu, Xianghai Cao, *Member, IEEE*, Xiangrong Zhang, *Senior Member, IEEE*, Licheng Jiao, *Fellow, IEEE*, and Tao Yu

***Abstract*—The extraction of joint spatial–spectral features has been proved to improve the classification performance of hyperspectral images (HSIs). Recently, utilizing convolutional neural networks (CNNs) to learn joint spatial–spectral features has become of great interest. However, the existing CNN models ignore complementary spatial–spectral information among the shallow and deep layers. Moreover, insufficient training samples in HSIs afflict these CNN models with overfitting problem. In order to address these problems, a novel CNN method for HSI classification is proposed. It considers multilayer spatial–spectral feature fusion and sample augmentation with local and nonlocal constraints, which is abbreviated as MSLN-CNN. In MSLN-CNN, a triple-architecture CNN is constructed to extract spatial–spectral features by cascading spectral features to dual-scale spatial features from shallow to deep layers. Then, multilayer spatial–spectral features are fused to learn complementary information among the shallow layers with detailed information and the deep layers with semantic information. Finally, the multilayer spatial–spectral feature fusion and classification are integrated into a unified network, and MSLN-CNN can be optimized in the end-to-end way. To alleviate the small sample size problem, the unlabeled samples having high confidences on local spatial constraint and nonlocal spectral constraint are selected and prelabeled. The nonlocal spectral constraint considers the structure information with spectrally similar samples in the nonlocal searching, while the local spatial constraint utilizes the contextual information with spatially adjacent samples. Experimental results on several hyperspectral datasets demonstrate that the proposed method achieves more encouraging classification performance than the current state-of-the-art classification methods, especially with the limited training samples.**

***Index Terms*—Convolutional neural networks (CNNs), hyperspectral image (HSI) classification, multilayer feature fusion, nonlocal information, spatial–spectral feature extraction.**

I. Introduction

HYPERSPECTRAL remote sensing has played an important role in the field of remote sensing technologies. The imaging spectrometer of hyperspectral remote sensing obtains hundreds of continuous and narrow spectral bands in the range of ultraviolet, visible light, near-infrared, and mid-infrared spectrum [1]. It can reach nanometer-scale spectral resolution. Hyperspectral images (HSIs) record spectral and spatial information of ground objects, which can be viewed as data cubes. In a data cube, each spectral band corresponds to an image with a particular wavelength. Compared with other types of remote sensing images, HSIs provide the potential for more accurate and detailed distinction of different materials and objects. Therefore, HSIs have been applied in various fields, such as military [2], astronomy [3], [4], agriculture [5], and mineralogy [6].

Classification of HSIs is a common technique in different applications. This technique involves two crucial issues: effective feature extraction and advanced classifier design. The traditional feature extraction methods reduce the dimensionality by linearly or nonlinearly transforming the original high-dimensional data into a new low-dimensional space [7]. Linear transformation methods include principal component analysis (PCA) [8], independent component analysis [9], linear discriminant analysis [10], and local Fisher's discriminant analysis (LFDA) [11]. Different atmospheric scattering conditions and intraclass variability make HSIs inherently non-linear [12]. Manifold learning can be used to address the nonlinear problem by seeking the intrinsic manifold structure. Manifold learning-based dimensionality reduction methods can be achieved by constructing a nonlinear mapping to observe certain properties of the manifold. Many manifold learning-based methods, such as isometric feature mapping [13] and local linear embedding [14], have

Manuscript received August 2, 2018; revised November 29, 2018 and January 23, 2019; accepted February 17, 2019. Date of publication March 12, 2019; date of current version April 17, 2019. This work was supported in part by the National Natural Science Foundation of China under Grants 61871306, 61772400, and 61773304, in part by the Project Funded by the China Postdoctoral Science Foundation under Grants 2015M570816 and 2016T90892, in part by the State Key Program of National Natural Science of China under Grant 61836009, in part by the Open Research Fund of Key Laboratory of Spectral Imaging Technology, Chinese Academy of Sciences, under Grant LSIT201803D, in part by the Fundamental Research Funds for the Central Universities under Grant JBX181707, in part by the Postdoctoral Research Program in Shaanxi Province of China, and in part by the Joint Fund of the Equipment Research of Ministry of Education. *(Corresponding author: Jie Feng.)*

J. Feng, J. Chen, L. Liu, X. Cao, X. Zhang, and L. Jiao are with the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, Xidian University, Xi'an 710071, China (e-mail: jiefeng0109@163.com; jiantongchen1123@163.com; liguoliu0619@163.com; xianghaicao@hotmail.com; xrzhang@mail.xidian.edu.cn; lchjiao@mail.xidian.edu.cn).

T. Yu is with the Key Laboratory of Spectral Imaging Technology, Chinese Academy of Sciences, Beijing 100864, China (e-mail: yutao@opt.ac.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/JSTARS.2019.2900705

been applied for HSIs. Another type of nonlinear methods focus on projecting the data from the original space into the kernel-induced space, such as KPCA [15] and KLFDA [16]. To deal with spatial variability of spectral signature, some methods try to incorporate spatial information into consideration. A series of spatial–spectral feature extraction methods were proposed, such as three-dimensional (3-D) Gabor filters [17] and 3-D morphological profile [18].

The features obtained by the feature extraction methods are fed into the classifiers. The representative classifiers include k-nearest neighbors [19], logistic regression [20], support vector machine (SVM) [21], sparse representation-based classification [22], and extreme learning machine [23]. Among these classifiers, SVM seeks to separate the samples with different classes by learning an optimal decision hyperplane. It has shown promising success in HSI classification due to its outstanding ability in dealing with high-dimensional feature space with small sample size.

The above-mentioned feature extraction methods adopt a series of handcrafted features, which highly rely on the experience and parameter setting of massive experts. In the last decade, deep learning methods have been a hot topic. Deep neural networks allow computational models of multiple processing layers to learn data representations with multiple levels of abstraction [24]. Different from traditional feature extraction methods, deep architecture extracts abstract and hierarchical features for classification automatically.

Recently, some deep learning methods have been proposed for HSI classification. Compared with other traditional methods, deep learning method achieves more promising performance for HSI classification. The representative deep learning models include stacked autoencoders (SAEs) [25], [26], deep belief networks (DBNs) [27], and convolutional neural networks (CNNs) [28]–[33]. In [25], a joint spatial–spectral SAE (JSSSAE) network was proposed for HSI classification. JSSSAE uses PCA to compress the original image. Then, it extracts spatial features from the compressed image. Then, the spatial features are flattened to a 1-D vector and cascaded with original spectral features. In [26], a DBN-based HSI classification method was proposed by learning the restricted Boltzmann machine network layer by layer. DBN utilizes a similar neighboring structure with JSSSAE to extract spatial features. JSSSAE and DBN adopt the full connection of different layers, which needs to train a large number of parameters. Moreover, these methods cannot make full use of spatial information. This is because it flattens the spatial information into a vector before the training stage. Compared with SAE and DBN, CNN extracts the spatial information by using local connections to the original data and shared weights to reduce the number of parameters. In [28], Hu *et al.* proposed a 1-D CNN-based method to learn hierarchical spectral features of HSIs.

Recently, some joint spatial–spectral CNN methods were proposed to improve the classification performance of HSIs [29]–[33]. In [29], a contextual CNN (CCNN) method was proposed. It uses a multiscale filter bank in the first convolutional layer to extract spatial and spectral features. Then, joint spatial–spectral features are obtained by concatenating the extracted spatial and spectral features. In [30] and [32], a dual-channel CNN (DCCNN) method was proposed. It utilizes a 1-D CNN to extract spectral features and a 2-D CNN to extract spatial features, and concatenates them together into a softmax regression classifier. However, a CCNN only extracts spatial–spectral features in the shallow layer, while a DCCNN extracts in the deep layer. In [31], a 3-D CNN (3-DCNN) method was proposed to extract spatial–spectral features by using a 3-D convolution operation. A 3-DCNN extracts spatial–spectral features of different layers, but it ignores complementary information among different layers. Moreover, a 3-DCNN has numerous parameters caused by the 3-D convolution operation, which aggravates the overfitting problem. These above-mentioned spatial–spectral CNN methods achieve promising performance in HSI classification when sufficient training samples are provided. For the small sample size problem of CNNs in HSIs, a CNN method based on deep pixel-pair features (PPF-CNN) was proposed [34]. A PPF-CNN constructs an expanded sample set by pairing with any two selected samples from available training samples. A PPF-CNN just reorganizes and relabels the existing training samples.

Some methods combining CNN and recurrent neural network (RNN) [35]–[39] have been developed for HSI classification. Most of these methods use RNN to extract spectral sequence information and CNN to extract spatial information. Compared to 1DCNN, RNN model is able to extract global spectral information. In [39], spatial and spectral features were extracted by replacing the full connection in the long short-term memory model with the convolution operation simultaneously.

In this paper, a novel CNN method based on multilayer spatial–spectral feature fusion and sample augmentation with local and nonlocal constraints (MSLN-CNN) is proposed for HSI classification. In MSLN-CNN, a triple-architecture CNN is constructed, where two architectures are devised to extract spatial features with two different scales and the other is devised to extract spectral features. The joint spatial–spectral features are extracted by cascading spectral features to dual-scale spatial features from shallow to deep layers. The shallow layers focus on detailed and boundary information, while the deep layers learn high-level abstract and semantic information. The multilayer spatial–spectral features are fused to provide complementary information among different hierarchical layers.

These fused spatial–spectral features contain two different scales because of the usage of dual-scale spatial features. Then, these features with different scales are fed into two softmax layers. MSLN-CNN achieves an end-to-end classification by forcing multilayer spatial–spectral feature fusion and softmax-based classification into a unified loss function. Finally, the class label of samples is predicted by the multidecision from the outputs of two softmax layers. To alleviate the small sample size problem, the local spatial constraint and nonlocal spectral constraint are designed to select and prelabel the unlabeled samples with high confidences. These prelabeled samples are used to augment the training set. In the spatial constraint, local contextual information is used to select the spatially adjacent samples. In the spectral constraint, patch-to-patch similarity is used to select the spectrally similar samples in the nonlocal searching.

The main contributions of this paper can be summarized as follows.

1) To make full use of complementary spatial–spectral information among different layers, MSLN-CNN devises multilayer spatial–spectral feature fusion. It can merge the detailed and boundary information of shallow layers and semantic information of deep layers.
2) MSLN-CNN combines multilayer spatial–spectral feature fusion and softmax-based classification into a unified optimization procedure, which can extract complementary spatial–spectral features and implement the classification simultaneously.
3) MSLN-CNN uses not only the local contextual information, but also the structural information in the nonlocal searching to alleviate the small sample size problem of HSIs. In MSLN-CNN, the expanded samples contain new information by introducing extra unlabeled samples. Compared with the original training set, the expanded sample set obtained by MSLN-CNN has higher diversity than that obtained by PPF-CNN.

The rest of this paper is organized as follows. Section II is a detailed description of the proposed MSLN-CNN method, including sample augmentation with nonlocal spectral constraint and local spatial constraint, multilayer spatial–spectral feature fusion, and softmax-based multidecision classification. In Section III, we show the experimental results and analysis on benchmark hyperspectral datasets. Finally, some concluding remarks and suggestions are provided for the further work in Section IV.

II. Proposed MSLN-CNN Method

Compared with other deep learning models, CNN has two special structures: local connection and weight sharing. When faced with computer vision problems, CNN can provide better generalization ability with such special structures. The architecture of CNN is based on the inspirations from neuroscience. A traditional CNN is constructed by stacking several convolutional layers, pooling layers, and full connection layers to form deep architecture. In order to solve HSI classification, a novel MSLN-CNN method is proposed.

The flowchart of the proposed MSLN-CNN method is shown in Fig. 1. As shown in Fig. 1, MSLN-CNN mainly consists of three stages: sample augmentation with nonlocal spectral constraint and local spatial constraint; multilayer spatial–spectral feature fusion; and softmax-based multidecision classification. At the sample augmentation stage, the nonlocal spectral constraint considers the structure information with patch-to-patch spectral similarity in the nonlocal searching, while the local spatial constraint considers the contextual information with spatially adjacent samples. At the stage of multilayer spatial–spectral feature fusion, a triple-architecture CNN is devised, where two architectures are devised to extract various spatial features with dual-scale convolution kernels and the other is devised to extract spectral features with 1×1 convolution kernels. Then, multilayer spatial–spectral features are extracted by cascading spectral features to dual-scale spatial features in all the convolutional layers. To make full use of complementary spatial–spectral information among different layers, multilayer spatial–spectral features are fused. At the stage of softmax-based multidecision classification, complementary spatial–spectral features with different scales are separately fed into the last softmax layer. MSLN-CNN is jointly optimized by combining complementary spatial–spectral feature learning and classification into a unified loss function. Finally, the class labels of samples are predicted by the multidecision from these two softmax layers.

A. Sample Augmentation Based on Local Spatial Constraint and Nonlocal Spectral Constraint

To alleviate the lack of training samples, a novel sample augmentation method based on local spatial constraint and nonlocal spectral constraint is proposed. In HSIs, the samples in a local spatial region generally share similar spectral characteristics, so these samples may have the same label with high probability. Since the samples belonging to the same class may be located in different regions, nonlocal information [40] is also vital for HSI classification. Considering the structural information of current samples, the pixel-to-pixel similarity is extended to patch-to-patch similarity.

1) Local Spatial Constraint: In HSIs, the training samples are represented by , where *m* is the number of training samples. The class labels of training samples are denoted as $\{y_1, y_2, \ldots, y_m\}$. The unlabeled samples are represented by $\{x_1^u, x_2^u, \ldots, x_n^u\}$, where *n* is the number of unlabeled samples. The coordinate positions of $x_i^l\{x_1^l, x_2^l, \ldots, x_m^l\}$ and x_j^μ are indicated as (α_i^l, β_i^l) and $(\alpha_j^\mu, \beta_j^\mu)$, respectively.

For an unlabeled sample x_j^μ, N_j^{spa} represents the set of neighboring samples falling into a window around x_j^μ. $P_k^{\mathrm{spa}}(x_j^\mu)$ indicates k spatial nearest training samples of x_j^μ. π_j^{spa} is defined as the intersection of N_j^{spa} and $P_k^{\mathrm{spa}}(x_j^\mu)$, namely $\pi_j^{\mathrm{spa}} = P_k^{\mathrm{spa}}(x_j^\mu) \cap N_j^{\mathrm{spa}}$. If the set π_j^{spa} exists, the statistical distribution of the class labels of samples in π_j^{spa} is calculated. The class with the most training samples is represented as y_j^{spa}. If the number of these training samples belonging to the class y_j^{spa} is larger than $(k-1)/2$, the unlabeled sample x_j^μ is prelabeled as y_j^{spa} in the local spatial constraint. On the contrary, if the above condition is not satisfied, this unlabeled sample x_j^μ is prelabeled as 0.

The local spatial constraint is formulated as follows:

$$Y_{\mathrm{spa}}\left(x_j^\mu\right) = \begin{cases} y_j^{\mathrm{spa}}, & \text{if num}\left(y_j^{\mathrm{spa}}\right) > (k-1)/2 \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

$$y_j^{\mathrm{spa}} = \begin{cases} \arg\max\limits_c \sum_{x_i^l \in \pi_j^{\mathrm{spa}}} I\left(y_i = c\right), & \text{if } \pi_j^{\mathrm{spa}} \neq \varnothing \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where $\mathrm{num}(y_j^{\mathrm{spa}})$ represents the number of training samples in π_j^{spa} belonging to the class label y_j^{spa}, and $c = 1, 2, \ldots, z$. z represents the number of class labels. $I(\cdot)$ is the indictor function.

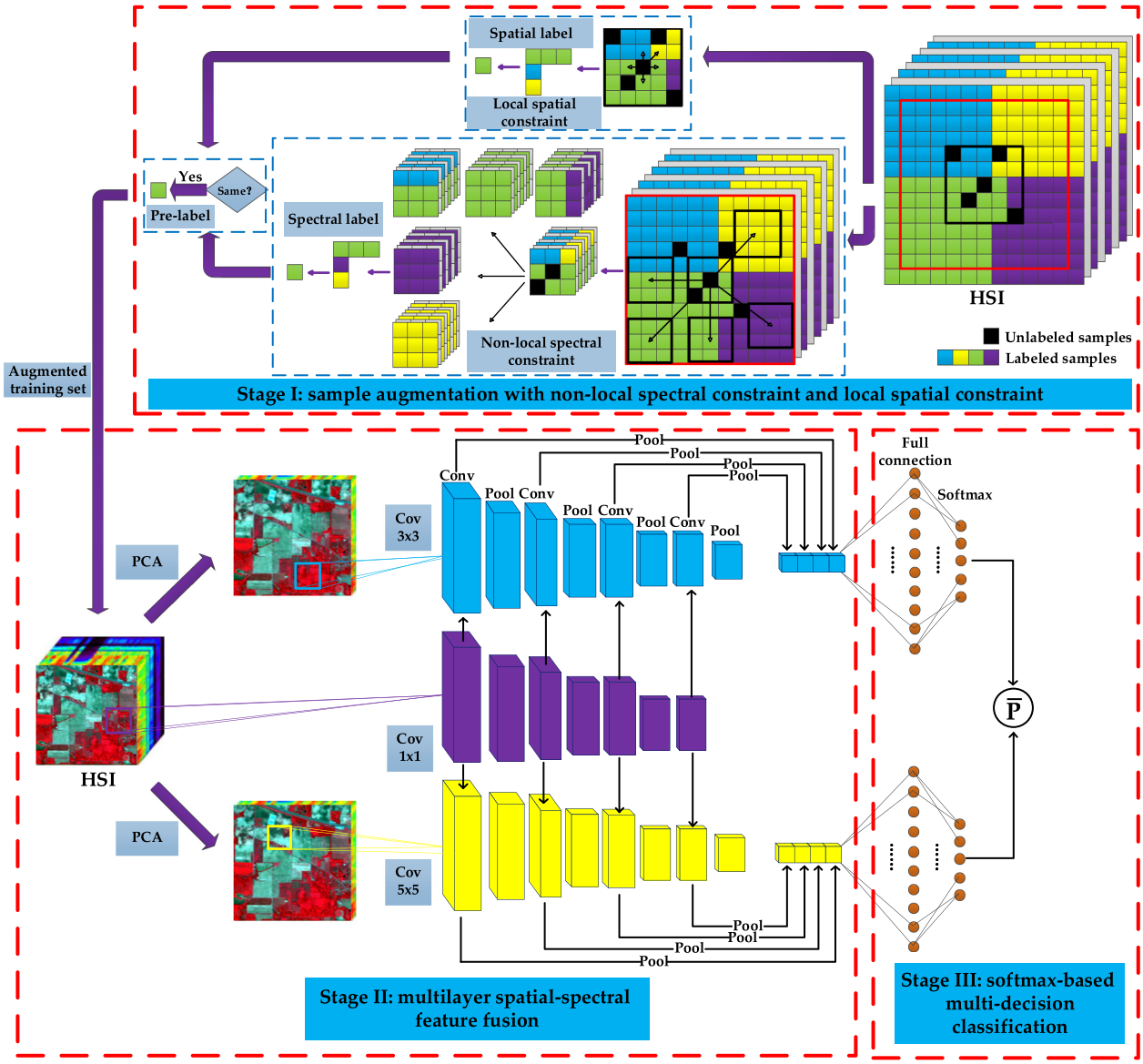


Fig. 1. Flowchart of the proposed MSLN-CNN.

Its value will be one if the condition in the bracket is satisfied, otherwise it will be zero. The spatial similarity $S_{\text{spa}}(x_j^{\mu}, x_i^{l})$ is measured by negative Euclidean distance between the unlabeled sample x_j^{μ} and any training sample x_i^{l} in π_j^{spa}, and $S_{\text{spa}}(x_j^{\mu}, x_i^{l}) = -\|(\alpha_i^{l}, \beta_i^{l}) - (\alpha_j^{\mu}, \beta_j^{\mu})\|$.

2) Nonlocal Spectral Constraint: In the nonlocal spectral constraint, N_j^{spe} represents the sets of neighboring samples falling into a nonlocal search window centered at the unlabeled sample x_j^{μ}. $P_k^{\text{spe}}(x_j^{\mu})$ indicates the set of k spectral nearest training samples of x_j^{μ}. π_j^{spe} is defined as the intersection of N_j^{spe} and $P_k^{\text{spe}}(x_j^{\mu})$, namely $\pi_j^{\text{spe}} = P_k^{\text{spe}}(x_j^{\mu}) \cap N_j^{\text{spe}}$. Similar to the local spatial constraint, the class distribution is calculated. The class containing the most training samples is recorded as y_j^{spe}. If the number of these training samples belonging to y_j^{spe} is larger than $(k-1)/2$, the unlabeled sample x_j^{μ} is selected and prelabeled as y_j^{spe}. On the contrary, the unlabeled sample x_j^{μ} is prelabeled as 0.

The nonlocal spectral constraint is formulated as follows:

$$Y_{\text{spe}}\left(x_j^{\mu}\right) = \begin{cases} y_j^{\text{spe}}, & \text{if num}\left(y_j^{\text{spe}}\right) > (k-1)/2 \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

$$y_j^{\text{spe}} = \begin{cases} \arg\max\limits_{c} \sum_{x_i^{l} \in \pi_j^{\text{spe}}} I\left(y_i = c\right), & \text{if } \pi_j^{\text{spe}} \neq \varnothing \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where the spectral similarity is measured by patch-to-patch similarity $S_{\text{spe}}(x_j^{\mu}, x_i^{l})$. It is calculated as follows:

$$S_{\text{spe}}\left(x_j^{\mu}, x_i^{l}\right) = -\left\|P_i - P_j\right\|_{2,G}^{2} \tag{5}$$

where the patches P_i and P_j represent the square neighbors around the samples x_i^{l} and x_j^{μ}, respectively, and G is the standard Gaussian kernel function.

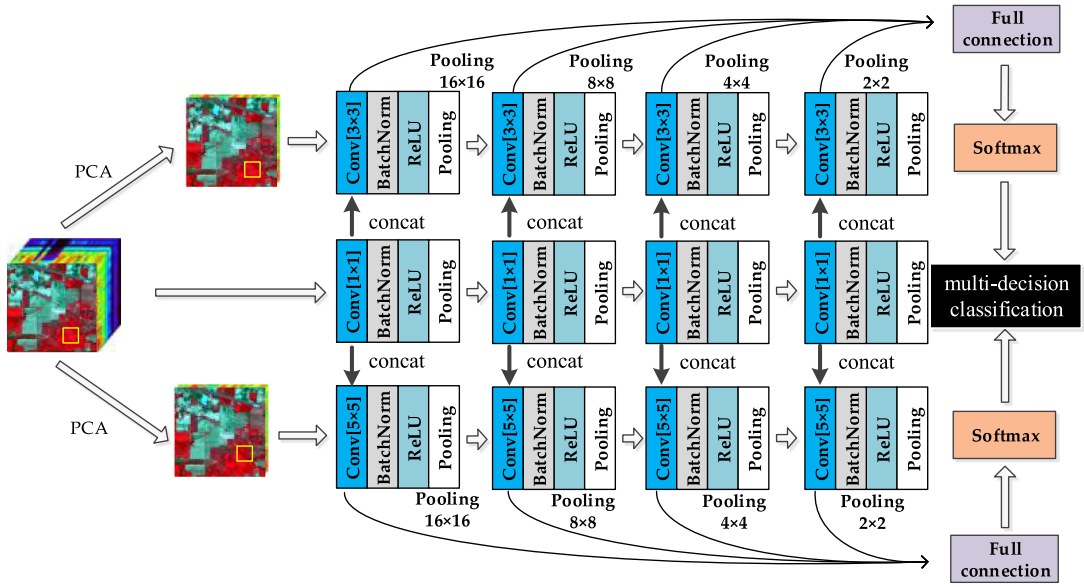


Fig. 2. Detailed architecture of MSLN-CNN.

3) Sample Augment Based on Both Local Spatial Constraint and Nonlocal Spectral Constraint: After the above two constraints, only the unlabeled samples having the same nonzero label in both local spatial constraint and nonlocal spectral constraint are selected. Other unlabeled samples are discarded. The selected unlabeled samples are labeled as $Y_{\mathrm{spa}}(x_j^{\mu})$ or $Y_{\mathrm{spe}}(x_j^{\mu})$, which is calculated as

$$Y\left(x_j^{\mu}\right) = \begin{cases} Y_{\mathrm{spa}}\left(x_j^{\mu}\right) \text{ or } Y_{\mathrm{spe}}\left(x_j^{\mu}\right), & \text{if } Y_{\mathrm{spa}}\left(x_j^{\mu}\right) = Y_{\mathrm{spe}}\left(x_j^{\mu}\right) \neq 0 \\ \text{none}, & \text{otherwise.} \end{cases} \tag{6}$$

B. CNN Based on Multilayer Spatial–Spectral Feature Fusion

HSIs are 3-D data cube, which has spatial and spectral information simultaneously. In HSIs, the spectral signatures of samples belonging to the same class may be different due to varied imaging conditions, e.g., changes in illumination, environment, atmospheric, and temporal conditions. Therefore, joint spatial–spectral feature extraction is critical for HSI classification.

The detailed architecture of MSLN-CNN is shown in Fig. 2. In MSLN-CNN, a triple-architecture CNN is constructed. In the middle architecture, the spatial features are extracted with 1×1 convolution kernels. To extract the spectral information sufficiently, the input of this architecture is spatial windows with all the spectral bands. 1×1 convolutional layers and corresponding max-pooling layers are stacked layer-by-layer. A 1×1 convolutional filter is proposed in network in network [41], which allows complex and learnable interactions of cross-channel information.

For the other two architectures, various spatial features are extracted with dual-scale convolution kernels. The input of dual-scale architectures is spatial windows with several principle components (PCs) by using PCA-based dimensionality reduction. In dual-scale architectures, 3×3 and 5×5 convolutional layers and corresponding max-pooling layers are stacked layer-by-layer. Finally, extracted spatial features are flattened to a 1-D vector, which is used as the input of next fully connected layer. In [42], Li *et al*. extracted the multiscale features of HSIs by Gaussian pyramid decomposition. Compared with the literature [42], dual-scale convolution kernels in MSLN-CNN show outstanding performance on local feature extraction.

Max-pooling layer with a size of 2×2 is used after each convolutional layer. The step of max-pooling layers is 2. The rectified linear unit is used as nonlinear activation function for all the convolutional layers. To improve the stability of MSLN-CNN, batch normalization strategy [43] is used for all the convolutional layers. In order to alleviate the overfitting, dropout strategy [44] is adopted for some hidden layers.

In MSLN-CNN, multilayer spatial–spectral features with different scales are obtained by cascading the spectral features and dual-scale spatial features in all the convolutional layers. Multilayer spatial–spectral features are fused by cascading the features of different layers from shallow to deep, which provides complementary information for classification. To achieve multilayer spatial–spectral feature fusion, the max-pooling layers with sizes of 2×2, 4×4, 8×8, and 16×16 are used to ensure the same size of multilayer spatial–spectral features before feature fusion. Then the outputs of these layers are flattened to the 1-D vector and cascaded to achieve multilayer spatial–spectral feature fusion.

In [45], a hypercolumn CNN (HCCNN) is proposed. In HCCNN, deep and shallow layers are combined by summing the features from different layers. Compared with HCCNN, MSLN-CNN uses cascade operation to preserve the original information of extracted features. Compared with CNN-RNN-based methods [35]–[39], the proposed method considers the fusion of joint spatial–spectral features from shallow to deep layers.

C. MSLN-CNN-Based Classification

These features with different scales are fed into two softmax layers, respectively. The outputs of the softmax layers represent the class probability distribution obtained from features of different scales. A loss function is defined by combining the outputs of two softmax layers. MSLN-CNN is optimized by minimizing the loss function in an end-to-end manner. In the test stage, the class labels of samples are predicted by multidecision from the outputs of these two softmax layers.

In MSLN-CNN, softmax is used for multiclass classification. The output of the softmax function can be used to represent the probability distribution of all the classes, where each entry is in the range of $(0, 1]$, and all the entries add up to 1.

To integrate complementary spatial–spectral feature learning and classifier training into a unified optimization procedure, a new loss function is constructed. The loss function is determined by the cross entropy of the real class probability and the output class probability of MSLN-CNN. The output class probability is calculated by averaging the outputs obtained by different scales of complementary spatial–spectral features. The loss function is defined as

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{c=1}^{z} I\{c = y_i\} \log\left[\left(p_c\left(x_i^l\right) + q_c\left(x_i^l\right)\right)/2\right] \tag{7}$$

where $p_c(x_i^l)$ and $q_c(x_i^l)$ are the probabilities of assigning the ith sample with different scales of complementary features to the cth class, respectively.

At the test stage, the classification results are decided by the class probability distributions obtained from these different scales of features. The labels $\{y_1^{\text{test}}, y_2^{\text{test}}, \ldots, y_s^{\text{test}}\}$ of testing samples $\{x_1^{\text{test}}, x_2^{\text{test}}, \ldots, x_s^{\text{test}}\}$ are predicted by the following equation:

$$y_t^{\text{test}} = \arg\max_c \left[\left(p_c\left(x_t^{\text{test}}\right) + q_c\left(x_t^{\text{test}}\right)\right)/2\right],\ t = 1, 2, \ldots, s. \tag{8}$$

The detailed procedure of MSLN-CNN can be summarized in Algorithm I.

Algorithm 1: The Procedure of the Proposed MSLN-CNN Method.

1. **INPUT**: The training set $\{x_1^l, x_2^l, \ldots, x_m^l\}$ and test set $\{x_1^{\text{test}}, x_2^{\text{test}}, \ldots, x_s^{\text{test}}\}$ from z classes, the class labels of training samples $\{y_1, y_2, \ldots, y_m\}$, minibatch size, the number of training epochs, the number of PCs r and the size of spatial windows p
2. **Begin**
3. Prelabel the unlabeled samples with local spatial constraint and nonlocal spectral constraint by (6)
4. Select the prelabeled samples to add in training set as $\{x_1^l, \ldots, x_m^l, x_g^u, \ldots, x_h^u\}$, the corresponding class labels are updated as $\{y_1, \ldots, y_m, Y(x_g^\mu), \ldots, Y(x_h^\mu)\}$
5. Apply PCA to the HSI and reserve the first r PCs
6. **Initialize**: The weights and biases are randomly initialized. They obey to the Gaussian distribution with mean 0 and standard deviation 0.1.
7. Input the spatial windows with all the spectral bands and with r PCs into the triple architectures of MSLN-CNN.
8. **for** every epoch
9. **for** training sample of every minibatch
10. compute the unified loss function (7)
11. update the parameters of MSLN-CNN by minimizing (7)
12. **end for**
13. **end for**
14. Calculate the labels of test set $\{x_1^{\text{test}}, x_2^{\text{test}}, \ldots, x_s^{\text{test}}\}$ by (8)
15. **END**
16. **OUTPUT**: the labels of the test samples classified by the trained MSLN-CNN

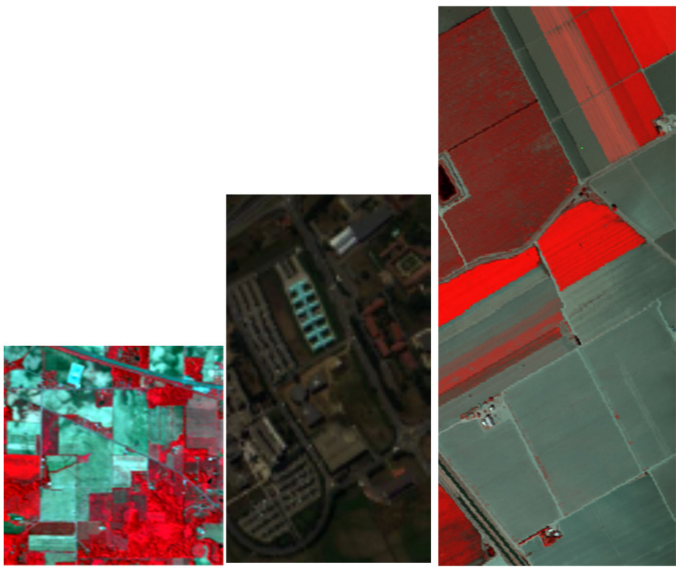
III. Experimental Results

In this section, we validate the performance of the proposed MSLN-CNN method on three benchmark hyperspectral datasets and compare with some state-of-the-art HSI classification approaches.

A. Data Description

Experiments are conducted on the hyperspectral datasets of the Indian Pines, Pavia University, and Salinas. The detailed descriptions of these datasets are listed as follows.

1) The Indian Pines dataset was collected by the airborne visible/infrared imaging spectrometer sensor (AVIRIS) over the Indian Pines test site in June 1992. It consists of 145×145 pixels and has 220 spectral bands. The dataset has 20 m per pixel spatial resolutions and 10 nm spectral resolutions covering a spectrum range of 200–2400 nm. In the experiment, 200 spectral bands were used by removing the absorption bands [100–104], [150–163], and 220. The ground truth is composed of 16 vegetation classes. The false-color composite image (bands 50, 27, 17) is shown in Fig. 3(a).
2) The Pavia University dataset was collected by the reflection optical system imaging spectrometer during a flight campaign over Pavia, Northern Italy. It consists of 610×340 pixels, and has a spatial resolution of 1.3 m per pixel. Removing 12 noise bands, 103 bands are retained in the experiments. The image contains nine classes. The false-color composite image (bands 53, 31, 8) is shown in Fig. 3(b).
3) The Salinas dataset was collected by the 224-band AVIRIS sensor over Salinas Valley, California. In this experiment, 204 bands are retained after removing 20 water absorption bands: [108–112], [154–167], and 224. The dataset consists of 512×217 pixels, and has a spatial resolution of 3.7 m per pixel. The ground truth contains 16 classes. Fig. 3(c) shows a false-color composite image (bands 50, 170, 190).



(a) (b) (c)

Fig. 3. False-color composite images. (a) Indian Pines. (b) Pavia University. (c) Salinas.

TABLE I
16 CLASSES OF THE INDIAN PINES IMAGE AND THE NUMBERS OF TRAINING AND TEST SAMPLES FOR EACH CLASS

Class		Number of samples	
No	Name	Training	Test
1	Alfalfa	2	42
2	Corn-notill	71	1286
3	Corn-mintill	42	746
4	Corn	12	213
5	Grass-pasture	24	435
6	Grass-trees	36	658
7	Grass-pasture-mowed	1	26
8	Hay-windrowed	24	430
9	Oats	1	18
10	Soybean-notill	49	874
11	Soybean-mintill	123	2209
12	Soybean-clean	30	533
13	Wheat	10	185
14	Woods	63	1139
15	Buildings-Grass-Trees-Drives	19	348
16	Stone-Steel-Towers	5	83
Total		512	9225

B. Experimental Setting

In order to verify the classification performance of the proposed MSLN-CNN method, five representative methods based on deep learning for HSI classification, JSSSAE [25], DBN [27], CNN [28], PPF-CNN [34], and 3-DCNN [31] are used as the comparison methods. In addition, the classical SVM with radial basis function (RBF-SVM) [21] is also used as a comparison method. The classification performance of all these methods is measured by three popular indexes: overall accuracy (OA), average accuracy (AA), and Kappa coefficient (Kappa). All the experimental results are obtained by running 30 times independently with a random division for training and test sets. All the experiments are implemented using Python language and tensorflow library [46]. TensorFlow is an open source software library for numerical computation using data flow graphs. A NVIDIA 1080Ti graphics card is used to implement GPU computation.

For RBF-SVM, one-against-all strategy is used to deal with multiclass classification tasks. In RBF-SVM, the penalty and gamma parameters are determined by fivefold cross validation. For JSSSAE and DBN, the radius of spatial neighborhood window is searched in the range of [3, 21] with the interval of 2. For CNN, as suggested by the literature [28], the input of the spatail window is set as 5×5. For PPF-CNN, the size of the block window of neighboring pixels is set to the default value in [34]. For 3-DCNN, the spatial window size of 3-D input is resized to 27×27 [31]. For MSLN-CNN, the spatial window size of the input is 27×27. Suggested by Qian and Ye [47], a 5×5 patch is chosen in nonlocal spectral constraint of MSLN-CNN. The MSLN-CNN network uses a minibatch gradient descent to guide the training process. In the training process of MSLN-CNN, the batch size is 128, the learning rate is 0.01, and the number of iterations is 1000.

C. Classification Results of Hyperspectral Datasets

1) Classification Results of the Indian Pines Dataset: In the Indian Pines dataset, 5% samples from each class are randomly selected as the training set. The unlabeled samples with the same number as the training samples are selected from prelabeled unlabeled samples. The remaining samples are used for test. The numbers of training and test samples are shown in Table I.

Table II records the average classification accuracies and the corresponding standard deviations of the seven algorithms over 30 independent runs. In Table II, the first 16 rows correspond to the results of each class, and the last three rows are the results of OA, AA, and Kappa for all the classes. In the seven algorithms, the best classification results are highlighted in gray regions. As shown in Table II, deep learning-based methods, JSSSAE, DBN, CNN, PPF-CNN, 3-DCNN, and MSLN-CNN are superior to RBF-SVM due to the ability of hierarchical nonlinear feature extraction. Compared with JSSSAE, DBN, and CNN, PPF-CNN achieves better classification results because of its enlargement of available training samples. Compared with PPF-CNN, 3-DCNN improves the classification performance by combining spatial and spectral features. Among the seven methods, MSLN-CNN achieves the best classification results in most classes because of effective sample augmentation and multilayer spatial–spectral feature fusion. It is worth noting that MSLN-CNN achieves completely correct classification for both Hay-windrowed and Woods classes. In addition, compared with other methods, MSLN-CNN improves at least 5.4% in the OA index, 5% in the AA index and 6% in the Kappa index.

Fig. 4 shows the classification visual maps of the seven algorithms on the Indian Pines dataset. As shown in Fig. 4(b)–(f), RBF-SVM, JSSSAE, DBN, CNN, and PPF-CNN misclassify many samples in the middle of regions, especially in the corn-notill, corn-mintill, soybean-notill, and soybean-mintill classes. To some extent, the misclassification leads to some visual noisy scattered points. Compared with these methods, 3-DCNN and

TABLE II
CLASSIFICATION RESULTS OF RBF-SVM, JSSAE, DBN, CNN, PPF-CNN, 3-DCNN, AND MSLN-CNN ON THE INDIAN PINES DATASET

Class	RBF-SVM	JSSSAE	DBN	CNN	PPF-CNN	3DCNN	MSLN-CNN
1	6.1±11.2	10.0±6.4	13.6±5.6	78.4±10.2	50.4±8.4	83.8±13.4	93.6±5.7
2	72.9±3.6	79.7±2.3	79.8±2.9	75.4±2.4	89.2±2.1	92.7±3.5	97.5±1.1
3	58.0±3.6	74.9±4.8	70.5±2.2	82.8±3.3	77.1±2.7	87.2±10.4	97.5±0.8
4	39.0±15.0	62.8±8.3	71.3±6.6	89.2±3.5	87.7±3.7	83.4±8.3	98.5±1.7
5	87.0±4.5	84.2±3.3	80.1±4.1	69.0±4.6	92.7±1.0	84.0±5.7	95.0±1.0
6	92.4±2.0	94.3±1.7	94.2±2.4	92.8±2.5	93.1±1.9	93.4±2.5	99.6±0.5
7	0±0	24.4±18.8	28.1±22.6	51.1±12.3	0±0	97.2±4.8	79.3±19.2
8	98.1±1.4	98.8±0.4	98.5±1.5	97.1±1.6	99.6±0.3	97.4±2.8	100.0±0
9	0±0	11.1±10.1	9.5±2.4	41.6±9.9	0±0	77.0±11.1	66.3±22.5
10	65.8±3.7	73.6±3.8	73.2±4.7	81.0±2.6	85.6±2.8	93.3±5.0	98.1±1.3
11	85.3±2.9	83.4±2.0	82.7±2.2	87.2±1.5	83.8±1.6	94.9±2.7	99.2±0.4
12	69.6±6.5	70.4±8.0	62.0±5.8	84.4±2.3	90.4±3.1	89.8±4.3	94.6±3.5
13	92.3±4.1	94.2±4.3	89.7±10.6	83.1±4.2	97.8±0.9	92.8±5.9	99.8±0.3
14	96.6±1.0	94.2±1.5	94.4±1.6	98.2±0.8	95.5±1.1	98.3±1.3	100.0±0
15	41.7±7.0	66.1±5.6	64.2±6.5	84.7±4.5	78.0±2.4	77.8±13.4	97.3±3.5
16	75.2±9.0	87.6±8.1	80.5±13.2	76.0±8.1	97.3±1.3	88.4±5.3	94.5±4.0
OA（%）	77.8±0.8	81.9±0.1	80.6±0.1	85.4±0.8	87.9±0.8	92.8±0.8	98.2±0.5
AA（%）	61.3±1.4	69.4±1.9	68.3±1.7	79.4±1.6	76.5±0.6	89.4±1.4	94.4±3.0
Kappa（%）	74.5±1.0	79.3±1.1	77.8±1.3	84.3±2.9	86.3±0.9	91.9±0.9	97.9±0.5

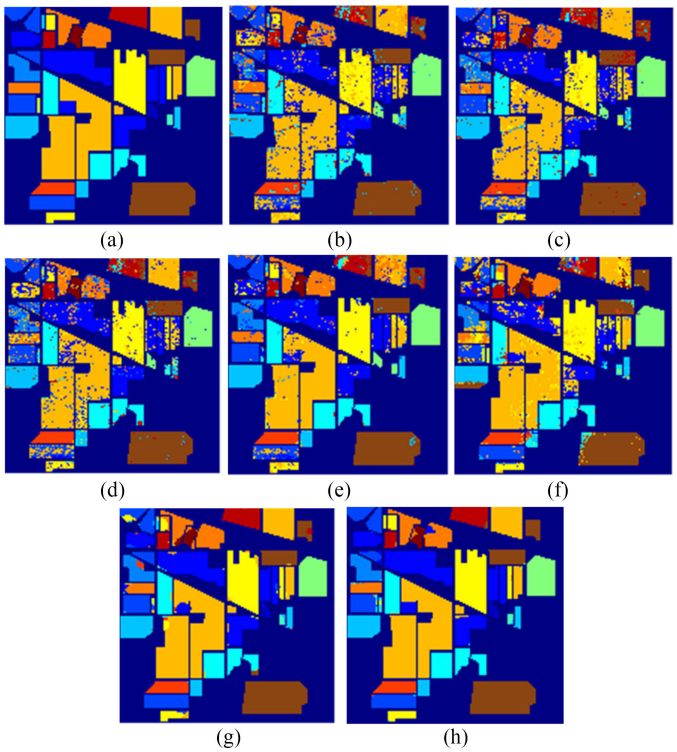


Fig. 4. (a) Ground truth and (b)–(h) classification visual maps of the Indian Pines dataset by RBF-SVM, JSSAE, DBN, PPF-CNN, CNN, 3-DCNN, and MSLN-CNN, respectively.

MSLN-CNN significantly improve the uniformity of regions. Compared with 3-DCNN, MSLN-CNN achieves better regional homogeneity in the soybean-clean and grass-trees classes, and better boundary localization in the Corn-notill class.

2) Classification Results of the Pavia University Dataset: In the Pavia University dataset, 3% samples from each class are randomly selected for training. The unlabeled samples with the same number as the training samples are selected for sample augmentation. The remaining samples are selected for test. The numbers of training and test samples for each class are given in Table III.

TABLE III
NINE CLASSES OF THE PAVIA UNIVERSITY IMAGE AND THE NUMBERS OF TRAINING AND TEST SAMPLES FOR EACH CLASS

Class		Number of samples	
No	Name	Training	Test
1	Asphalt	199	6233
2	Meadows	559	17531
3	Gravel	63	1973
4	Trees	92	2880
5	Painted metal sheets	40	1265
6	Bare Soil	151	4727
7	Bitumen	40	1250
8	Self-Blocking Bricks	110	3462
9	Shadows	28	891
Total		1282	40212

The results of statistical classification on the Pavia University dataset are summarized in Table IV. As shown in Table IV, CNN, PPF-CNN, 3-DCNN, and MSLN-CNN are superior to RBF-SVM, JSSSAE, and DBN by extracting spatial information with local connections and reducing network parameters with weight sharing. For the gravel class, the classification results of RBF-SVM, JSSSAE, DBN, and CNN are not satisfying. Compared with these four algorithms, MSLN-CNN has increased by 39.2%, 27.3%, 29.7%, and 21.5%, respectively. For all the classes, the classification accuracy of MSLN-CNN is over 92%. In particular, MSLN-CNN achieves completely correct classification results in both meadows and painted metal sheets classes. Among the seven algorithms, MSLN-CNN obtains the best statistical results in terms of the OA, AA, and Kappa indexes.

Fig. 5 shows the classification visual maps of the seven algorithms on the Pavia University dataset. As shown in Fig. 5(b)–(f), many samples belonging to the bitumen class are misclassified as the asphalt class due to similar spectral characteristics. The proposed MSLN-CNN method provides a better distinction

TABLE IV
CLASSIFICATION RESULTS OF RBF-SVM, JSSAE, DBN, CNN, PPF-CNN, 3-DCNN, AND MSLN-CNN ON THE PAVIA UNIVERSITY DATASET

Class	RBF-SVM	JSSSAE	DBN	CNN	PPF-CNN	3DCNN	MSLN-CNN
1	90.7±1.1	92.3±1.1	91.6±0.8	93.1±1.4	98.0±0.1	95.5±1.2	99.8±0.1
2	96.8±0.7	97.6±0.3	97.4±0.4	97.6±0.9	99.2±0.2	99.4±0..3	100.0±0
3	60.2±5.4	72.1±3.5	69.7±6.0	77.9±4.5	84.9±1.8	92.6±5.4	99.4±0.7
4	90.8±2.0	90.9±1.4	91.2±1.4	86.4±3.6	95.8±0.8	75.2±4.9	92.6±2.4
5	98.8±0.4	98.7±0.4	98.6±0.6	98.5±1.4	99.8±0.1	95.4±4.3	100.0±0
6	79.5±4.9	86.9±1.9	85.6±2.2	91.0±2.8	96.4±0.3	99.4±0.6	99.3±0.7
7	74.3±5.1	78.1±4.9	74.8±4.8	81.2±2.9	89.2±0.8	91.5±3.4	99.4±0.7
8	88.8±2.2	87.8±1.4	88.2±1.3	92.5±2.2	93.7±1.2	94.8±1.4	99.7±0.4
9	99.8±0.1	99.5±0.3	99.6±0.1	79.0±4.1	98.5±0.7	77.4±2.8	92.5±1.8
OA（%）	90.3±0.6	92.4±0.3	91.9±0.3	93.0±0.6	96.9±0.2	95.2±0.7	99.1±0.2
AA（%）	86.6±0.9	89.3±0.7	88.5±0.8	88.6±0.8	95.1±0.2	91.2±1.1	98.1±0.1
Kappa（%）	87.1±0.8	89.9±0.4	89.2±0.4	90.7±0.7	96.0±0.2	93.8±0.9	98.9±0.2

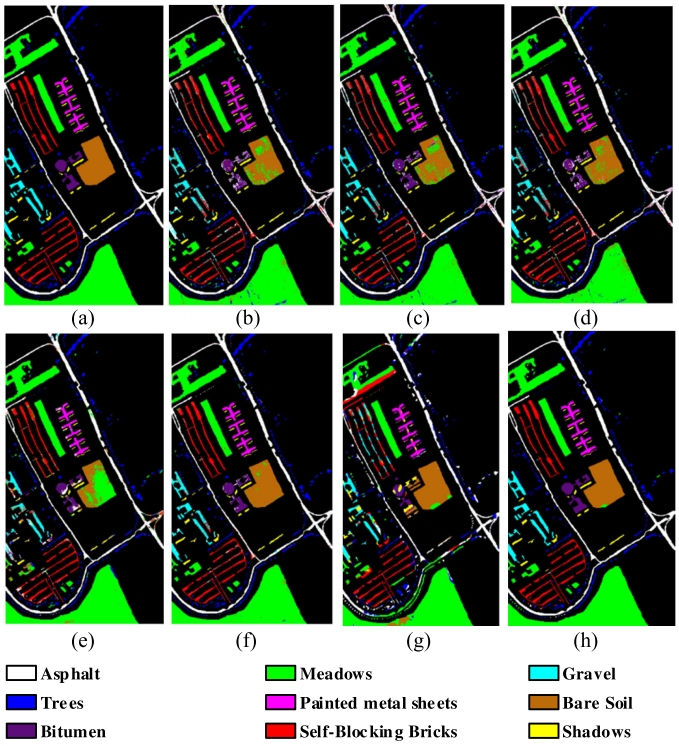


Fig. 5. (a) Ground truth and (b)–(h) classification visual maps of the Pavia University dataset by RBF-SVM, JSSAE, DBN, PPF-CNN, CNN, 3-DCNN, and MSLN-CNN, respectively.

between these two classes. Compared with other methods, MSLN-CNN achieves better regional uniformity in the bare soil class. In addition, MSLN-CNN obtains better boundary localization in the meadows class.

3) Classification Results on the Salinas Dataset: In the Salinas dataset, 1% samples from each class are randomly selected as the training set. The unlabeled samples with the same number as the training samples are selected for sample augmentation. The remaining samples are used as the test set. The numbers of each class in training and test samples are shown in Table V.

The classification results of the seven algorithms are listed in Table VI. It can be seen that all the seven algorithms exceed 90% classification accuracy in most classes. However, RBF-SVM, JSSSAE, DBN, CNN, and PPF-CNN misclassify many

TABLE V
16 CLASSES OF THE SALINAS IMAGE AND THE NUMBERS OF TRAINING AND TEST SAMPLES FOR EACH CLASS

Category		Number of samples	
No	Name	Training	Test
1	Brocoli_green_weeds_1	20	1969
2	Brocoli_green_weeds_2	37	3652
3	Fallow	20	1936
4	Fallow_rough_plow	14	1366
5	Fallow_smooth	27	2624
6	Stubble	40	3879
7	Celery	36	3507
8	Grapes_untrained	113	11045
9	Soil_vinyard_develop	62	6079
10	Corn_senesced_green	33	3212
11	Lettuce_romaine_4wk	11	1046
12	Lettuce_romaine_5wk	19	1889
13	Lettuce_romaine_6wk	9	898
14	Lettuce_romaine_7wk	11	1048
15	Vinyard_untrained	73	7122
16	Vinyard_vertical	18	1771
Total		543	53043

samples in the vinyard_untrained class. Compared with these methods, MSLN-CNN obviously improves the classification results. MSLN-CNN achieves absolutely correct classification results in the fallow and soil_vinyard_develop classes. Compared with other methods, MSLN-CNN achieves higher classification accuracy in most classes and obtains better statistical results in terms of the OA, AA, and Kappa indexes.

Fig. 6 shows the classification visual maps of the seven algorithms on the Salinas datasets. As shown in Fig. 6(b)–(f), many samples belonging to the grapes_untrained class are misclassified as the vinyard_untrained class by RBF-SVM, JSSSAE, DBN, CNN, and PPF-CNN. Compared with them, 3-DCNN and MSLN-CNN provide a better distinction between these two classes. Moreover, compared with 3-DCNN, MSLN-CNN achieves better uniformity in fallow and vinyard_vertical_trellis classes.

D. Investigation on Running Time

Tables VII–IX list the training time and test time for the seven algorithms on the Indian Pines, Pavia University, and Salinas datasets, respectively. As shown in Tables VII–IX,

TABLE VI
CLASSIFICATION RESULTS OF RBF-SVM, JSSAE, DBN, CNN, PPF-CNN, 3-DCNN, AND MSLN-CNN ON THE SALINAS DATASET

Class	RBF-SVM	JSSSAE	DBN	CNN	PPF-CNN	3DCNN	MSLN-CNN
1	97.4±1.5	97.9±0.5	98.5±0.8	93.3±8.7	98.5±0.5	99.4±0.6	99.1±1.5
2	99.7±0.2	99.1±0.5	98.9±0.2	97.4±1.2	99.7±0.2	99.4±0.6	99.9±0.2
3	93.7±1.5	95.3±0.6	97.5±0.1	86.4±4.1	99.8±0.1	98.8±1.8	100.0±0.0
4	97.8±1.3	99.5±0.6	99.0±0.3	98.2±1.8	99.7±0.2	98.8±1.9	95.6±3.5
5	97.5±1.1	98.5±0.4	97.5±0.2	98.1±1.0	96.8±0.2	99.0±0.7	99.7±0.5
6	99.5±0.3	99.9±0.1	99.3±0.1	99.9±0.2	99.8±0.3	99.8±0.3	99.8±0.2
7	99.3±0.2	99.2±0.1	99.0±0.3	99.0±0.9	99.5±0.2	97.9±2.5	99.8±0.3
8	88.9±2.9	82.7±0.7	83.0±1.4	88.4±2.8	89.9±0.9	96.7±2.2	96.5±0.9
9	99.2±0.3	99.2±0.1	99.0±0.1	95.1±0.7	99.8±0.2	99.8±0.3	100.0±0
10	88.8±1.9	88.9±0.9	92.8±0.1	93.6±2.4	88.3±2.7	98.2±2.1	99.7±0.4
11	87.5±4.6	93.6±7.0	91.7±0.1	97.6±1.0	93.4±2.9	97.3±3.8	98.5±0.4
12	98.0±2.3	98.6±1.0	99.0±0.1	98.9±1.1	99.7±0.7	98.5±2.1	98.4±0.6
13	98.0±0.8	99.2±0.7	99.3±0.2	94.7±2.4	98.6±0.7	97.4±5.4	96.8±1.6
14	89.6±2.6	94.8±0.2	92.0±7.4	92.5±3.9	92.3±1.7	98.9±1.0	98.3±1.0
15	53.9±7.6	75.9±2.4	69.5±0.2	80.1±5.3	72.9±2.5	96.5±1.5	99.2±0.5
16	90.8±5.0	96.1±1.9	96.1±2.5	93.6±2.6	95.7±1.8	94.7±5.1	98.0±1.1
OA（%）	89.3±0.7	91.5±0.1	90.8±0.6	92.3±1.2	92.8±0.4	97.9±0.4	98.7±0.2
AA（%）	92.5±0.6	94.9±0.2	94.5±0.8	94.2±0.9	95.5±0.7	98.0±0.6	98.7±0.3
Kappa（%）	88.0±0.8	90.6±0.4	89.7±0.7	91.4±1.4	91.9±0.4	97.7±0.5	98.6±0.3

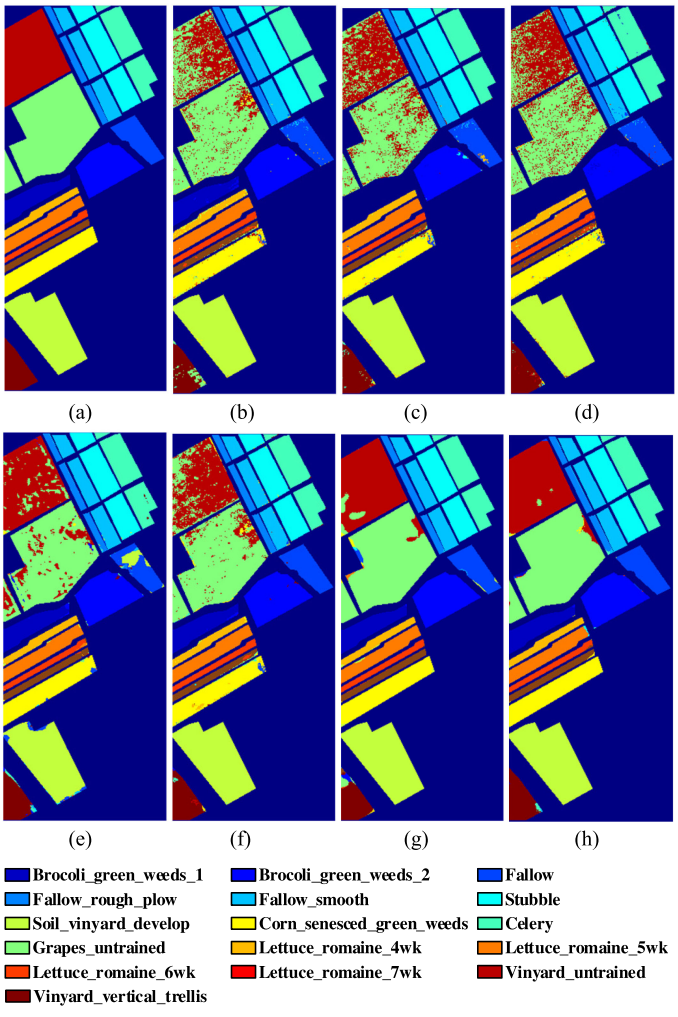


Fig. 6. (a) Ground truth and (b)–(h) classification visual maps of the Salinas dataset by RBF-SVM, JSSAE, DBN, PPF-CNN, CNN, 3-DCNN, and MSLN-CNN, respectively.

TABLE VII
RUNNING TIME OF RBF-SVM, JSSAE, DBN, CNN, PPF-CNN, 3-DCNN, AND MSLN-CNN ON THE INDIAN PINES DATASET

Dataset	Method	Training Time (s)	Test Time (s)
Indian Pines	RBF-SVM	0.4±0.1	1.2±0.1
	JSSSAE	76.3±8.4	0.2±0.1
	DBN	114.3±20.1	0.2±0.1
	CNN	220.7±27.9	0.5±0.1
	PPF-CNN	2056.0±36.7	5.3±0.3
	3DCNN	2690.2±57.9	16.0±0.1
	MSLN-CNN	458.2±13.5	0.8±0.1

TABLE VIII
RUNNING TIME OF RBF-SVM, JSSAE, DBN, CNN, PPF-CNN, 3-DCNN, AND MSLN-CNN ON THE PAVIA UNIVERSITY DATASET

Dataset	Method	Training Time (s)	Test Time (s)
Pavia University	RBF-SVM	0.5±0.1	3.5±0.1
	JSSSAE	82.2±5.3	0.3±0.1
	DBN	147.0±10.6	0.4±0.2
	CNN	371.8±15.3	1.2±0.1
	PPF-CNN	4367.9±29.5	7.2±0.4
	3DCNN	1979.0±12.6	31.4±5.5
	MSLN-CNN	885.4±22.0	4.0±0.3

TABLE IX
RUNNING TIME OF RBF-SVM, JSSAE, DBN, CNN, PPF-CNN, 3-DCNN, AND MSLN-CNN ON THE SALINAS DATASET

Dataset	Method	Training Time (s)	Test Time (s)
Salinas	RBF-SVM	0.4±0.1	2.7±0.1
	JSSSAE	70.1±2.4	0.6±0.1
	DBN	102.6±9.1	0.5±0.2
	CNN	165.1±2.1	0.7±0.1
	PPF-CNN	1940.1±17.4	64.6±1.5
	3DCNN	1157.7±25.7	28.1±0.4
	MSLN-CNN	320.4±6.7	2.3±0.2

compared with RBF-SVM, six deep learning-based methods, JSSSAE, DBN, PPF-CNN, CNN, 3-DCNN, and MSLN-CNN, cost more training time on the construction of deep network models. JSSSAE and DBN are faster on the training time than PPF-CNN, CNN, 3-DCNN, and MSLN-CNN due to the input of 1-D vector. Among all the comparison methods, PPF-CNN and 3-DCNN are time-consuming on the training time. A 3-DCNN takes more time due to the increasing parameters of the network caused by 3-D convolution operations. PPF-CNN takes more time because of the expansion of a large number of training samples, especially when the number of training samples is large. MSLN-CNN is faster on the training time than 3-DCNN and PPF-CNN.

For the test time, JSSSAE, DBN, CNN, and MSLN-CNN have more obvious advantages than RBF-SVM, PPF-CNN, and 3-DCNN. PPF-CNN is slower due to the usage of voting strategy with the surrounding samples. A 3-DCNN takes more time due to the usage of complex 3-D convolutions. MSLN-CNN only costs 0.8, 4.0, and 2.3 s on the Indian Pines, Pavia University, and Salinas datasets, respectively.

E. Sensitivity to the Number of Training Samples

Fig. 7(a)–(c) records the classification results of the seven algorithms with different ratios of training samples. Specifically, 1%, 3%, 5%, 7%, and 9% samples from each class on the Indian Pines datasets, 1%, 2%, 3%, 4%, and 5% on the Pavia University datasets, and 1%, 1.5%, 2%, 2.5%, and 3% on the Salinas datasets are randomly selected as the training samples. Generally, deep learning-based methods are usually heavily parameterized and a large number of training samples are required to guarantee the performance. When the ratio of training samples decreases, the classification performance of all the seven algorithms declines. Compared with RBF-SVM, JSSSAE, DBN, CNN, PPF-CNN, and 3-DCNN, MSLN-CNN consistently provides superior performance with different ratios of training samples. Additionally, MSLN-CNN declines more slower than other algorithms with less than 3% training samples on all the three datasets. Thus, MSLN-CNN is a better choice when the number of training samples is limited.

F. Performance Analysis to the Challenging Dataset

We have added the experiment on the Pavia University dataset with 3921 training samples and 42 776 test samples. Table X lists the number of training and test samples on the Pavia University dataset. The classification results of the Pavia University dataset are summarized in Table XI.

As shown in Table XI, classification of the Pavia University dataset with fixed training and test sets is a challenge. CNN, PPF-CNN, 3-DCNN, and MSLN-CNN are superior to SVM, JSSSAE, and DBN due to local connection and weight sharing. Compared with RBF-SVM, JSSSAE, DBN, CNN, and 3-DCNN, PPF-CNN has higher classification accuracy due to the usage of pixel-pair sample augmentation. Among the seven methods, MSLN-CNN achieves the best classification results due to multilayer spatial–spectral feature fusion.

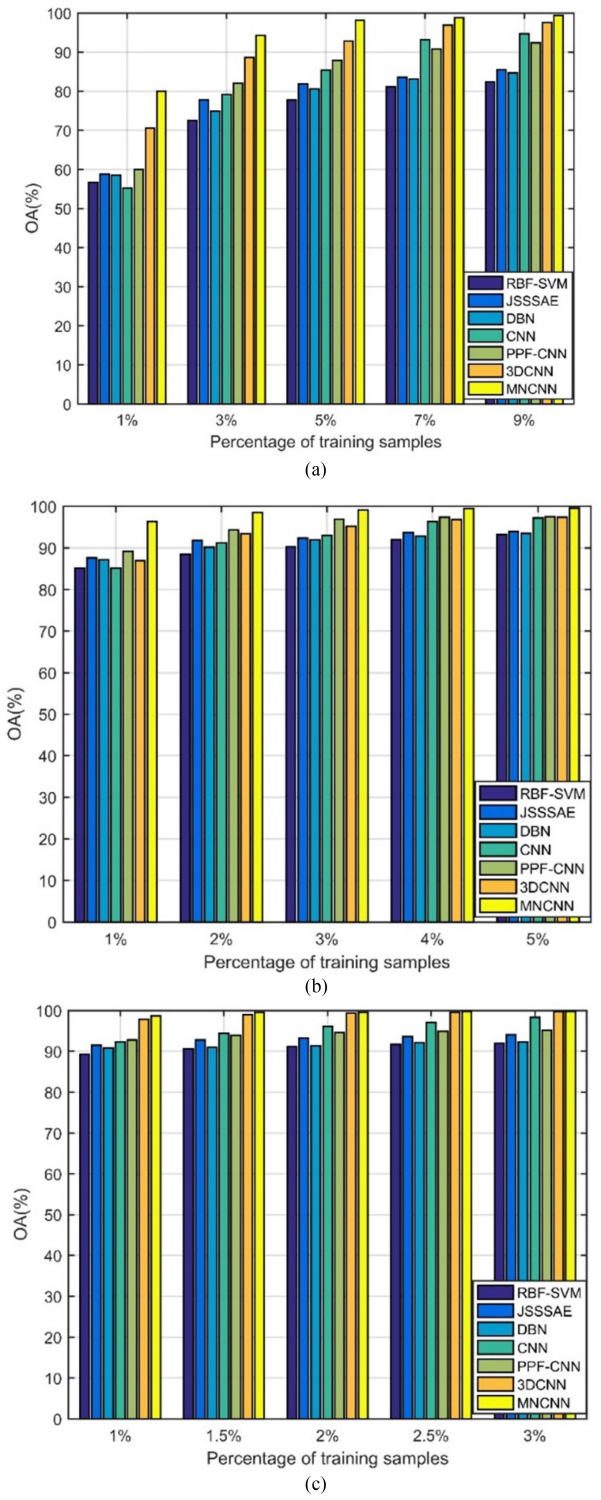


Fig. 7. OA results of RBF-SVM, JSSAE,DBN, CNN, PPF-CNN, 3-DCNN, and MSLN-CNN with different ratios of training samples on (a) the Indian Pines, (b) the Pavia University, and (c) the Salinas datasets.

G. Comparison With Other Classification Techniques

In Table XII, four representative methods, the fusion of deep and shallow features into 3-DCNN (3-DCNN-FDS), the convolutional recurrent neural network (CRNN) [35], the

TABLE X
NUMBER OF TRAINING AND TEST SAMPLES ON THE PAVIA UNIVERSITY DATASET WITH THE AVAILABLE TRAINING AND TEST SETS

Category		Number of samples	
No	Name	Training	Test
1	1.Asphalt	548	6631
2	2.Meadows	540	18649
3	3.Gravel	392	2099
4	4.Trees	524	3064
5	5.Painted metal sheets	265	1345
6	6.Bare Soil	532	5029
7	7.Bitumen	375	1330
8	8.Self-Blocking Bricks	514	3682
9	9.Shadows	231	947
Total		3921	42776

TABLE XI
CLASSIFICATION RESULTS BY RBF-SVM, JSSSAE, DBN, CNN, PPF-CNN, 3-DCNN, AND MSLN-CNN ON THE PAVIA UNIVERSITY DATASET WITH THE AVAILABLE TRAINING AND TEST SETS

Methods	Pavia University Dataset		
	OA (%)	AA (%)	Kappa (%)
RBF-SVM	77.1±0.5	84.0±0.8	71.3±0.7
JSSSAE	80.6±0.7	86.3±0.6	75.3±0.7
DBN	82.1±0.4	85.4±0.6	76.7±0.6
CNN	83.3±0.9	85.2±0.8	78.1±1.0
PPF-CNN	86.5±0.3	87.2±0.4	82.4±0.4
3DCNN	84.2±1.0	82.5±0.9	79.5±1.1
MSLN-CNN	88.1±0.9	79.4±1.0	84.4±1.0

spectral–spatial residual network (SSRN) [48], and the multigrained network (MugNet) [49], are used for comparison.

As shown in Table XII, compared with CRNN, 3-DCNN-FDS increases by 4.1%, 2.6%, and 3.4% in terms of OA index on the three datasets. CRNN uses 1DCNN and RNN to extract spectrally contextual information. The classification result of CRNN is determined by integrating the surrounding samples in the local spatial regions. Compared with CRNN, 3-DCNN uses 3-D convolution operator to extract joint spatial–spectral features simultaneously. Compared with CRNN and 3-DCNN, MSLN-CNN obtains better classification results due to effective sample augmentation and multilayer spatial–spectral feature fusion.

Compared with SSRN, MSLN-CNN increases by 4.0%, 0.5%, and 0.3% in terms of OA index on three HSI datasets. SSRN combines spatial–spectral residual learning and 3-DCNN for HSI classification. Compared SSRN, MNLS-CNN extracts complementary information by fusing joint spatial–spectral features from shallow to deep layers. MugNet adopts two parallel branches: spectral MugNet and spatial MugNet. Each branch uses a semisupervised principal component analysis network (S^2PCANet) based on multigrained scanning. In MugNet, unlabeled samples are selected to train S^2PCANet randomly. Compared with MugNet, MSLN-CNN uses local spatial and nonlocal spectral constraints to prelabel and select the unlabeled samples. Only unlabeled samples with high confidences are used for sample augmentation. MSLN-CNN increases by 7.8%, 4.4%, and 3.5% in terms of OA index on three HSI datasets.

H. Effectiveness Analysis to Data Augmentation in MSLN-CNN

We have added the experiment to verify the effectiveness of data augmentation in Table XIII. The comparison methods are RBF-SVM, JSSSAE, DBN, CNN, and 3-DCNN with sample augmentation, which are abbreviated as RBF-SVM-DA, JSSSAE-DA, DBN-DA, CNN-DA, and 3-DCNN-DA, respectively.

In Table XIII, the performance of the comparison methods has certain degrees of improvement by adding the proposed sample augmentation. For deep learning methods, the improvement is more obvious because of numerous parameters involved in deep neural networks. Compared with other comparison methods with sample augmentation, MSLN-CNN still achieves better classification results.

I. Effectiveness Analysis to Each Step in MSLN-CNN

We have added the experiment to verify the effectiveness of each step separately in Table XIV. The comparison methods are the proposed method without data augmentation (PM-WDA), the proposed method without integration of deep and shallow features (PM-WIDSF), the proposed method without fusion of 1-D and 2-D CNNs (PM-WFCNN), and the proposed method with only spatial CNN with a kernel size of 3×3 (PM-WSCNN3).The experimental results on the Indian Pines, Pavia University, and Salinas datasets are recorded.

In Table XIV, MSLN-CNN increases by about 1% than PM-WDA in terms of OA. It can be shown that adding effective data augmentation improves the classification performance. Compared with PM-WIDSF, MSLN-CNN improves 0.7%, 0.6%, and 0.8% on three HSI datasets, respectively. It is shown that complementary information from different layers is beneficial for classification. Compared with PM-WFCNN, MSLN-CNN improves 1.2%, 0.8%, and 0.9% on three HSI datasets, respectively. It is shown that joint spatial–spectral information is more effective than single information for HSI classification. Compared with MSLN-CNN, the OA of PM-WSCNN3 is dropped by about 0.4%. This is because MSLN-CNN uses the multiscale feature fusion.

J. Analysis of Free Parameters in MSLN-CNN

There are two important parameters k and r in MSLN-CNN. k is the number of nearest neighbors in the nonlocal spectral constraint and local spatial constraint. k controls the number of prelabeled unlabeled samples. r is the number of PCs in PCA. Fig. 8 shows the OA results of MSLN-CNN under different values of k. Fig. 9 shows the classification results under different numbers k and r.

In Fig. 8, r is fixed as 5. When the parameter k is in the range of [5, 7], MSLN-CNN obtains better classification results on three HSI datasets. When the value of k is too large or too small, the classification accuracy is degraded. When the value of k

TABLE XII
CLASSIFICATION RESULTS OF 3-DCNN-FDS, CRNN, SSRN, MUGNET, AND MSLN-CNN ON THE INDIAN PINES, PAVIA UNIVERSITY, AND SALINAS DATASETS

Dataset	Classification Index	3DCNN -FDS	CRNN	SSRN	MugNet	MSLN -CNN
Indian Pines Dataset	OA (%)	94.5±0.3	90.4±0.5	94.2±0.9	93.7±1.6	98.2±0.5
	AA (%)	93.3±2.1	84.5±1.2	84.1±1.3	89.1±1.9	94.4±3.0
	Kappa (%)	93.7±0.3	89.1±0.6	93.4±1.1	92.2±1.7	97.9±0.5
Pavia University Dataset	OA (%)	97.3±0.3	94.7±0.3	98.6±0.3	95.2±1.0	99.1±0.2
	AA (%)	94.7±0.4	94.6±0.6	97.2±0.4	93.8±1.8	98.1±0.1
	Kappa (%)	96.5±0.2	93.1±0.4	98.0±0.1	94.4±1.7	98.9±0.2
Salinas Dataset	OA (%)	98.6±0.2	95.2±0.3	98.4±0.3	94.0±1.1	98.7±0.2
	AA (%)	98.5±0.1	94.9±0.7	98.3±0.5	93.7±1.0	98.7±0.3
	Kappa (%)	98.4±0.1	94.6±0.2	97.9±0.3	93.8±1.1	98.6±0.3

TABLE XIII
OA RESULTS OF RBF-SVM-DA, JSSSAE-DA, DBN-DA, CNN-DA, 3-DCNN-DA, AND MSLN-CNN ON THE INDIAN PINES, PAVIA UNIVERSITY, AND SALINAS DATASETS

Methods	INDIAN PINES	PAVIA UNIVERSITY	SALINAS
RBF-SVM-DA	78.4±0.6	91.7±0.3	90.0±0.2
JSSSAE -DA	83.0±0.1	93.6±0.2	93.0±0.4
DBN-DA	81.5±0.2	93.5±0.3	91.8±0.5
CNN-DA	86.6±0.9	95.3±0.3	94.2±1.3
3DCNN-DA	94.0±0.6	97.8±0.6	98.4±0.3
MSLN-CNN	98.2±0.5	99.1±0.2	98.7±0.2

TABLE XIV
OA RESULTS OF PM-WDA, PM-WIDSF, PM-WFCNN, PM-WSCNN3, AND MSLN-CNN ON THE INDIAN PINES, PAVIA UNIVERSITY, AND SALINAS DATASETS

Methods	INDIAN PINES	PAVIA UNIVERSITY	SALINAS
PM-WDA	97.4±0.4	98.1±0.3	97.8±0.3
PM-WIDSF	97.5±0.5	98.5±0.3	97.9±0.2
PM-WFCNN	97.0±0.6	98.3±0.2	96.8±0.3
PM-WSCNN3	97.8±0.3	98.7±0.2	98.4±0.3
MSLN-CNN	98.2±0.5	99.1±0.2	98.7±0.2

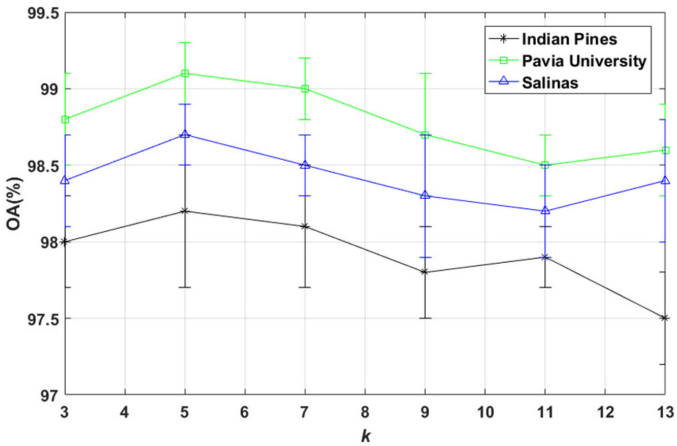


Fig. 8. Analysis of parameter k in MSLN-CNN.

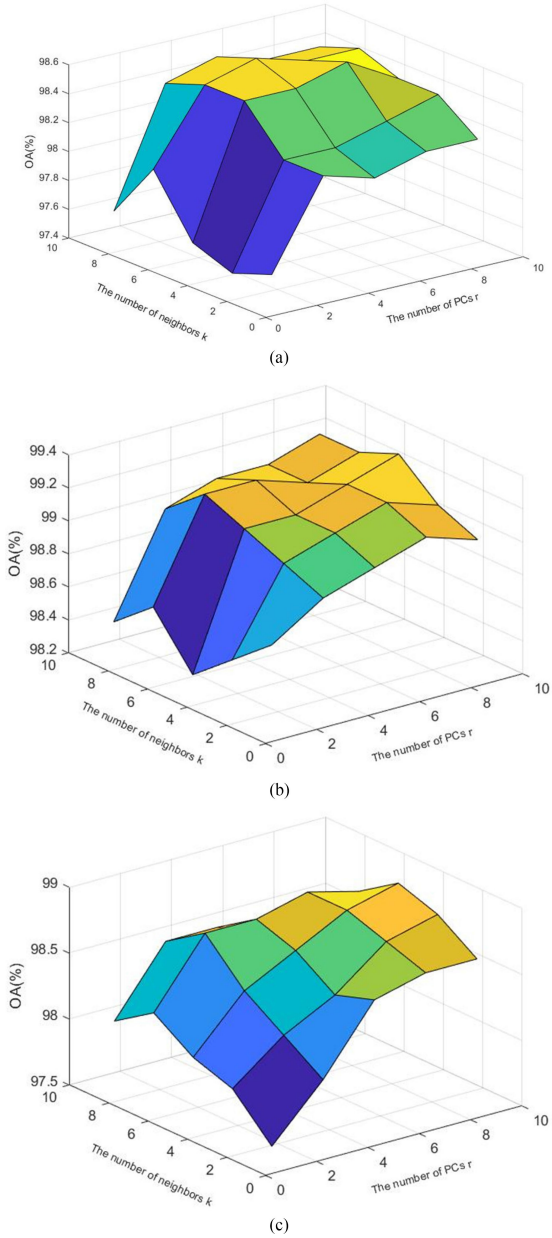


Fig. 9. Sensitivity analysis of parameters r and k in (a) the Indian Pines, (b) the Pavia University, and (c) the Salinas datasets.

is too large, the constraints of sample augment become strict. Fewer unlabeled samples are selected to prelabel. In this case, the proposed method has limited ability to alleviate overfitting. When the value of k is too small, a large number of unlabeled samples are selected to prelabel due to the loose constraints. In this case, k nearest samples is too few to extract enough contextual and structural information for data augmentation.

In Fig. 9, when k is fixed, OA value of MSLN-CNN is obviously improved with r in the range of [1, 5]. When r exceeds 5, the trend of improvement is not obvious. When the value of r reaches large enough, more information can be reserved. Finally, $r = 5$ is selected.

IV. Conclusion

This paper designs a novel MSLN-CNN method for HSI classification. Compared with existing spatial–spectral CNN methods, a triple-architecture CNN is constructed in MSLN-CNN. It effectively utilizes complementary spatial–spectral information by fusing the shallow features with detailed information and the deep features with semantic information. MSLN-CNN can achieve an end-to-end classification by jointly optimizing multilayer spatial–spectral feature fusion and classification. Furthermore, MSLN-CNN is a promising method to deal with the overfitting problem by considering both local spatial constraint and nonlocal spectral constraint. Experimental results demonstrated the effectiveness of MSLN-CNN for HSI classification.

In HSIs, the phenomenon of imbalance samples may appear, when some classes have much fewer samples than other classes. In the future, the sample imbalance problem will be considered in MSLN-CNN to improve the classification performance of the classes with fewer samples. Additionally, we will focus on the fusion of other deep architectures to further improve the classification performance of HSIs.

References

[1] C. I. Chang, *Hyperspectral Data Exploitation: Theory and Applications*. Hoboken, NJ, USA: Wiley, 2007.

[2] I. Makki, R. Younes, C. Francis, T. Bianchi, and M. Zucchetti, "A survey of landmine detection using hyperspectral imaging," *ISPRS J. Photogramm. Remote Sens.*, vol. 124, pp. 40–53, Feb. 2017.

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

[4] A. J. Brown, M. R. Walter, and T. J. Cudahy, "Hyperspectral mapping of ancient hydrothermal systems and applications for mars," in *Proc. AGU Fall Meeting Abstr.*, 2005, pp. 1–5.

[5] C. M. Gevaert, J. Suomalainen, J. Tang, and L. Kooistra, "Generation of spectral–temporal response surfaces by combining multispectral satellite and hyperspectral UAV imagery for precision agriculture applications," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 3140–3146, Jun. 2015.

[6] F. V. D. Meer, "Analysis of spectral absorption features in hyperspectral imagery," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 5, no. 1, pp. 55–68, 2004.

[7] X. Jia, B.-C. Kuo, and M. Crawford, "Feature mining for hyperspectral image classification," *Proc. IEEE*, vol. 101, no. 3, pp. 676–697, Mar. 2013.

[8] X. D. Kang, X. L. Xiang, S. T. Li, and J. A. Benediktsson, "PCA-based edge-preserving features for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 7140–7151, Dec. 2017.

[9] A. Villa, J. A. Benediktsson, J. Chanussot, and C. Jutten, "Hyperspectral image classification with independent component discriminant analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 12, pp. 4865–4876, Dec. 2011.

[10] T. V. Bandos, L. Bruzzone, and G. Camps-Valls, "Classification of hyperspectral images with regularized linear discriminant analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 3, pp. 862–873, Mar. 2009.

[11] 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.

[12] A. Mohan, G. Sapiro, and E. Bosch, "Spatially coherent nonlinear dimensionality reduction and segmentation of hyperspectral images," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 2, pp. 206–210, Apr. 2007.

[13] J. B. Tenenbaum, V. D. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[14] Y. Fang *et al.*, "Dimensionality reduction of hyperspectral images based on robust spatial information using locally linear embedding," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 10, pp. 1712–1716, Oct. 2014.

[15] M. Fauvel, J. Chanussot, and J. A. Benediktsson, "Kernel principal component analysis for the classification of hyperspectral remote-sensing data over urban areas," *EURASIP J. Adv. Signal Process.* vol. 2009, Feb. 2009, Art. no. 783194.

[16] W. Li, S. Prasad, J. E. Fowler, and L. M. Bruce, "Locality-preserving discriminant analysis in kernel-induced feature spaces for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 5, pp. 894–898, Sep. 2011.

[17] L. Shen and S. Jia, "Three-dimensional Gabor wavelets for pixel-based hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 12, pp. 5039–5046, Dec. 2011.

[18] J. Zhu, J. Hu, S. Jia, X. Jia, and Q. Li, "Multiple 3-D feature fusion framework for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 1873–1886, Apr. 2018.

[19] C. Cariou and K. Chehdi, "A new k-nearest neighbor density-based clustering method and its application to hyperspectral images," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2016, pp. 6161–6164.

[20] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 11, pp. 4085–4098, Nov. 2010.

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

[22] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification using dictionary-based sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 10, pp. 3973–3985, Oct. 2011.

[23] 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.

[24] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.

[25] 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.

[26] C. Tao, H. Pan, Y. Li, and Z. Zou, "Unsupervised spectral–spatial feature learning with stacked sparse autoencoder for hyperspectral imagery classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 12, pp. 2438–2442, Dec. 2015.

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

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

[29] H. Lee and H. Kwon, "Going deeper with contextual CNN for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 26, no. 10, pp. 4843–4855, Oct. 2017.

[30] H. Zhang, Y. Li, Y. Zhang, and Q. Shen, "Spectral-spatial classification of hyperspectral imagery using a dual-channel convolutional neural network," *Remote Sens. Lett.*, vol. 8, no. 5, pp. 438–447, 2017.

[31] Y. Chen, H. L. Jiang, and C. Y. Li, "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.

[32] J. Yang, Y. Zhao, J. C. Chan, and C. Yi, "Hyperspectral image classification using two-channel deep convolutional neural network," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Beijing, China, 2016, pp. 5079–5082.
[33] B. Pan, Z. Shi, N. Zhang, and S. Xie, "Hyperspectral image classification based on nonlinear spectral–spatial network," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 12, pp. 1782–1786, Dec. 2016.
[34] W. Li, G. D. Wu, and F. Zhang, "Hyperspectral image classification using deep pixel-pair features," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 844–853, Feb. 2016.
[35] H. Wu and S. Prasad, "Convolutional recurrent neural networks for hyperspectral data classification," *Remote Sens.*, vol. 9, no. 3, 2017, Art. no. 298.
[36] Y. Xu, L. Zhang, B. Du, and F. Zhang, "Spectral-spatial unified networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 10, pp. 5893–5909, Oct. 2018.
[37] B. Liu, X. Yu, A. Yu, and G. Wan, "Deep convolutional recurrent neural network with transfer learning for hyperspectral image classification," *J. Appl. Remote Sens.*, vol. 12, no. 2, pp. 1–17, 2018.
[38] P. Shamsolmoali, Z. Masoumeh, and Y. Jie, "Convolutional neural network in network (CNNiN): Hyperspectral image classification and dimensionality reduction," *IET Image Process.*, vol. 9, no. 2, pp. 246–253, Feb. 2019.
[39] Q. Liu, F. Zhou, R. Hang, and X. Yuan, "Bidirectional-convolutional LSTM based spectral-spatial feature learning for hyperspectral image classification," *Remote Sens.*, vol. 9, no. 2, pp. 1330–1348, 2017.
[40] A. Buades, B. Coll, and J. M. Morel, "A non-local algorithm for image denoising," in *Proc. IEEE Comput. Soc. Conf.*, 2005, pp. 60–65.
[41] M. Lin, Q. Chen, and S. Yan, "Network in network," in *Hyperspectral Data Exploitation: Theory and Applications*, C. I. Chang, Ed. Hoboken, NJ, USA: Wiley, 2007.
[42] S. Li, Q. Hao, X. Kang, and J. A. Benediktsson, "Gaussian pyramid based multiscale feature fusion for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 9, pp. 3312–3324, Sep. 2018.
[43] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
[44] N. Srivastava *et al.*, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[45] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, 2015, pp. 447–456.
[46] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. [Online]. Available: https://www.tensorflow.org
[47] Y. Qian and M. Ye, "Hyperspectral imagery restoration using nonlocal spectral-spatial structured sparse representation with noise estimation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 2, pp. 499–515, Apr. 2013.
[48] 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.
[49] B. Pan, Z. Shi, and X. Xu, "MugNet: Deep learning for hyperspectral image classification using limited samples," *ISPRS J. Photogram. Remote Sens.*, vol. 145, pp. 108–119, 2018.

Jie Feng (M'15) received the B.S. degree in electronic and information engineering from Chang'an University, Xi'an, China, in 2008, and the Ph.D. degree in electronic science and technology from Xidian University, Xi'an, China, in 2014.

She is currently an Associate Professor with the Laboratory of Intelligent Perception and Image Understanding, Xidian University. Her current research interests include remote sensing image processing, deep learning, and machine learning.

Jiantong Chen received the B.S. degree in electronical information science and technology from the Qingdao University of Science and Technology, Qingdao, China, in 2017. He is currently working toward the M.S. degree of pattern recognition and intelligent system in the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education, School of Electronic Engineering, Xidian University, Xi'an, China.

His current research interests include machine learning, remote sensing image processing, and pattern recognition.

Liguo Liu received the B.S. degree in electronic and information engineering from the Xi'an University of Science and Technology, Xi'an, China, in 2014, and the M.S. degree in electronics and communication engineering from Xidian University, Xi'an, China, in 2018.

His current interests include machine learning, remote sensing image processing, and pattern recognition.

Xianghai Cao (M'13) received the B.E. and Ph.D. degrees in electronic science and technology from the School of Electronic Engineering, Xidian University, Xi'an, China, in 1999 and 2008, respectively.

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

Xiangrong Zhang (SM'14) received the B.S. and M.S. degrees in computer science and technology and the Ph.D. degree in pattern recognition and intelligent system from Xidian University, Xi'an, China, in 1999, 2003, and 2006, respectively.

She is currently a Professor with the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education, School of Electronic Engineering, Xidian University. Her current research interests include visual information analysis and understanding, pattern recognition, and machine learning.

Licheng Jiao (SM'89–F'18) received the B.S. degree in high voltage from Shanghai Jiaotong University, Shanghai, China, in 1982, and the M.S. and Ph.D. degrees in theoretical electrician from Xi'an Jiaotong University, Xi'an, China, in 1984 and 1990, respectively.

He has authored or coauthored more than 150 scientific papers. His research interests include image processing, natural computation, machine learning, and intelligent information processing. He has been in charge of about 40 important scientific research projects, and has published more than 20 monographs and 100 papers in international journals and conferences.

Tao Yu received the B.S. and M.S. degrees in precision instrument from Wuhan University, Wuhan, China, in 2004 and 2009, respectively, and the Ph.D. degree in optical engineering from the University of Chinese Academy of Sciences, Beijing, China, in 2016.

He is currently working with the Key Laboratory of Spectral Imaging Technology, Chinese Academy of Sciences, Beijing. His current interests include spectral detection and polarization spectrum imaging.