Hyperspectral Image Classification With Deep Feature Fusion Network

Weiwei Song, *Student Member, IEEE*, Shutao Li, *Senior Member, IEEE*, Leyuan Fang, *Senior Member, IEEE*, and Ting Lu, *Member, IEEE*

***Abstract*— Recently, deep learning has been introduced to classify hyperspectral images (HSIs) and achieved good performance. In general, deep models adopt a large number of hierarchical layers to extract features. However, excessively increasing network depth will result in some negative effects (e.g., overfitting, gradient vanishing, and accuracy degrading) for conventional convolutional neural networks. In addition, the previous networks used in HSI classification do not consider the strong complementary yet correlated information among different hierarchical layers. To address the above two issues, a deep feature fusion network (DFFN) is proposed for HSI classification. On the one hand, the residual learning is introduced to optimize several convolutional layers as the identity mapping, which can ease the training of deep network and benefit from increasing depth. As a result, we can build a very deep network to extract more discriminative features of HSIs. On the other hand, the proposed DFFN model fuses the outputs of different hierarchical layers, which can further improve the classification accuracy. Experimental results on three real HSIs demonstrate that the proposed method outperforms other competitive classifiers.**

***Index Terms*— Convolutional neural networks (CNNs), feature fusion, hyperspectal image classification, residual learning.**

I. Introduction

HYPERSPECTRAL images (HSIs) are usually composed of several hundreds of spectral bands spanning from the visible spectrum to infrared spectrum. In HSIs, each pixel can be represented by a high-dimensional vector, whose entries correspond to the spectral reflectance in a specific wavelength. With the rich spectral information, HSIs have been applied in many fields, such as military [1], agriculture [2], and environment monitoring [3].

In the last decades, the HSI classification, assigning pixels to one specific class based on their spectral characteristics, has become a very active research topic in the remote sensing. A large number of methods (e.g., neural networks [4], support vector machines (SVMs) [5], multinomial logistic regression [6], [7], and active learning [8]) have been developed to build pixelwise-based classifiers for analyzing HSIs. In general, although these classifiers make full use of spectral information of HSIs, the obtained classification maps are still noisy, since the spatial contexts are not considered. More recently, many spectral–spatial features-based classification frameworks are proposed, which incorporate the spatial–contextual information into the pixelwise classifiers [9]. For instance, Benediktsson *et al.* [10], [11] utilize multiple morphological operations to construct spectral–spatial features of HSIs. Multiple kernel learning (e.g., composite kernel [12] and morphological kernel [13], [14]) based on spectral–spatial information is designed to improve the SVM classifier. Sparse presentation, as a powerful signal processing tool, is also introduced to analyze and process HSIs [15]–[21]. The sparse method is based on the observation that hyperspectral pixels can usually be represented by a linear combination of a few common pixels from the same class. Then, the contextual and spectral information of HSIs within a neighboring region is incorporated into a sparse model. In [22]–[24], spatial consistency is explored by segmenting HSIs into multiple superpixels based on the similarity of either intensity or texture. In [25], edge-preserving filtering (EPF) is proposed as a postprocessing technique to optimize the probabilistic results of the SVM.

Recently, the deep learning (DL), which is a powerful feature extraction technique, has made great breakthrough in many fields (e.g., image classification [26], [27], object detection [28], [29], and natural language processing [30]). Motivated by these successful applications, a few attempts based on the DL have been applied to analyze remote sensing images [31]–[48]. Chen *et al.* [37] propose a deep architecture composed of multilayer stacked autoencoders (SAEs) to extract spectral–spatial features of the HSIs, and then the learned features are processed by logistic regression to achieve classification. In [38] and [39], some improved deep networks based on autoencoder are also proposed for HSI classification. Specifically, a spatial updated deep auto-encoder is proposed to consider sample similarity by adding a regularization term in the energy function [38]. Liu *et al.* [39] combine the stacked denoising autoencoders and superpixel-based spatial constraints to obtain an improvement in classification performance. In addition, a deep belief network (DBN), as another deep model, is also proposed for HSI classification [41], [42]. Even though the aforementioned deep models can effectively

Manuscript received May 10, 2017; revised August 28, 2017 and November 1, 2017; accepted January 4, 2018. This work was supported in part by the National Natural Science Fund of China for International Cooperation and Exchanges under Grant 61520106001 and in part by the Fund of Hunan Province for the Science and Technology Plan Project under Grant 2017RS3024. *(Corresponding author: Shutao Li.)*

The authors are with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China (e-mail: weiwei_song@hnu.edu.cn; shutao_li@hnu.edu.cn; fangleyuan@gmail.com; tingluhnu@gmail.com).

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/TGRS.2018.2794326

extract deep features to boost discrimination among different classes, the way of transforming the input into 1-D vector actually cannot make full use of spatial information of HSIs. Very recently, the convolutional neural network (CNN)-based HSI classification methods are developed to solve the above-mentioned problem. In [43], a CNN-based spatial–spectral feature extraction framework is proposed to directly process the small cubes of HSIs. In [46], the deep pixel-pair features are extracted via the CNN by combining the center pixel and its surrounding pixels, which can increase the number of training samples. In [48], the spatial and spectral features of HSIs are extracted by the CNN and the balanced local discriminative embedding algorithm, respectively. In general, the network depth is of crucial importance for many visual recognition tasks, especially for processing HSIs with very complex spatial–spectral characteristics. However, excessively increasing depth will bring some negative effects (e.g., overfitting, gradient vanishing, and accuracy degrading) for the conventional CNN. Due to this reason, the previous networks used in HSI classification only adopt several convolutional layers (e.g., three, ten, and three convolutional layers for [43], [46], and [48], respectively), and thus, the deeper discriminative features cannot be sufficiently extracted. In addition, considering that the network is composed of multiple hierarchical layers, the strong complementary yet correlated information among different hierarchical layers is not exploited in previous works.

In this paper, we propose a novel deep feature fusion network (DFFN) to classify HSIs. Different from the previous networks used in HSI classification, we introduce the residual learning [49] to optimize several convolutional layers as the identity mapping, which can ease the training of a deep network and benefit from increasing depth. With the help of residual learning, we build a very deep network to extract more discriminative features of HSIs for classification. Furthermore, considering that the different layers can extract features of different scales that can provide complementary yet correlated information for classification, we further adopt a fusing mechanism to make full use of the multiple-layer features.

The remainder of this paper is organized as follows. Section II reviews the CNN and the related HSI classification method. Section III introduces the proposed network for the HSI classification. The experiments conducted on three real HSIs are shown in Section IV. Section V concludes this paper and suggests some future works.

II. Related Works

A. CNN

Instead of manually designing features, the deep networks can automatically learn high-level features with the way of layerwise representation. Compared with the other deep models (e.g., the DBN [50] and the SAE [51]), the CNN can directly process 2-D inputs, which can reserve spatial structure of images. A CNN mainly consists of a stack of alternating convolution layers and pooling layers with a number of fully connected layers. The convolutional layers first extract features in a way of filtering. Then, the pooling layers reduce the size of feature maps created by the convolutional layers to generate more general and abstract features. Finally, several fully connected layers are used to generate the final deep features. Each component is described as follows.

1) Convolutional Layers: Convolutional layers are the most important parts of the CNN. By stacking multiple convolutional layers, the CNN can extract high-level and robust features. At each convolutional layer, all feature maps of the previous layer are convolved with filters, creating multiple output maps. Let $\mathbf{X}$ be the input of a convolutional layer and its size is $M \times N \times C$, where $M \times N$ refers to spatial size of $\mathbf{X}$, C is the number of channels, and $\mathbf{x}_i$ is the ith feature map of $\mathbf{X}$. Supposing that the convolutional layer has k filters denoted as $\mathbf{W}$ and the bias parameter is $\mathbf{b}$, the jth output of convolutional layer can be represented as follows:

$$\mathbf{z}_j = \sum_{i=1}^{C} \sigma(\mathbf{x}_i * \mathbf{w}_j + b_j), \quad j = 1, 2, \ldots, k \tag{1}$$

where $\mathbf{w}_j$ and b_j are the jth component of $\mathbf{W}$ and $\mathbf{b}$, respectively, the operator $*$ represents discrete convolution operation, and σ refers to activation function, which is utilized to improve the nonlinearity of network. Recently, $ReLU$ [26] is the mostly used activation function due to the fast convergence, which is denoted as

$$\sigma(\mathbf{x}) = \max(0, \mathbf{x}). \tag{2}$$

2) Pooling Layers: As the abundant redundancy information exists in images, it is common to periodically insert a pooling layer after several convolutional layers in the CNN. The pooling operation progressively reduces the spatial size of the feature maps, which actually reduces the amount of parameters and computation of the network. By applying convolutional and pooling layers, the size of feature maps becomes smaller and the representation of extracted features becomes more abstract. Specifically, for a $P \times P$ window denoted as $\mathbf{S}$, the averaging pooling operation can be denoted as

$$z = \frac{1}{T} \sum_{(i,j) \in \mathbf{S}} x_{ij} \tag{3}$$

where T is the number of elements of $\mathbf{S}$ and x_{ij} is the activation value corresponding to the position (i, j).

3) Fully Connected Layers: Fully connected layers are used to combine all features of the previous layer by reshaping them into a n-dimension vector (e.g., $n = 4096$ in AlexNet [26]). Particularly, the number of neurons in the last fully connected layer is equal to number of classes for classification tasks. Finally, the feature vector is input to the $softmax$, an extension of logistic regression, to generate the probability distribution of outputs.

B. HSI Classification Based on CNN

With the strong ability of extracting features, the CNN has been recently introduced to classify HSIs [43], [46], [48]. These networks are expected to extract either the spatial–spectral features from the original HSIs [43] or the spatial-related features from the first several principal components of HSIs [48]. Specifically, let $\mathbf{X}$ be an HSI with the size of

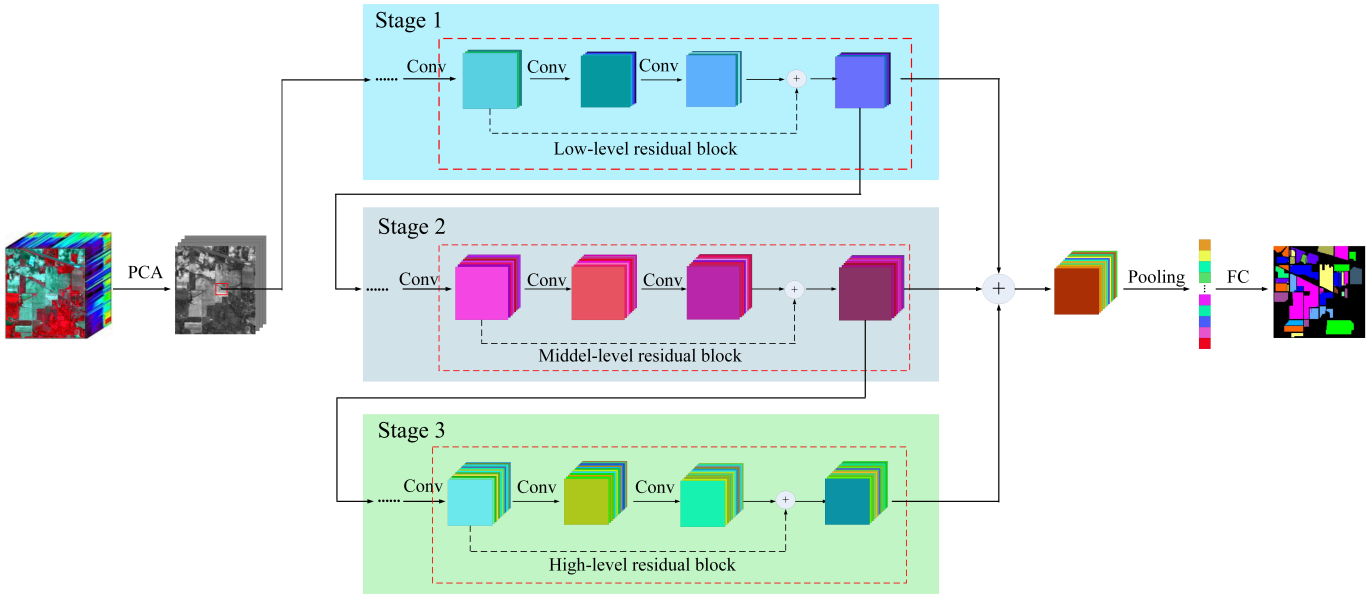


Fig. 1. Overall flowchart of HSI classification based on the DFFN. For convenience, the batch normalization, activation layers followed by convolutional layers, and dimension-matching operation are not given.

$M \times N \times C$, and $\mathbf{x}^k \in \mathbb{R}^{C\times 1}$, $k = 1, \ldots, K$ be a hyperspectral pixel, where C and K are the number of spectral bands and training pixels, respectively. If there are T possible classes in this HSI, the truth label of $\mathbf{x}^k$ can be encoded as $\mathbf{t}^k$, which is actually a vector of length T with value "1" at the position of the correct label and "0" elsewhere. Different from the natural image classification which inputs the whole images, the HSI classification based on the CNN uses image patches centered on labeled pixels as the input samples. Let $\mathbf{Y}^k \in R^{W\times W\times C}$ be the image patch centered on $\mathbf{x}^k$ with the window size of $W \times W$. Given input samples $\{\mathbf{Y}^k, \mathbf{t}^k\}_{k=1}^K$, the output of network can be computed by a series of convolutional, pooling, and fully connected operations

$$\mathbf{z}^k = f(\mathbf{W}\mathbf{Y}^k + \mathbf{b}) \tag{4}$$

where f is a composite function of input $\mathbf{Y}^k$, which is obtained by multiple linear and nonlinear operations, and $\mathbf{W}$ and $\mathbf{b}$ are the weight and bias parameters of network, respectively. As mentioned in Section II-A, the $softmax$ function is used to generate the probability distribution of output, which is denoted as

$$\mathbf{p}_i^k = \frac{e^{\mathbf{z}_i^k}}{\sum_{j=1}^T e^{\mathbf{z}_j^k}}, \quad i = 1, \ldots, T \tag{5}$$

where $\mathbf{z}_i^k$ is the ith value of $\mathbf{z}^k$. Based on the predicted and truth values, the loss function $\mathcal{L}$ can be defined as

$$\mathcal{L} = -\frac{1}{K}\sum_{k=1}^{K}\sum_{i=1}^{T}\mathbf{t}_i^k \log \mathbf{p}_i^k. \tag{6}$$

To minimize the loss function, the stochastic gradient descent (SGD) algorithm is used to solve the network parameters $\mathbf{W}$ and $\mathbf{b}$. Once the optimization is completed, the trained CNN can be used to predict the label of test hyperspectral pixel $\mathbf{x}' \in \mathbf{X}$ based on the maximum probability p_i

$$Class(\mathbf{x}') = \arg\max_{i=1,2,\ldots,T} p_i. \tag{7}$$

III. Proposed DFFN Model for HSI Classification

Due to the existence of spectral mixing and spatial variability of spectral signatures, HSIs usually have very complex spatial–spectral characteristics. In general, only several convolutional layers cannot sufficiently extract more discriminative features of HSIs for accurate classification. However, excessively increasing network depth will bring some problems (e.g., overfitting, gradient vanishing, and accuracy degrading) for the conventional CNN. Besides, considering that the conventional CNN is composed of multiple hierarchical layers, the strong complementary yet correlated information among different hierarchical layers is not exploited in previous works. In this paper, we propose a novel DFFN model to solve the abovementioned two issues. On the one hand, by introducing the residual learning [49] to optimize several convolutional layers as the identity mapping, we build a very deep network to extract more discriminative features of HSIs without suffering from degradation of performance. On the other hand, we further adopt a fusing mechanism to make full use of the multiple-layer features of network.

In the following, we will describe the whole framework of HSI classification based on the proposed DFFN. First, the principal component analysis (PCA) algorithm is performed on hyperspectral data to extract the most informative components, which can reduce the cost of computation. Then, training image patches centered on labeled pixels are built to train the DFFN. Finally, the labels of test pixels can be predicted by the trained network. Fig. 1 gives the overall flowchart of HSI classification based on the DFFN. The whole architecture can be divided into three stages in terms of the

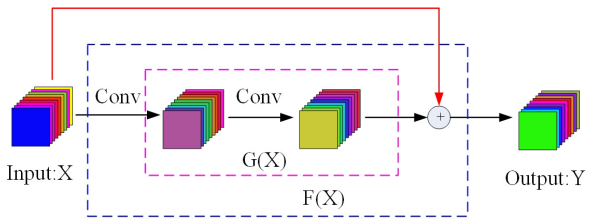


Fig. 2. Illustration of a residual block.

number of convolutional filters. Specifically, there are 16, 32, and 64 convolutional filters in stage 1, stage 2, and stage 3, respectively. Overall, the main procedures of classification are detailed in the following three steps.

A. Constructing a Very Deep Network

In the first place, we construct a very deep network to extract more discriminative features of HSIs. Specifically, the proposed network consists of multiple residual blocks and each block contains two convolutional layers. Following [49], batch normalization [52] is adopted to accelerate convergence of network after each convolution and before activation. Fig. 2 gives the illustration of a residual block. Let $F(\mathbf{X})$ be the original underlying function to be learned by two convolutional layers, where $\mathbf{X}$ denotes the input of the first layer of residual block. The residual learning actually attempts to optimize the two convolutional layers as an identity mapping, which is achieved by using the short connection, as seen in Fig. 2 (red line). By introducing residual function $G(\mathbf{X}) = F(\mathbf{X}) - \mathbf{X}$, the object function from the original $F(\mathbf{X}) = \mathbf{X}$ equivalently converts to $G(\mathbf{X}) = 0$. As described earlier, $F(\mathbf{X})$ can be written by the following form:

$$F(\mathbf{X}) = G(\mathbf{X}) + \mathbf{X}. \tag{8}$$

As shown in Fig. 2, $G(\mathbf{X})$ can be computed by twice convolution with $\mathbf{X}$

$$G(\mathbf{X}) = \sigma(\sigma(\mathbf{X} * \mathbf{W}_1 + \mathbf{b}_1) * \mathbf{W}_2 + \mathbf{b}_2) \tag{9}$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ are convolutional kernels, $\mathbf{b}_1$ and $\mathbf{b}_2$ are trainable bias parameters, and σ refers to the $ReLU$ function. By stacking multiple residual blocks, the extracted features become more and more discriminative.

B. Fusing Multiple-Layer Features

In recent years, the idea of feature fusion has been applied to many visual tasks. Specifically, Long *et al.* [53] combine the deep semantic features with shallow appearance features to produce accurate and detailed segmentations. Chaib *et al.* [54] effectively fuse the deep features extracted from the first and second fully connected layers for the very high-resolution remote sensing scene classification. In addition, a DL strategy is presented to fuse multiple semantic cues for complex event recognition [55]. In this paper, we introduce the feature fusion mechanism to exploit the strong complementary and correlated information among different hierarchical layers for HSI classification.

Considering that different layers may have the different numbers of feature maps, we use the dimension-matching function (i.e., linear projection) to ensure they have the same spectral dimensionality before feature fusion. Specifically, assume that $\mathbf{F}_L$, $\mathbf{F}_M$, and $\mathbf{F}_H$ refer to the outputs of low-level, middle-level, and high-level layers, respectively, and they have 16, 32, and 64 feature maps, respectively. Then, we can use 64 kernels with the size of 1×1 to convolute them. With such convolution operation, the numbers of feature maps of $\mathbf{F}_L$, $\mathbf{F}_M$, and $\mathbf{F}_H$ all become 64. Finally, the feature fusion can be easily performed with a way of elementwise summation. The whole process can be represented by the following equation:

$$\mathbf{z} = pooling(g_1(\mathbf{F}_L) + g_2(\mathbf{F}_M) + g_3(\mathbf{F}_H)) \tag{10}$$

where $\mathbf{z}$ represents the fused features, g_1, g_2, and g_3 are the dimension-matching function as mentioned above, and $pooling$ is the global averaging function.

C. Classifying HSIs Based on DFFN

After processing by several fully connected layers, the fused features are transformed into an output feature vector. Then, the feature vector is input to a $softmax$ layer to calculate the conditional probabilities of each class. For the feature vector $\mathbf{z}$, the probability distribution can be denoted as

$$p_i = \frac{e^{\mathbf{z_i}}}{\sum_{j=1}^{T} e^{\mathbf{z_j}}}, \quad i = 1, \ldots, T. \tag{11}$$

Since the operations of residual learning and feature fusing are all differentiable, the optimization of DFFN is the same as the typical CNN [56]. Specifically, the SGD is utilized to learn the network parameters. Once the deep network is well trained, the label of an arbitrary test hyperspectral pixel $\mathbf{x}'$ can be determined based on the maximum probability

$$Class(\mathbf{x}') = \arg\max_{i=1,2,\ldots,T} p_i. \tag{12}$$

IV. Experimental Results and Analysis

A. Data Sets

In this section, we perform several experiments to verify the effectiveness of the proposed network on three real HSI data sets, including the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Indian Pines image, Reflective Optics System Imaging Spectrometer (ROSIS-03) University of Pavia image, and AVIRIS Salinas image.

The Indian Pines image was captured by the AVIRIS sensor over the agricultural Indian Pine test site in northwestern Indiana in 1992. This scene has 220 data channels across the spectral range from 0.2 to 2.4 μm, and each band is of size 145×145. The image has the spatial resolution of 20 m/ pixel and it contains 16 ground-truth classes, most of which are different types of crops. Fig. 3(a) and (b) shows the false color composite of Indian Pines image and the corresponding ground truth data, respectively. Before the experiments, 20 water absorption bands were removed.

The University of Pavia image, which captures an urban area surrounding the University of Pavia, Pavia, Italy, was recorded by the ROSIS-03 sensor. The image is of size $610 \times 340 \times 115$ with a spatial resolution of 1.3 m/pixel and spectral coverage

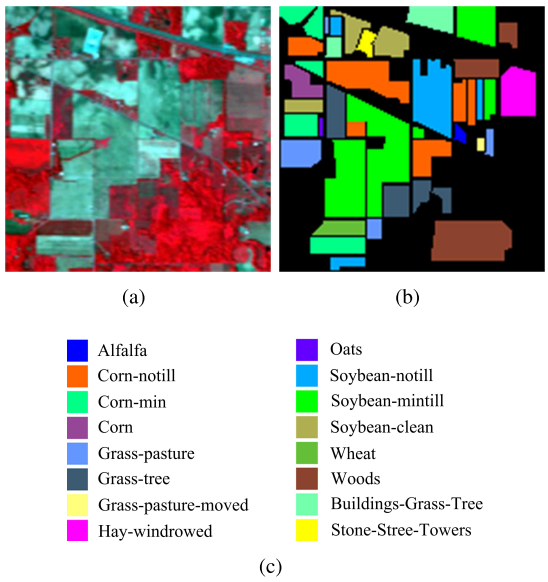


Fig. 3. Indian Pines image. (a) Three-band false color composite. (b) Ground truth data. (c) Color code.

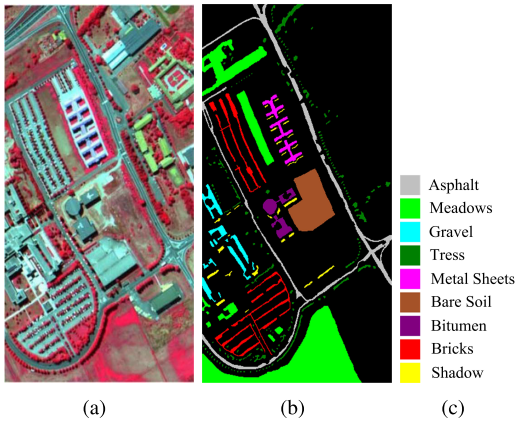


Fig. 4. University of Pavia image. (a) Three-band false color composite. (b) Ground truth data. (c) Color code.

ranging from 0.43 to 0.86 μm. Nine classes of interest are considered for this image. Before the experiments, 12 very noisy bands were removed. Fig. 4(a) and (b) shows the color composite of the University of Pavia image and the corresponding reference data.

The Salinas image was also captured by the AVIRIS sensor over the area of Salinas Valley, CA, USA, and with a spatial resolution of 3.7 m/pixel. The image has 224 spectral bands of size 512×217. As with the Indian Pines image, 20 water absorption bands were discorded. This image has 16 ground-truth classes. Fig. 5(a) and (b) shows the color composite of the Salinas image and the corresponding reference data.

B. Compared Methods

The performance of the proposed DFFN is compared with other methods. These compared approaches are divided into two kinds of groups: one is based on the conventional machine learning methods, including the SVM [5], extended morphological profiles (EMPs) [11], EPF [25], and joint sparse

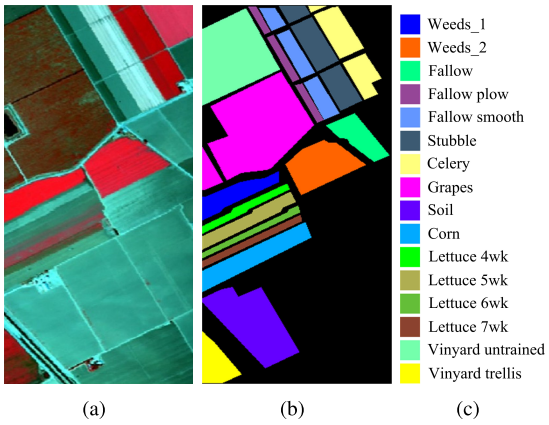


Fig. 5. Salinas image. (a) Three-band false color composite. (b) Ground truth data. (c) Color code.

representation (JSR) [15]; and the other one is based on DL, including the SAEs [37], rolling guidance filter and vertex component analysis network (RVCANet) [47], a deep CNN (DCNN, a plain network without using residual learning and feature fusion), and a deep residual network (DRN, an improved network without using feature fusion) [49]. The SVM that only utilizes spectral pixels is implemented by using support vector machines library [57] with a Gaussian kernel. The other methods explore both spatial and spectral features by means of different ways. For the EPF and RVCANet, the edge-preserving filter is adopted to make full use of spatial structure feature. For the EMP method, the spatial information of HSIs is exploited by the adoption of the morphological profiles. As to the JSR, SAE, DCNN, DRN, and DFFN methods, the spatial information within a fixed-size local region is utilized by the corresponding technology.

It is worth mentioning that the above three networks, i.e., the DCNN, DRN, and DFFN, have the similar architecture. The DCNN is a plain network, which consists of multiple convolutional layers, one pooling layer, and one fully connected layer. The DRN and the DFFN have introduced residual learning to optimize the DCNN. In addition, the DFFN has also utilized feature fusion mechanism to consider multiple-layer features compared with the DRN.

In our experiments, we empirically choose the network parameters for the DCNN, DRN, and DFFN. Table I shows the network architecture on the Indian Pines, University of Pavia, and Salinas images, respectively. N, S, D, and N_f refer to the number of extracted principal components, the size of image patches, network depth, and the number of filters, respectively. For the type of layers, C, P, and FC represent the convolutional, pooling, and fully connected operations, respectively. The size of all filters is 3×3 and the pooling function is the globally averaging operation. In addition, the training parameters are also set and keep unchanged on the three test images. Specifically, the initial learning rate is 0.1 and then divided by 10 when the error plateaus. These networks are trained for 2×10^4 iterations and the training minibatch has a size of 100. We use a weight decay of 0.0001 and a momentum of 0.9. In order to recover the results of other

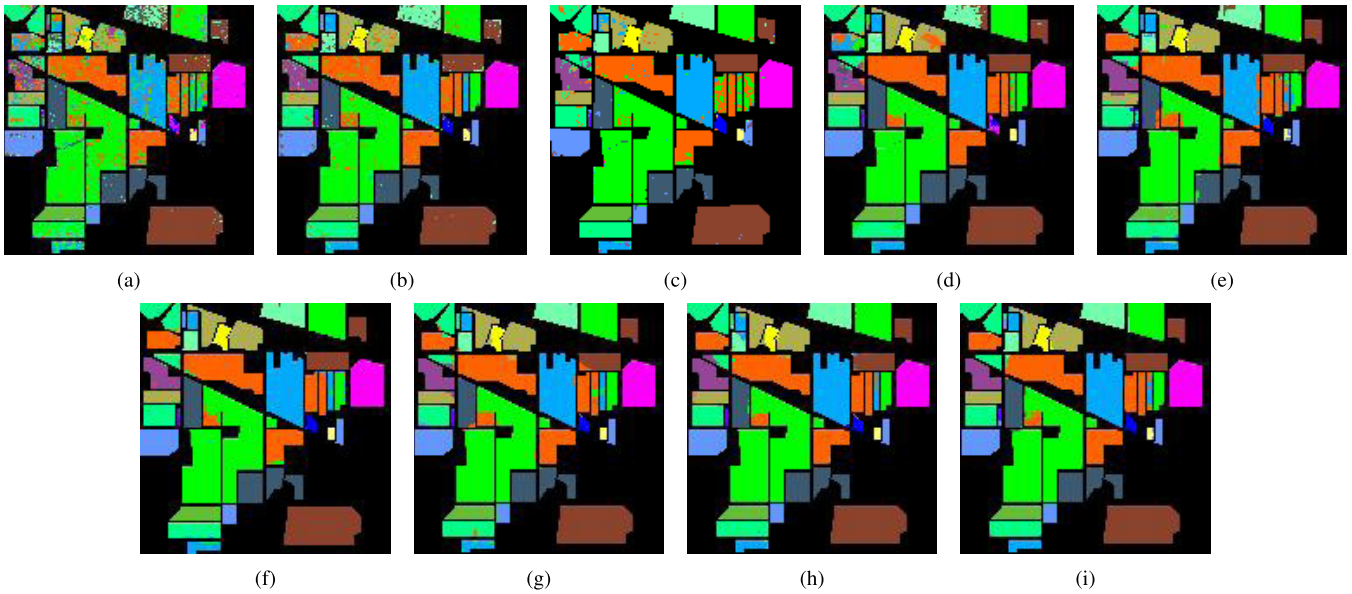


Fig. 6. Classification maps on the Indian Pines data set obtained by (a) SVM [5], (b) SAE [51], (c) EMP [11], (d) EPF [25], (e) JSR [15], (f) RVCANet [47], (g) DCNN, (h) DRN [49], and (i) DFFN.

TABLE I
ARCHITECTURE OF THE DCNN, DRN, AND DFFN ON THREE TEST IMAGES. N, S, D, AND N_f REFER TO THE NUMBER OF EXTRACTED PRINCIPAL COMPONENTS, THE SIZE OF IMAGE PATCHES, NETWORK DEPTH, AND THE NUMBER OF FILTERS, RESPECTIVELY. FOR THE TYPE OF LAYERS, C, P, AND FC REPRESENT THE CONVOLUTIONAL, POOLING, AND FULLY CONNECTED OPERATIONS, RESPECTIVELY

Images	N	S	D	Layers	Type	N_f
Indian Pines	3	25×25	27	1-9	C	16
				10-17	C	32
				18-25	C	64
				26	P	-
				27	FC	16
University of Pavia	5	23×23	33	1-11	C	16
				12-21	C	32
				22-31	C	64
				32	P	-
				33	FC	9
Salinas	10	27×27	27	1-9	C	16
				10-17	C	32
				18-25	C	64
				26	P	-
				27	FC	16

methods, the parameters for the SVM are obtained by fivefold cross validation, and the parameters for the EMP, JSR, EPF, RVCANet, and SAE are set to the default values in [11], [15], [25], [47], and [51], respectively.

C. Quantitative Metrics

In order to quantitatively evaluate the performance of different methods, three objective metrics, i.e., overall accuracy (OA), average accuracy (AA), and the Kappa coefficient, are adopted in these experiments. Specifically, the OA is computed by the ratio between the number of correctly classified test samples and the total number of test samples. The AA is the mean of all class accuracies. The Kappa coefficient is computed by weighting the measured accuracies, which represents the robust measure of the degree of agreement.

D. Classification Results

The first experiment is conducted on the Indian Pines image. All labeled pixels are divided into training and test subsets. Since the SAE requires a large amount of samples to train parameters of network, the training samples account for about 50% of the whole labeled reference data for the SAE. For the other methods, about 10% of samples are randomly selected to train classifiers, and the rest 90% are used as test samples, as seen in the second column in Table II. Fig. 6 shows classification maps obtained by different approaches. As can be observed, the SVM obtains poor classification performance and exhibits very noisy estimations in its classification map, since it does not consider the spatial information. In contrast, the other methods deliver smoother appearance in their results by combining spatial and spectral information. In general, the visual performances obtained by DL-based methods (e.g., the RVCANet, DCNN, DRN, and DFFN) are better than that of the compared conventional machine learning methods (e.g., the SVM, EMP, EPF, and JSR). Furthermore, compared with the RVCANet, DCNN, and DRN, the proposed DFFN can accurately classify pixels in the near-edge regions and provide more similar results to the reference map, as shown in Fig. 3(b). Apart from visual comparison, Table II gives the quantitative results of various methods on the Indian Pines image. To fairly compare the classification results, the SAE is excluded in quantitatively compare section. Specifically, these classification accuracy values are averaged over ten experiments with different randomly selected training data. As can be seen, DL-based methods achieve a better performance. In addition, by introducing residual learning, the DRN and the DFFN get much improvement compared with the DCNN. Moreover, by comparing the results of the DRN and the

TABLE II

NUMBER OF TRAINING AND TEST SAMPLES OF THE INDIAN PINES IMAGE AND CLASSIFICATION ACCURACIES (IN PERCENTAGES) OBTAINED BY THE SVM [5], EMP [11], EPF [25], JSR [15], RVCANET [47], DCNN, DRN [49], AND DFFN. THE STANDARD DEVIATION VALUES ARE ALSO GIVEN IN THE BRACKETS

Class	Training/Test	SVM	EMP	EPF	JSR	RVCANet	DCNN	DRN	DFFN
Alfalfa	5/41	27.80	95.61	23.41	92.68	**98.78**	90.24	97.56	97.56
Corn-no till	143/1285	75.91	84.65	90.63	95.03	95.56	97.66	97.66	**97.75**
Corn-min till	83/747	65.94	92.57	84.66	90.99	96.70	97.72	98.11	**98.31**
Corn	24/213	50.66	85.87	87.00	93.38	95.87	97.70	97.51	**98.12**
Grass/Pasture	49/434	87.60	91.45	95.23	92.83	96.85	97.63	96.96	**98.66**
Grass/Trees	73/657	95.92	96.13	**99.98**	94.99	99.92	99.16	98.57	99.62
Grass/Pasture-mowed	3/25	65.20	86.80	88.00	83.20	**98.40**	97.20	96.00	97.60
Hay-windrowed	48/430	97.86	99.56	**100**	99.51	99.98	99.88	99.70	**100**
Oats	2/18	27.78	66.67	17.78	31.67	**98.44**	73.33	98.33	93.89
Soybeans-no till	98/874	75.58	86.49	91.03	91.77	96.53	97.16	97.72	**98.20**
Soybeans-min till	245/2210	82.59	93.93	97.81	96.24	98.39	98.53	98.18	**98.76**
Soybean-clean	60/533	64.24	84.09	92.74	92.16	**97.38**	96.17	96.12	96.00
Wheat	21/184	96.96	98.97	99.46	90.54	99.46	98.53	97.88	**99.51**
Woods	126/1139	95.18	99.11	99.84	99.52	**99.89**	99.37	99.57	99.82
Building-Grass-Trees-Drives	39/347	53.26	97.09	77.32	90.89	**99.26**	97.06	98.30	98.65
Stone-steel Towers	10/83	76.51	97.59	86.75	86.63	**96.45**	87.23	92.53	90.60
OA	-	80.03 (0.79)	92.20 (0.52)	93.59 (0.88)	94.65 (0.44)	97.92 (0.37)	97.93 (0.47)	98.36 (0.42)	**98.52** (0.23)
Kappa	-	77.15 (0.90)	91.04 (1.67)	92.66 (1.01)	93.90 (0.50)	97.63 (0.42)	97.65 (0.53)	98.13 (0.48)	**98.32** (0.26)
AA	-	71.19 (1.38)	91.08 (0.59)	83.23 (1.29)	88.88 (1.17)	97.42 (0.86)	95.17 (1.60)	97.62 (0.79)	**97.69** (0.74)

TABLE III

NUMBER OF TRAINING AND TEST SAMPLES OF THE UNIVERSITY OF PAVIA IMAGE AND CLASSIFICATION ACCURACIES (IN PERCENTAGES) OBTAINED BY THE SVM [5], EMP [11], EPF [25], JSR [15], RVCANET [47], DCNN, DRN [49], AND DFFN. THE STANDARD DEVIATION VALUES ARE ALSO GIVEN IN THE BRACKETS

Class	Training/Test	SVM	EMP	EPF	JSR	RVCANet	DCNN	DRN	DFFN
Asphalt	132/6499	90.10	98.70	97.94	76.64	93.91	96.11	98.75	**98.82**
Meadows	372/18277	97.57	96.57	**99.85**	99.15	99.06	98.99	99.66	99.74
Gravel	42/2057	70.95	93.86	83.31	85.25	77.71	90.68	96.19	**98.14**
Tress	62/3002	90.34	**95.47**	94.36	85.22	89.47	92.83	92.37	94.43
Metal sheets	27/1318	98.91	99.29	**99.99**	97.55	99.79	98.97	99.17	99.14
Bare soil	101/4928	81.88	75.51	94.13	94.21	90.83	99.31	99.68	**99.69**
Bitumen	27/1303	82.19	96.36	**99.49**	96.33	85.80	93.90	97.44	96.68
Bricks	74/3608	87.58	96.97	97.71	94.64	90.19	97.03	98.49	**98.95**
Shadows	19/928	99.81	99.40	**99.92**	39.01	94.89	89.43	88.86	89.54
OA	-	91.50 (0.43)	94.39 (0.51)	97.49 (0.27)	91.54 (0.65)	94.28 (0.91)	97.19 (0.67)	98.52 (0.34)	**98.73** (0.31)
Kappa	-	88.65 (0.56)	94.68 (0.59)	96.65 (0.37)	88.76 (0.85)	92.40 (1.23)	96.28 (0.89)	98.03 (0.45)	**98.31** (0.41)
AA	-	88.81 (0.58)	92.47 (0.68)	96.30 (0.59)	85.33 (1.08)	91.29 (1.53)	95.25 (1.47)	96.78 (0.70)	**97.24** (0.53)

DFFN, the feature-fusion mechanism can further improve the performance, which demonstrates the effectiveness of feature fusion. Overall, our method performs better than all other cited methods in terms of three metrics (the OA, AA, and Kappa coefficient).

The second and third experiments are performed on the University of Pavia and Salinas images, respectively. For the University of Pavia image, 2% of labeled pixels are randomly selected as training samples (apart from the SAE for 50%), and the rest of samples are utilized for testing. For the Salinas image, we only use 0.5% samples per class to train classifiers (apart from the SAE for 50%), and the rest of samples are used as the test samples. Figs. 7 and 8 show the classification maps obtained by different methods. Tables III and IV present the training and test samples and quantitative comparison results of different methods. In the two examples, it can be seen that the proposed DFFN significantly improved the OA from 97.19% to 98.73% for the University of Pavia image, and 95.05% to 98.87% for the Salinas image compared with the DCNN.

E. Analysis of Parameters

For the proposed DFFN classification method, the PCA algorithm is first performed on the original HSIs with the purpose of extracting first several principal components. Then, in the dimensionality-reduced HSIs, the fixed-size image patches centered on labeled pixels are input to the proposed network. Apart from the number of principal components (denoted as N) and the size of image patches (denoted as S), the network depth (denoted as D) is also a crucial factor for classification performance. Therefore, the effects of these three parameters will be analyzed in this section.

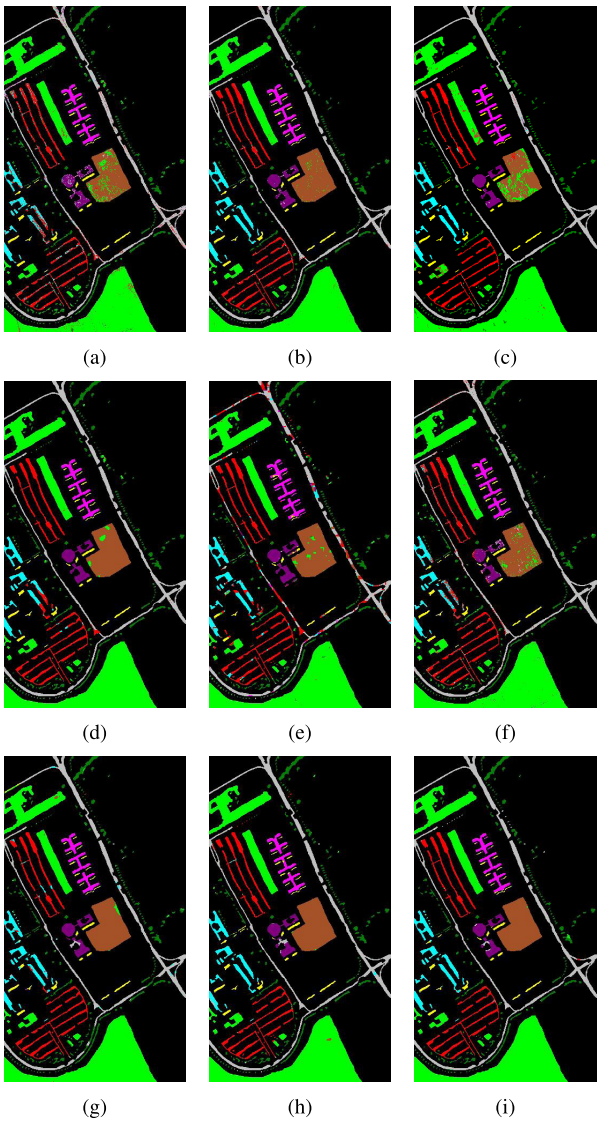


Fig. 7. Classification maps on the University of Pavia data set obtained by (a) SVM [5], (b) SAE [51], (c) EMP [11], (d) EPF [25], (e) JSR [15], (f) RVCANet [47], (g) DCNN, (h) DRN [49], and (i) DFFN.

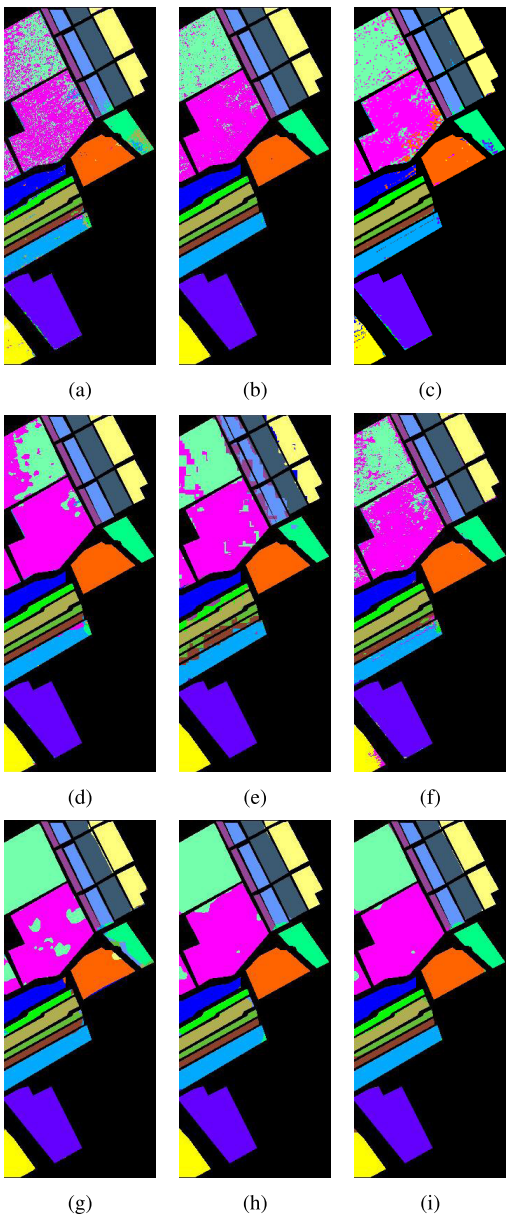


Fig. 8. Classification maps on the Salinas data set obtained by (a) SVM [5], (b) SAE [51], (c) EMP [11], (d) EPF [25], (e) JSR [15], (f) RVCANet [47], (g) DCNN, (h) DRN [49], and (i) DFFN.

1) Effect of N on Classification Accuracies: In this analysis, S is empirically set to 25×25, 23×23, and 27×27 for the Indian Pines, University of Pavia, and Salinas images, respectively. D is set to 28, 34, and 28 for the three test images, respectively. Fig. 9 shows the OA values with the different N values on three test images. It can be observed that the OA values first improve, and then become stable as N increases. Actually, most of the information in HSIs exists in the first several principal components. Therefore, the OA values tend to rise as n increases in the beginning. Then, utilizing a larger number of principal components does not further improve performance.

2) Effect of S on Classification Accuracies: In order to discuss the influence of the size of image patches, N is empirically set to 3, 5, and 10 for the Indian Pines, University of Pavia, and Salinas images, respectively. D is set to 28, 34, and 28 for the three test images, respectively. As can be seen from Fig. 10, the OA values on the Indian Pines and

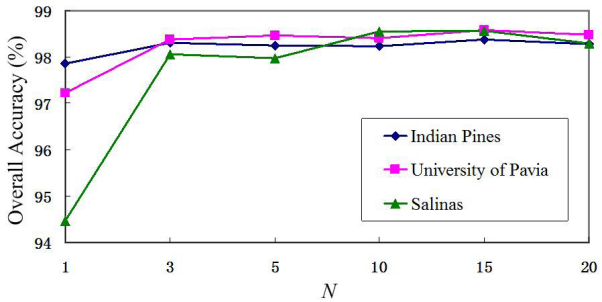


Fig. 9. Effect of N on classification accuracies in the three HSIs.

Salinas images generally increase or become comparatively stable as S becomes larger. By contrast, the influence of S on the University of Pavia image is relatively sensitive, and

TABLE IV

NUMBER OF TRAINING AND TEST SAMPLES OF THE SALINAS IMAGE AND CLASSIFICATION ACCURACIES (IN PERCENTAGES) OBTAINED BY THE SVM [5], EMP [11], EPF [25], JSR [15], RVCANET [47], DCNN, DRN [49], AND DFFN. THE STANDARD DEVIATION VALUES ARE ALSO GIVEN IN THE BRACKETS

Class	Training/Test	SVM	EMP	EPF	JSR	RVCANet	DCNN	DRN	DFFN
Weeds_1	11/1998	97.54	98.63	99.67	**100**	99.64	98.20	99.46	99.23
Weeds_2	19/3707	99.24	98.28	**100**	99.93	99.76	96.92	99.79	99.94
Fallow	10/1966	90.53	92.78	93.44	98.88	98.98	84.62	98.73	**99.95**
Fallow_plow	7/1387	98.25	98.30	**99.98**	56.89	97.04	96.28	91.98	97.49
Fallow_smooth	14/2664	95.21	95.00	97.73	83.52	**99.85**	94.76	98.47	96.70
Stubble	20/3939	98.82	96.48	**99.93**	95.67	99.59	99.07	99.26	99.15
Celery	18/3561	99.34	98.16	**99.90**	96.04	98.85	98.12	99.01	99.62
Grapes	57/11214	79.13	85.58	92.18	91.42	89.58	89.84	97.28	**98.16**
Soil	32/6171	98.75	98.35	99.93	99.80	99.38	97.07	99.88	**99.96**
Corn	17/3261	85.18	91.55	94.34	91.01	91.06	96.43	98.98	**99.43**
Lettuce 4wk	6/1062	85.30	93.95	98.50	81.95	**99.15**	93.87	97.14	97.16
Lettuce 5wk	10/1917	97.62	99.67	**100**	84.34	**100**	95.64	99.12	98.53
Lettuce 6wk	5/911	98.01	97.96	**100**	73.82	99.34	96.24	96.83	95.81
Lettuce 7wk	6/1064	89.49	94.58	96.39	81.06	90.91	95.10	98.04	**98.53**
Vinyard untrained	37/7231	56.81	84.30	63.68	85.21	79.04	97.03	98.94	**99.08**
Vinyard trellis	10/1797	90.14	90.06	93.25	98.25	90.45	98.11	97.78	**99.35**
OA	-	86.97 (1.30)	92.48 (0.95)	92.44 (2.84)	91.32 (0.73)	93.65 (0.82)	95.05 (1.15)	98.48 (0.36)	**98.87** (0.38)
Kappa	-	85.46 (1.50)	94.60 (0.91)	91.56 (3.23)	90.34 (0.82)	92.92 (0.92)	94.50 (1.28)	98.31 (0.40)	**98.75** (0.43)
AA	-	91.21 (1.73)	91.65 (1.06)	95.56 (2.08)	88.61 (0.63)	95.79 (0.57)	95.46 (1.42)	98.17 (0.49)	**98.63** (0.49)

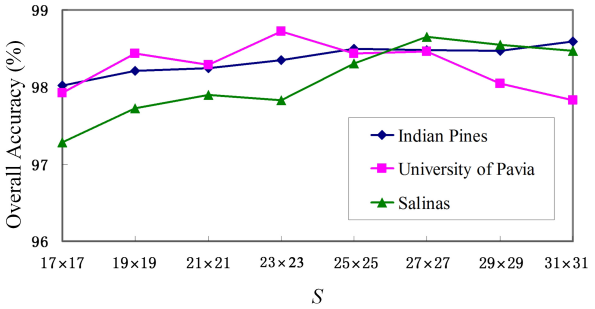


Fig. 10. Effect of S on classification accuracies in the three HSIs.

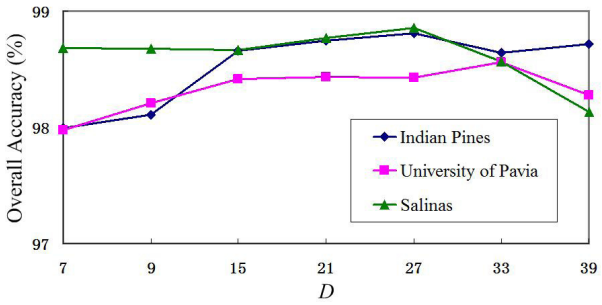


Fig. 11. Effect of D on classification accuracies in the three HSIs.

concretely, the OA value first rises and then dramatically declines. In addition, the curves reach the best OA values when S values are set to 25×25, 23×23, and 27×27 for the Indian Pines, University of Pavia, and Salinas images, respectively. The main reason is that the Indian Pines and Salinas images have larger smooth regions, and the University of Pavia image has more detailed regions.

3) Effect of D on Classification Accuracies: In this paper, we focus on building a very deep network to extract discriminative features of HSIs. Therefore, network depth is a crucial factor in our experiments. As the same in the above experimental setup, we fix the other two factors and analyzed the effect of D on classification results. Specifically, S is set to 25×25, 23×23, and 27×27 for the Indian Pines, University of Pavia, and Salinas images, respectively. N is set to 3, 5, and 10 for the three test images, respectively. As can be seen from Fig. 11, increasing D can improve the classification accuracies. However, too deep networks also result in the slight variations of accuracies. The main reason of this phenomenon is that the limited training samples (e.g., only 10%, 2%, and 0.5% of labeled pixels per class are used for the Indian Pines, University of Pavia, and Salinas images, respectively) are not enough to train networks with excessive depth. In addition, all OAs are above 98% under different D values, which demonstrate that our proposed network can effectively balance the network depth and classification performance.

F. Effect of Different Numbers of Training Samples

In this section, the influence of different training and test sets on several methods is analyzed on the Indian Pines, University of Pavia, and Salinas images, respectively. Due to the poor performance of the SVM and requirement of large samples by the SAE, the two methods are not discussed in this section. The parameters for the other methods are kept the same as that in Section IV-B. For the Indian Pines, University of Pavia, and Salinas images, different percentages (from 5% to 30% for the Indian Pines image, 0.5% to 5% for the University of Pavia image, and 0.1% to 2% for the Salinas image) of labeled pixels per class are randomly selected as training samples, and the rest of samples are used as test samples. Specifically, all experimental results are averaged over ten times with different randomly selected training data.

Fig. 12 shows the overall classification accuracy for each classifier under different numbers of training samples. From this curve, we can observe that the performances of all methods generally improve as the numbers of training samples

TABLE V
OA VALUES (IN PERCENTAGES) OBTAINED BY DIFFERENT FUSION STRATEGIES ON THE THREE HSIs

Images	Metrics	DRN	DFFN2 -LH	DFFN2 -MH	DFFN6	DFFN9	DFFN
Indian Pines	OA	98.36	98.40	98.38	98.07	97.02	**98.52**
	Kappa	98.13	98.24	98.20	97.80	96.60	**98.32**
	AA	97.62	97.59	97.55	97.64	95.56	**97.69**
University of Pavia	OA	98.52	98.54	98.62	98.39	97.32	**98.73**
	Kappa	98.03	98.06	98.22	97.77	97.69	**98.31**
	AA	96.78	96.96	97.20	96.07	96.05	**97.24**
Salinas	OA	98.48	98.55	98.59	98.54	98.30	**98.87**
	Kappa	98.31	98.33	98.29	98.38	98.11	**98.75**
	AA	98.17	98.23	98.32	98.20	97.70	**98.63**

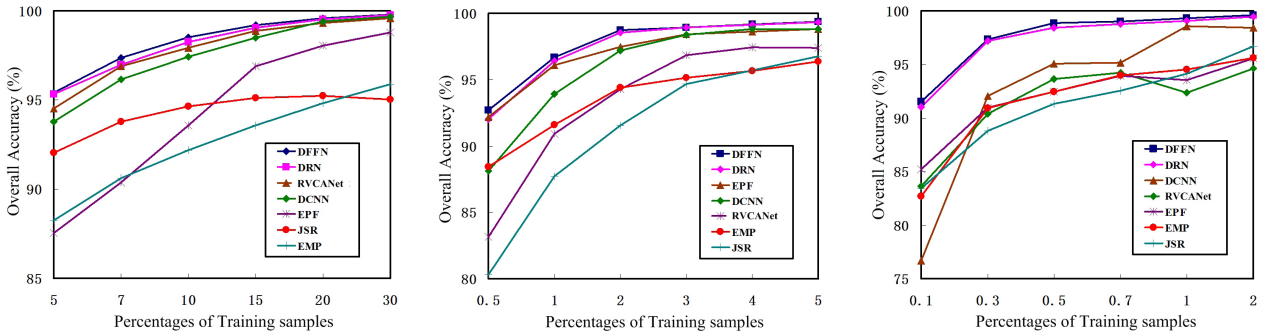


Fig. 12. Effect of the different numbers of training samples for EMP [11], EPF [25], JSR [15], RVCANet [47], DCNN, DRN [49], and DFFN on (a) Indian Pines image, (b) University of Pavia image, and (c) Salinas image.

increase. In addition, the proposed DFFN method consistently provides superior performances over the other compared methods under all different training samples. Specifically, when less samples are used, our proposed method has much advantage over other classifiers.

G. Comparison of Different Feature Fusion Strategies

In this paper, we fuse multiple-layer features to explore the strong complementary and correlated information existing in deep network from three different perspectives (i.e., low, middle, and high layers). In this section, we compare different fusion methods to prove the effectiveness of the proposed fusion strategy. Table V shows the OA values obtained by different fusion methods on the three HSIs. In Table V, the DFFN2, DFFN6, and DFFN9 refer to methods that fuse two, six, and nine hierarchical layers, respectively. Particularly, the DFFN2-LH represents the fusion of a low layer and a high layer, and the DFFN2-MH refers to the fusion of a middle layer and a high layer.

From Table V, we can observe that fusing several layers can improve the classification results to some extent compared with the DRN, and the proposed fusion strategy DFFN indeed outperforms other methods. However, fusing too many layers conversely may bring in redundant information, which greatly degrades the performance (e.g., DFFN9).

V. CONCLUSION

In this paper, a novel DL-based method is presented for HSI classification. Compared with the previous networks, the proposed DFFN can extract deeper features and obtain the state-of-the-art performance. In addition, a fusing mechanism is exploited to make full use of the multiple-layer features. The experimental results on three real HSIs demonstrate the superiority of the proposed method over several well-known classical and DL methods, in terms of both visual qualities on the classification map and quantitative metrics.

Although a feature-fusing mechanism is adopted in our proposed network, we only fuse three layers with the elementwise adding way. In the future works, we will systematically research the fusion strategies and fuse more representative layers to further improve the classification accuracies.

ACKNOWLEDGMENT

The authors would like to thank Dr. B. Pan for providing the software for the RVCANet. They would also like to thank the Editor-in-Chief, the Associate Editor, and the anonymous reviewers for their insightful comments and suggestions which have greatly improved this paper.

REFERENCES

[1] L. Zhang, L. Zhang, D. Tao, X. Huang, and B. Du, "Hyperspectral remote sensing image subpixel target detection based on supervised metric learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 4955–4965, Aug. 2014.

[2] B. Luo, C. Yang, J. Chanussot, and L. Zhang, "Crop yield estimation based on unsupervised linear unmixing of multidate hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 1, pp. 162–173, Jan. 2013.

[3] X. Yang and Y. Yu, "Estimating soil salinity under various moisture conditions: An experimental study," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2525–2533, May 2017.

[4] Y. Zhong and L. Zhang, "An adaptive artificial immune network for supervised classification of multi-/hyperspectral remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 3, pp. 894–909, Mar. 2012.

[5] 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.

[6] 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.

[7] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Spectral–spatial hyperspectral image segmentation using subspace multinomial logistic regression and Markov random fields," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 3, pp. 809–823, Mar. 2012.

[8] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Hyperspectral image segmentation using a new Bayesian approach with active learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 10, pp. 3947–3960, Oct. 2011.

[9] 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.

[10] J. A. Benediktsson, M. Pesaresi, and K. Amason, "Classification and feature extraction for remote sensing images from urban areas based on morphological transformations," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 9, pp. 1940–1949, Sep. 2003.

[11] J. A. Benediktsson, J. A. Palmason, and J. R. Sveinsson, "Classification of hyperspectral data from urban areas based on extended morphological profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 480–491, Mar. 2005.

[12] G. Camps-Valls, L. Gomez-Chova, J. Munoz-Mari, J. Vila-Frances, and J. Calpe-Maravilla, "Composite kernels for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 1, pp. 93–97, Jan. 2006.

[13] M. Fauvel, J. Chanussot, and J. A. Benediktsson, "A spatial–spectral kernel-based approach for the classification of remote-sensing images," *Pattern Recognit.*, vol. 45, no. 1, pp. 381–392, Jan. 2012.

[14] L. Fang, S. Li, W. Duan, J. Ren, and J. A. Benediktsson, "Classification of hyperspectral images by exploiting spectral–spatial information of superpixel via multiple kernels," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 12, pp. 6663–6674, Dec. 2015.

[15] 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.

[16] 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.

[17] L. Fang, S. Li, X. Kang, and J. A. Benediktsson, "Spectral–spatial classification of hyperspectral images with a superpixel-based discriminative sparse model," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4186–4201, Aug. 2015.

[18] L. Fang, C. Wang, S. Li, and J. A. Benediktsson, "Hyperspectral image classification via multiple-feature-based adaptive sparse representation," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 7, pp. 1646–1657, Jul. 2017.

[19] W. Fu, S. Li, L. Fang, X. Kang, and J. A. Benediktsson, "Hyperspectral image classification via shape-adaptive joint sparse representation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 2, pp. 556–567, Feb. 2016.

[20] L. Fang, H. Zhuo, and S. Li, "Super-resolution of hyperspectral image via superpixel-based sparse representation," *Neurocomputing*, vol. 273, pp. 171–177, Jan. 2018.

[21] T. Lu, S. Li, L. Fang, Y. Ma, and J. A. Benediktsson, "Spectral–spatial adaptive sparse representation for hyperspectral image denoising," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 1, pp. 373–385, Jan. 2016.

[22] T. Lu, S. Li, L. Fang, X. Jia, and J. A. Benediktsson, "From subpixel to superpixel: A novel fusion framework for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 8, pp. 4398–4411, Aug. 2017.

[23] S. Li, T. Lu, L. Fang, X. Jia, and J. A. Benediktsson, "Probabilistic fusion of pixel-level and superpixel-level hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7416–7430, Dec. 2016.

[24] L. Fang, N. He, S. Li, P. Ghamisi, and J. A. Benediktsson, "Extinction profiles fusion for hyperspectral images classification," *IEEE Trans. Geosci. Remote Sens.*, to be published, doi: 10.1109/TGRS.2017.2768479.

[25] 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.

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1097–1105.

[27] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[28] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 580–587.

[29] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[30] A. Bordes, X. Glorot, J. Weston, and Y. Bengio, "Joint learning of words and meaning representations for open-text semantic parsing," in *Proc. Int. Conf. Artif. Intell. Statist. (AISTATS)*, 2012, pp. 127–135.

[31] L. Zhang, L. Zhang, and B. Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geosci. Remote Sens. Mag.*, vol. 4, no. 2, pp. 22–40, Jun. 2016.

[32] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7405–7415, Dec. 2016.

[33] X. Yao, J. Han, G. Cheng, X. Qian, and L. Guo, "Semantic annotation of high-resolution satellite images via weakly supervised learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 6, pp. 3660–3671, Jun. 2016.

[34] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proc. IEEE*, vol. 105, no. 10, pp. 1865–1883, Oct. 2017.

[35] J. Han, D. Zhang, G. Cheng, L. Guo, and J. Ren, "Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 6, pp. 3325–3337, Jun. 2015.

[36] X. Lu, X. Zheng, and Y. Yuan, "Remote sensing scene classification by unsupervised representation learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 9, pp. 5148–5157, Sep. 2017.

[37] 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.

[38] X. Ma, H. Wang, and J. Geng, "Spectral–spatial classification of hyperspectral image based on deep auto-encoder," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 9, pp. 4073–4085, Sep. 2016.

[39] Y. Liu, G. Cao, Q. Sun, and M. Siegel, "Hyperspectral classification *via* deep networks and superpixel segmentation," *Int. J. Remote Sens.*, vol. 36, no. 13, pp. 3459–3482, Jul. 2015.

[40] 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.

[41] 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.

[42] P. Zhong, Z. Gong, S. Li, and C.-B. Schönlieb, "Learning to diversify deep belief networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 6, pp. 3516–3530, Jun. 2017.

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

[44] A. Romero, C. Gatta, and G. Camps-Valls, "Unsupervised deep feature extraction for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 3, pp. 1349–1362, Mar. 2016.

[45] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "Convolutional neural networks for large-scale remote-sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 645–657, Feb. 2017.

[46] W. Li, G. Wu, F. Zhang, and Q. Du, "Hyperspectral image classification using deep pixel-pair features," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 844–853, Feb. 2017.

[47] B. Pan, Z. Shi, and X. Xu, "R-VCANet: A new deep-learning-based hyperspectral image classification method," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 5, pp. 1975–1986, May 2017.

[48] W. Zhao and S. Du, "Spectral–spatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4544–4554, Aug. 2016.
[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[50] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.
[51] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Proc. 19th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, 2007, pp. 153–160.
[52] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," unpublished paper, 2015. [Online]. Available: https://arxiv.org/abs/1502.03167
[53] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.
[54] S. Chaib, H. Liu, Y. Gu, and H. Yao, "Deep feature fusion for VHR remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 8, pp. 4775–4784, Aug. 2017.
[55] X. Zhang *et al.*, "Deep fusion of multiple semantic cues for complex event recognition," *IEEE Trans. Image Process.*, vol. 25, no. 3, pp. 1033–1046, Mar. 2016.
[56] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[57] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, Apr. 2011, Art. no. 27.

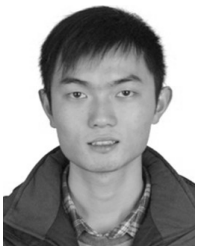
Weiwei Song (S'17) received the B.S. degree from the College of Electrical and Information Engineering, Southwest Minzu University, Chengdu, China, in 2015. He is currently pursuing the Ph.D. degree with the Laboratory of Vision and Image Processing, Hunan University, Changsha, China.

His research interests include hyperspectral image processing, deep learning, and spare representation.

Shutao Li (M'07–SM'15) received the B.S., M.S., and Ph.D. degrees from Hunan University, Changsha, China, in 1995, 1997, and 2001, respectively.

He was a Research Associate with the Department of Computer Science, The Hong Kong University of Science and Technology, Hong Kong, in 2011. From 2002 to 2003, he was a Post-Doctoral Fellow with the Royal Holloway College, University of London, London, U.K., with Prof. J. Shawe-Taylor. In 2005, he visited the Department of Computer Science, The Hong Kong University of Science and Technology, as a Visiting Professor. He joined the College of Electrical and Information Engineering, Hunan University, in 2001, where he is currently a Full Professor. He has authored or co-authored over 160 refereed papers. His research interests include compressive sensing, sparse representation, image processing, and pattern recognition.

Dr. Li is a member of the Editorial Board of the journals *Information Fusion* and *Sensing and Imaging*. He is currently an Associate Editor of the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING and the IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT. He was a recipient of two Second-Grade National Awards at the Science and Technology Progress of China in 2004 and 2006.



Leyuan Fang (S'10–M'14–SM'17) received the B.S. and Ph.D. degrees from the College of Electrical and Information Engineering, Hunan University, Changsha, China, in 2008 and 2015, respectively.

From 2011 to 2012, he was a Visiting Ph.D. Student with the Department of Ophthalmology, Duke University, Durham, NC, USA, supported by the China Scholarship Council. Since 2017, he has been an Associate Professor with the College of Electrical and Information Engineering, Hunan University. His research interests include sparse representation and multiresolution analysis in remote sensing and medical image processing.

Dr. Fang received the Scholarship Award for Excellent Doctoral Student from the Chinese Ministry of Education in 2011.

Ting Lu (S'16–M'17) received the B.S. and Ph.D. degrees from Hunan University, Changsha, China, in 2011 and 2017, respectively.

From 2014 to 2015, she was a Visiting Ph.D. Student with the Department of Information Engineering and Computer Science, University of Trento, Trento, Italy, supported by the China Scholarship Council. Since 2017, she has been an Assistant Professor with the College of Electrical and Information Engineering, Hunan University. Her research interests include sparse representation, image fusion, and remote sensing image processing.