

Classification of Hyperspectral Images by Gabor Filtering Based Deep Network

Xudong Kang¹, Senior Member, IEEE, Chengchao Li, Shutao Li², Senior Member, IEEE, and Hui Lin³

Abstract—In this paper, a novel spectral-spatial classification method based on Gabor filtering and deep network (GFDN) is proposed. First, Gabor features are extracted by performing Gabor filtering on the first three principal components of the hyperspectral image, which can typically characterize the low-level spatial structures of different orientations and scales. Then, the Gabor features and spectral features are simply stacked to form the fused features. Afterwards, deep features are captured by training a stacked sparse autoencoder deep network with the fused features obtained above as inputs. Since the number of training samples of hyperspectral images is often very limited, which negatively affects the classification performance in deep learning, an effective way of constructing virtual samples is designed to generate more training samples, automatically. By jointly utilizing both the real and virtual samples, the parameters of the deep network can be better trained and updated, which can result in classification results of higher accuracies. Experiments performed on four real hyperspectral datasets show that the proposed method outperforms several recently proposed classification methods in terms of classification accuracies.

Index Terms—Deep learning, Gabor filter, hyperspectral image (HSI) classification, stacked sparse autoencoders (SSAE), virtual samples.

I. INTRODUCTION

HYPERSPECTRAL images (HSI) captured by satellite sensors have several hundreds of spectral bands for each pixel. The rich spectral information preserved in these bands are highly related with the physical properties of different materials, which can be utilized to distinguish materials in the observed scene. Due to this advantage, hyperspectral imaging plays a significant role in a wide variety of remote-sensing applications, such as target detection [1]–[3], spectral unmixing [4], and classification [5]–[7].

During the last decades, a large number of classification methods have been proposed to address the hyperspectral image classification problem. Among these researches, intensive research

work has been focused on how to build powerful pixelwise classifiers for analyzing and classifying HSI, e.g., support vector machine (SVM) [8], random forest [9], multinomial logistic regression (MLR) [10] and neural networks [11]. In the aforementioned methods, the SVM has been proved to show a good performance in terms of classification accuracy [8], which presents low sensitivity to high dimensionality. Furthermore, the sparse representation, as a powerful tool in image representation and analysis, is also successfully introduced for HSI classification [12], [13].

Due to the fact that the spectral information of HSI pixels from different classes may be easily mixed, the aforementioned approaches, which only utilize the spectral information often lead to noisy classification results. To further improve the classification performance, many researchers are working on spectral-spatial classification, which incorporates the spatial contextual information into classification process. For example, the extended morphological attribute profiles (EMAP) [14], [15] is proposed to construct spectral-spatial features by a series of attribute profiles. In [16], the spatial information is exploited by a postprocessing procedure, which smooths multiple probability maps by edge-preserving filtering so as to get smoothed probabilities aligned with real object boundaries. Furthermore, the multiscale adaptive sparse representation model [17] is proposed for the improvement of classification performance, which exploits spatial information at multiple scales via an adaptive sparse strategy. Gabor features have currently been successfully applied for hyperspectral image analysis due to the ability to represent spatial structures of different scales and orientations in hyperspectral images [18], [19]. In addition, there are lots of works that utilize spectral and spatial information by other advancing tools, such as composite kernel methods [20], intrinsic image decomposition [21], extended random walker [22], [23], tensor representation [24], hypergraph [25], and anisotropic diffusion [26].

Recently, deep learning has been a very active research topic in the field of image processing, and has been applied in many applications that involve speech recognition [27], face representation [28], and image classification [29]. Deep learning model often involves deep architectures, which can extract more abstract and invariant features of data, leading to promising performances compared with traditional classifiers with shallower structures. In recent years, some deep models have been applied to HSI processing. In [30], stacked autoencoder (SAE) deep model is first proposed for HSI classification. Afterwards, an improved autoencoder (AE) deep model based on sparse

Manuscript received May 31, 2017; revised August 24, 2017 and September 24, 2017; accepted October 17, 2017. Date of publication November 21, 2017; date of current version April 11, 2018. This work was supported in part by the National Natural Science Fund of China for International Cooperation and Exchanges under Grant 61520106001, in part by the National Natural Science Foundation of China under Grant 61601179, in part by the National Natural Science Fund of China for Distinguished Young Scholars under Grant 61325007, and in part by the Fund of Hunan Province for 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: xudong_kang@163.com; chengchao_li@163.com; shutao_li@hnu.edu.cn; linhui1965@126.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/JSTARS.2017.2767185

constraint, i.e., stacked sparse autoencoder (SSAE), is proposed [31]. Deep belief network deep model is applied to HSI in [32]. These deep models have the ability of extracting the robust features and outperform other traditional classification methods in terms of classification accuracy [33], [34]. However, the aforementioned deep models need to learn and update lots of parameters in the process of training. Since available training samples of HSI are limited, the parameters of the network are hard to be trained optimally. Furthermore, the above mentioned deep models utilize fixed-size patch to extract spatial similarity information from principal component analysis (PCA) based main components, while rich structure information is neglected. Therefore, noisy classification results around edge areas are easily caused.

Different from previous deep learning based methods, the major motivation behind the proposed method is to make full use of the deep spectral and structural information of the hyperspectral images. Various feature extraction methods have been proposed, such as attribute profiles (APs) [14], extinction profiles (EPs) [35], [36], and local binary patterns (LBP)[37] which focus on higher-level spatial properties of objects, e.g., area of regions, standard deviation, length of the diagonal of the bounding box, and binary patterns in a local region. For example, Aptoula *et al.* proposed a method based on the combination of the attribute profiles and the convolution network [38]. In this paper, the Gabor filters are used to capture the spatial structures of different orientations and scales in images, which actually correspond to the low-level spatial structures in the scene. The deep learning network makes an important role in transferring the low-level features of a hyperspectral image, i.e., spectra, spatial structures of different orientations and scales, into deep features, and thus, is expected to be able to obtain higher classification accuracies.

The proposed Gabor filtering based deep network (GFDN) consists of the following steps. First, Gabor features are extracted from the first three principal components of HSI. Then, Gabor features and spectral features are combined to form the fused features. Next, the SSAE deep network yields deep features from the fused features, which are more abstract and robust features for classification. Furthermore, because limited training samples may negatively affect the parameters training and learning in deep model, a virtual sample constructing method is designed to simulate more training samples for network training. By this way, the parameters of the deep network can be better trained and updated by using both the real and virtual samples. Once the deep network training process is finished, an MLR, e.g., softmax regression, is used as the output layer of the deep network to solve the image classification problem.

The rest of the paper is organized as follows. Section II briefly introduces relevant works. In Section III, the proposed method is illustrated in detail. Experimental results and analysis are shown in Section IV, and the conclusions are summarized in Section V.

II. RELATED WORKS

A. Stacked Sparse Autoencoder

The sparse AE is the major component of the SSAE network, which can be used for learning high-level structured representations of the original data [39].

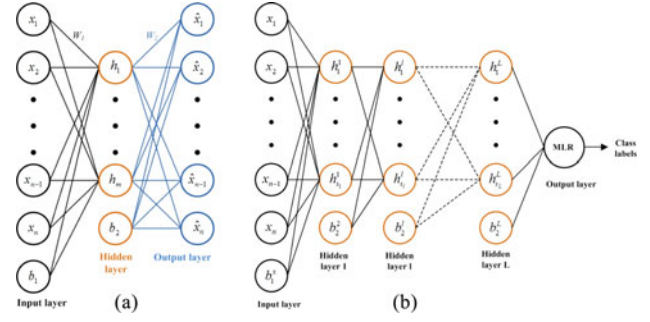


Fig. 1. Schematic of (a) a three-layer AE and (b) a deep network constructed using SSAE and MLR.

A sparse AE network, as shown in Fig. 1(a), typically consists of an input layer, a hidden layer, and an output layer, in which the output layer has the same dimensionality as the input layer. The process of training a sparse AE network includes the encode and decode processes. The input layer represents the original data or input feature vectors $\mathbf{x} \in \mathbb{R}^n$. The hidden layer serves as an encoder that gets corresponding feature representation $\mathbf{h} \in \mathbb{R}^m$ of the input data \mathbf{x} . The output layer can be considered as a decoder that yields the reconstruction $\hat{\mathbf{x}} \in \mathbb{R}^n$ of the input data \mathbf{x} . Training of the sparse AE network aims at finding appropriate parameters minimizing the reconstruction error between the input \mathbf{x} and its reconstruction $\hat{\mathbf{x}}$. Mathematically, it can be formulated as follows:

$$J_{\text{sparse}}(\theta) = \frac{1}{2K} \sum_{i=1}^K \|\hat{\mathbf{x}}_i - \mathbf{x}_i\|_2^2 + \beta \sum_{j=1}^{s_l} KL(\rho \|\hat{\rho}_j) \quad (1)$$

$$\mathbf{h} = f(\mathbf{W}_1 \mathbf{x}_i + \mathbf{b}_1) \quad (2)$$

$$\hat{\mathbf{x}} = f(\mathbf{W}_2 \mathbf{h} + \mathbf{b}_2) \quad (3)$$

where K is the number of training examples, and $f(\cdot)$ is the activation function, such as sigmoid function $f(x) = 1/(1 + \exp(-x))$. Through learning the encoding weights \mathbf{W}_1 , the decoding weights \mathbf{W}_2 , and the biases $\{\mathbf{b}_1, \mathbf{b}_2\}$ with the stochastic gradient descent algorithm [40], the process of minimizing the reconstruction error can be efficiently implemented. Here, $KL(\rho \|\hat{\rho}_j)$ is the Kullback–Leibler (KL) divergence, denoted as follows:

$$KL(\rho \|\hat{\rho}_j) = \rho \log \frac{\rho}{\hat{\rho}_j} + (1 - \rho) \log \frac{1 - \rho}{1 - \hat{\rho}_j} \quad (4)$$

where ρ represents the sparsity parameter that is typically a small value close to 0, and $\hat{\rho}_j$ is the average activation of hidden unit j , which is averaged over the training set. We would like the average activation $\hat{\rho}_j$ of each hidden unit j to be close to ρ , aims at penalizing $\hat{\rho}_j$ deviating significantly from ρ and achieving the sparsity constraint.

As shown in Fig. 1(b), multiple sparse AEs construct an SSAE deep network, in which the feature representation \mathbf{h} of each sparse AE is the input of the successive sparse AE. The loss function of the SSAE deep network over the whole training set is defined as

$$J(\mathbf{W}, \mathbf{b}) = J_{\text{sparse}}(\theta) + \frac{\lambda}{2} \sum_{l=1}^L \sum_{i=1}^{s_{l+1}} \sum_{j=1}^{s_l} (W_{ij}^l)^2. \quad (5)$$

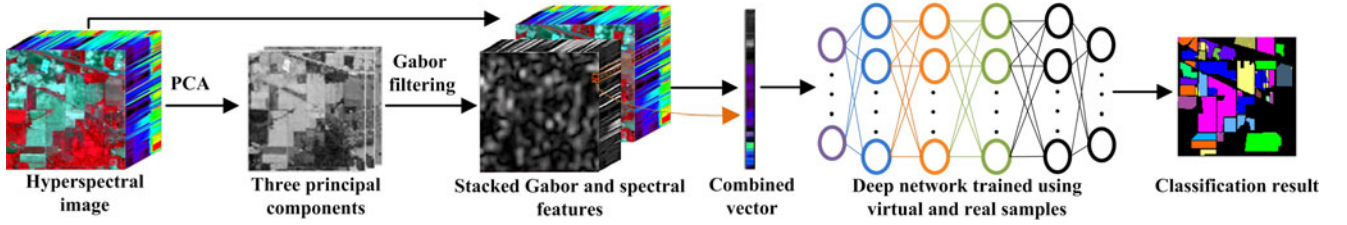


Fig. 2. Schematic of the proposed GFDN method

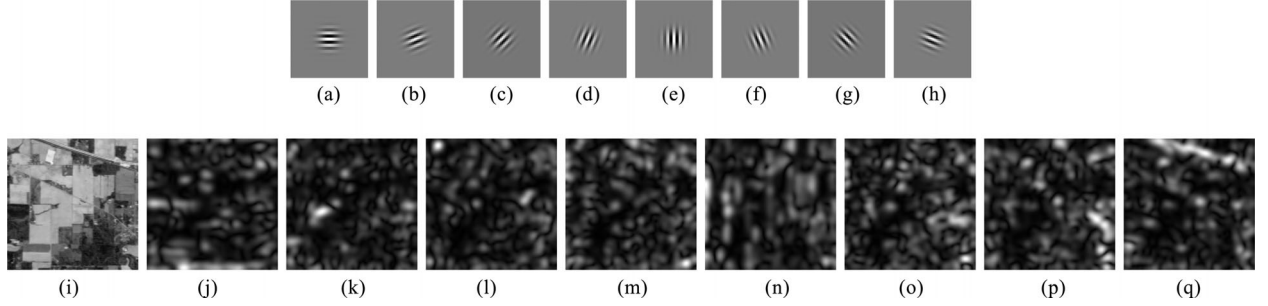


Fig. 3. Gabor filters and Gabor features. (a)-(h) Gabor filters with 8 different orientations, (i) First principal component of the Indian Pines image, (j)-(q) Convolution results of first principal component of the Indian Pines image.

The second term is a regularization term, or called a weight decay term, which aims at decreasing the magnitude of the weights, and helps to prevent overfitting. λ is the weight decay parameter. L is the number of layers and s_l is the number of units in the l th layer. W_{ij}^l denotes the weight between the j th unit in the l th layer and the i th unit in the $(l + 1)$ th layer.

B. Gabor Filter

The Gabor filter is modulated by a Gaussian function and a sinusoidal plane wave. Discriminant features can be extracted from Gabor filtered images, specially texture and orientation information [41]–[43]. The generating function can be defined as follows:

$$\Phi_{u,v}(x, y) = \frac{f^2}{\pi\gamma\eta} \exp(-(\alpha^2 x'^2 + \beta^2 y'^2)) \exp(j2\pi f x') \quad (6)$$

$$x' = \left(x - \frac{m+1}{2}\right) \cos \theta + \left(y - \frac{n+1}{2}\right) \sin \theta \quad (7)$$

$$y' = -\left(x - \frac{m+1}{2}\right) \sin \theta + \left(y - \frac{n+1}{2}\right) \cos \theta \quad (8)$$

where f is the distribution coefficient of the Gabor function in the frequency domain, and θ represents the rotation angle of the Gaussian function. α defines the sharpness of the Gaussian function along the major axis parallel to the wave. β is the sharpness of the Gaussian function along the minor axis perpendicular to the wave. $\gamma = \frac{f}{\alpha}$ and $\eta = \frac{f}{\beta}$ are defined as the constraint conditions that keep the ratio between frequency and sharpness. α and β are assumed to be equal and $\gamma = \eta = \sqrt{2}$, is adopted in most of previous researches [44]. Generally, m and n are equal to an odd number d , which represents the size of the Gabor filter, i.e., $d = m = n$.

III. PROPOSED METHOD

Fig. 2 shows the schematic of the proposed GFDN method for HSI classification. In general, the proposed method consists of following three main parts:

- 1) Extraction of Gabor features,
- 2) construction of virtual samples, and
- 3) deep feature learning and classification based on the SSAE deep network.

A. Extraction of Gabor features

To extract Gabor features, a set of Gabor filters with different frequencies and orientations are required as follows:

$$f = \frac{f_{\max}}{\sqrt{2}^u}, \theta = \frac{v}{8}\pi \quad u = 0, \dots, U-1, v = 0, \dots, V-1. \quad (9)$$

Here, f_{\max} is the highest peak frequency of Gabor function. f and θ are the different frequencies and the orientation angles of Gabor filters, respectively. U is the number of scales of Gabor filters, which determines different frequencies. V represents the number of orientations of Gabor filters determining different orientations.

In the spatial domain, the Gabor features $\mathbf{G}_{u,v}$ of HSI can be represented via convolving the first three principal components \mathbf{I}^{PCA} of HSI with a family of Gabor filters $\Phi_{u,v}$. The convolution of a principal component $\mathbf{I}^{\text{PCA}}(x, y)$ and Gabor filter $\Phi_{u,v}(x, y)$ can be defined as follows:

$$\mathbf{G}_{u,v}(x, y) = |\mathbf{I}^{\text{PCA}}(x, y) * \Phi_{u,v}(x, y)| \quad (10)$$

where $\mathbf{G}_{u,v}(x, y)$ denotes the Gabor features that is the convolution result corresponding to the Gabor filter at orientation u and scale v . Here, $|\cdot|$ denotes the magnitude of the convolution. Fig. 3 shows the convolution results of the first principal component of the Indian Pines image with eight Gabor filters at eight different orientations of one scale.

B. Construction of Virtual Samples

It is known that the SSAE deep network needs to train a lot of weight parameters. In order to obtain proper weights, extensive training samples are required in the process of training. The objective of constructing virtual samples is to generate new training samples from the real training samples. Based on the fact that the objects of the same class usually represent similar hyperspectral characteristics in a certain range, a virtual sample \mathbf{v}_i can be generated from two training samples of the same class.

In this paper, a virtual sample is generated according to the affinity between two training samples coming from the same class. The affinity between \mathbf{x}_i and \mathbf{x}_j can be defined as follows:

$$\mathbf{A}_{ij} = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2). \quad (11)$$

Certainly, many different ways can be exploited to define an affinity matrix, but the heat kernel, as defined above, has been proven to generate very effective locality preserving properties [45]. Hence, the virtual sample \mathbf{v}_i is constructed by the weighted average of two real training samples \mathbf{x}_i and \mathbf{x}_j coming from same class as follows:

$$\mathbf{v}_i = \mathbf{A}_{ij}\mathbf{x}_i + (1 - \mathbf{A}_{ij})\mathbf{x}_j. \quad (12)$$

Virtual sample \mathbf{v}_i is assigned with the same class label as the label of the samples \mathbf{x}_i and \mathbf{x}_j . Furthermore, in order to increase the discrimination of virtual samples, low-correlated real training samples are randomly chosen to construct virtual samples, in which the correlation $\rho_{\mathbf{x}_i, \mathbf{x}_j}$ between samples from the selected real training samples is compared by the Pearson correlation coefficients [46] as follows:

$$\rho_{\mathbf{x}_i, \mathbf{x}_j} = \frac{\sum(\mathbf{x}_i - \bar{\mathbf{x}}_i)(\mathbf{x}_j - \bar{\mathbf{x}}_j)}{\sqrt{\sum(\mathbf{x}_i - \bar{\mathbf{x}}_i)^2 \sum(\mathbf{x}_j - \bar{\mathbf{x}}_j)^2}}. \quad (13)$$

C. Feature Learning and Classification Based on the SSAE

After finishing Gabor feature extraction, spatial information of the Gabor features $\mathbf{G}_{u,v}$ and the original HSI are stacked to form the combined features, in which each combined feature vector is $\mathbf{x}_i \in \mathbb{R}^n$. Afterwards, virtual samples \mathbf{v} are constructed from selected real training samples. Therefore, all of labeled training samples including virtual and real samples can be represented as $\mathbf{R} = \{(\mathbf{r}_i, c_i) | \mathbf{r}_i = \mathbf{x}_i \cup \mathbf{v}_i, \mathbf{x}_i \in \mathbb{R}^n, \mathbf{v}_i \in \mathbb{R}^n, c_i \in C, i = 1, \dots, K\}$. Here, K is the number of total training samples, and c_i denotes the corresponding label. C is the label set of all training samples. In order to integrate more abstract and representative features, the SSAE deep network consists of multilayers. Meanwhile, a set of training samples \mathbf{r} are used to train the SSAE deep network. During the procedure of training, the training sample \mathbf{r}_i can yield the feature representation $\mathbf{h}^{(l,i)}$ corresponding to l th layer. The feature representation $\mathbf{h}^{(l,i)}$ of l th layer is used as the input of the next hidden layer as follows:

$$\mathbf{h}^{(l+1,i)} = f(\mathbf{W}^{l+1}\mathbf{h}^{(l,i)} + \mathbf{b}^{l+1}). \quad (14)$$

The whole training process is repeated until the feature representation of all subsequent hidden layers is obtained. When all of the training samples \mathbf{r} have experienced such training process, the parameters \mathbf{W}^l , and \mathbf{b}^l of each layer l are learned

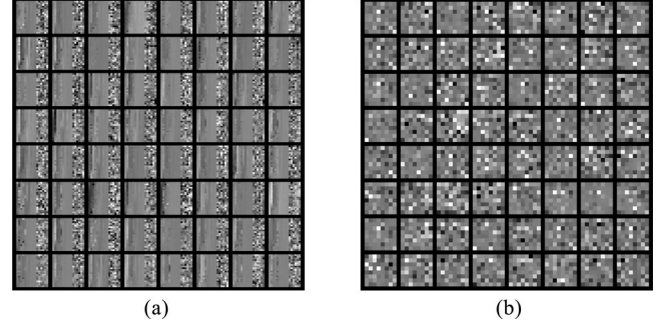


Fig. 4. Learned weights of the first and second sparse AE layers on the Indian Pines image. There are 64 hidden units per layer, thus, we can see 64 tiny weight images. The intensity of each pixel stands for the value of corresponding weight. (a) Learned weights of the first layer. (b) Learned weights of the second layer.

and updated thoroughly. In other words, the SSAE deep network yields a map function that is $f(\mathbf{W}, \mathbf{b})$. This map function can transform the raw features to the corresponding deep feature representations. Fig. 4 shows the learned weights of the first and second sparse AE layers on the Indian Pines image. It is indicated that the different features are extracted by different units of different layers, and the extracted features are very abstract.

Finally, the whole pretrained network are tuned with a MLR, which uses softmax, as its output layer activation to get the final class label for each pixel. Given S classes, for any test sample \mathbf{x}_i and the corresponding deep feature representation $\mathbf{h}^{(L,i)}$, the class probability of \mathbf{x}_i belonging to the r th class is defined as

$$P(c_i = s | \mathbf{x}_s) = \frac{e^{\mathbf{W}_s \mathbf{h}^{(L,i)} + \mathbf{b}_s}}{\sum_{s=1}^S e^{\mathbf{W}_s \mathbf{h}^{(L,i)} + \mathbf{b}_s}}, s = 1, \dots, S \quad (15)$$

where \mathbf{W}_s and \mathbf{b}_s are the weights and biases of the MLR output layer. Therefore, the class label for the sample \mathbf{x}_i is decided by

$$\text{Class}(\mathbf{x}_i) = \arg \max_{s=1, \dots, S} P(c_i = s | \mathbf{x}_i). \quad (16)$$

IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

A. Experimental Setup

1) *Datasets*: Experiments are performed on four hyperspectral datasets, which are the Indian Pines image, the University of Pavia image, the Salinas image, and the Houston image. The Indian Pines image, which records the agricultural Indian Pine test site in Northwestern Indiana, was acquired by the Airborne visible/infrared imaging spectrometer (AVIRIS) sensor. The image is of size 145×145 pixels with 220 spectral reflectance bands in the wavelength ranging from 0.4 to $2.5 \mu\text{m}$, which has a spatial resolution of 20 m per pixel. Before the experiments, 20 water absorption bands, no. 104–108, 150–163, and 220, are removed. The color composite of the Indian Pines image and the corresponding ground truth data are presented in Fig. 5(a)–(c), which consists of 16 reference classes.

The University of Pavia image was recorded by the reflective optics system imaging spectrometer (ROSIS-03) satellite sensor, which captures the campus area in the University of Pavia. The image is of size 610×340 pixels with 115 bands in the wavelength ranging from 0.43 to $0.86 \mu\text{m}$, which has a spatial

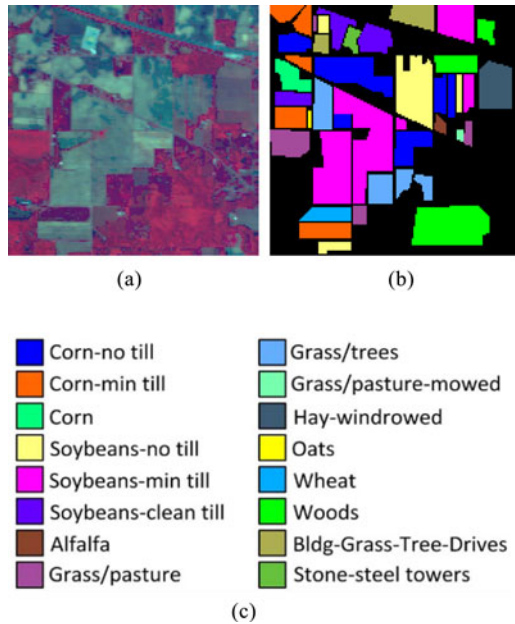


Fig. 5. (a) Three-band color composite of the Indian Pines image. (b) and (c) Corresponding ground truth data of the Indian Pines image.

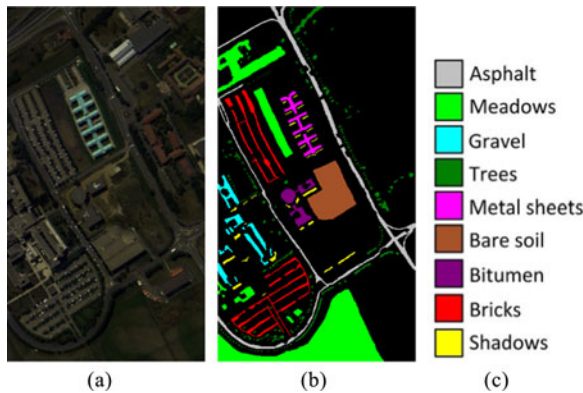


Fig. 6. (a) Three-band color composite of the University of Pavia image. (b) and (c) Corresponding ground truth data of the University of Pavia image.

resolution of 1.3 m per pixel. Due to high noise, 12 most noisy bands are removed before the experiments. The color composite of the University of Pavia image and the corresponding ground truth data are presented in Fig. 6(a)–(c), which contains 9 reference classes.

The Salinas image, which records the area of Salinas Valley, CA, USA, was acquired by the AVIRIS sensor. The image is of size 512×217 pixels with 224 spectral reflectance bands and a spatial resolution of 3.7 m per pixel. Before the experiments, 20 water absorption bands, no. 108–112, 154–167, and 224, are removed. The color composite of the Salinas image and the corresponding ground truth data are presented in Fig. 7(a)–(c), which contains 16 different classes.

The Houston image, which was acquired over the University of Houston, consists of size 349×1905 pixels. It has 144 spectral reflectance bands covering a spectra range from 380 to 1050 nm. The color composite of the Houston image and the

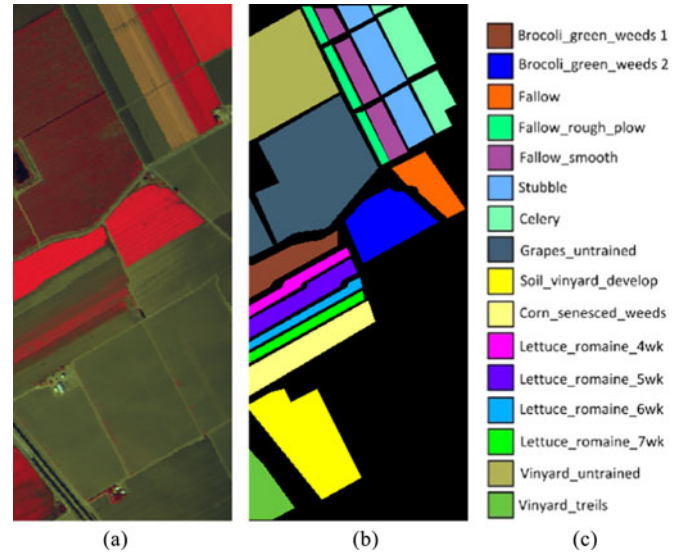


Fig. 7. (a) Three-band color composite of the Salinas image. (b) and (c) Corresponding ground truth data of the Salinas image.

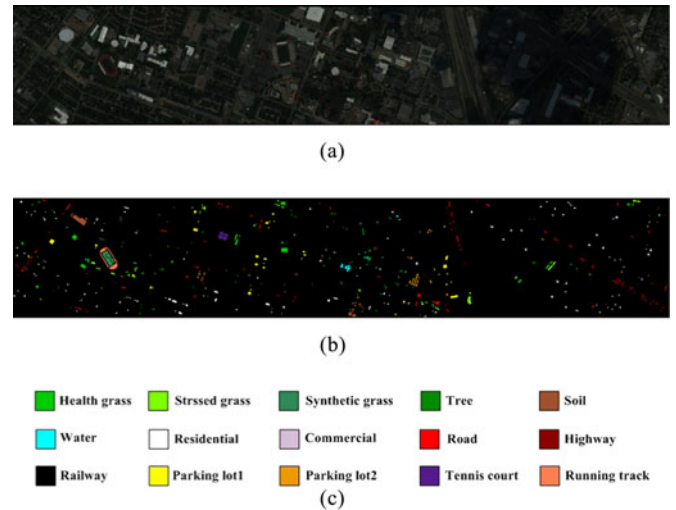


Fig. 8. (a) Three-band color composite of the Houston image. (b) and (c) Corresponding ground truth data of the Houston image.

corresponding ground truth data are presented in Fig. 8(a)–(c), which contains 15 different classes.

B. Classification Results

In this section, the proposed GFDN method is compared with several widely used classification methods: SVM [8], SAE [30], EMAP [14], sparse representation based classification (SRC) [12], edge-preserving filtering (EPF) [16], joint sparse representation classification (JSRC) [12], and multiple feature learning (MFL) [47]. The SVM method is implemented in the LIBSVM library [48], in which the parameters C and σ are obtained by fivefold cross-validation. For the SRC and JSRC, their parameters are set to the default values given in [12]. For the SAE, the parameters of the deep learning network are kept the same as the proposed method and fixed-size patch is set as 7 to extract spatial similarity information from 10 principal components.

TABLE I
NUMBER OF TRAINING AND TEST SAMPLES OF THE INDIAN PINES IMAGE AND CLASSIFICATION ACCURACIES (IN %) FOR THE SVM [8], SRC [12], SAE [30], EMAP [14], JSRC [12], MFL [47], EPF [16], GFDN*, AND GFDN METHODS

| Class | Training | Test | SVM | SRC | SAE | EMAP | JSRC | MFL | EPF | GFDN* | GFDN |
|-----------------|----------|------|-------|-------|-------|------------|-------|--------------|------------|--------------|--------------|
| Alfalfa | 4 | 42 | 46.73 | 27.27 | 36.53 | 95.48 | 95.50 | 91.63 | 89.59 | 98.37 | 94.90 |
| Corn-N | 115 | 1313 | 72.05 | 53.54 | 68.04 | 89.77 | 92.04 | 91.87 | 88.87 | 95.94 | 97.17 |
| Corn-M | 67 | 763 | 66.21 | 47.78 | 59.75 | 94.82 | 90.56 | 95.36 | 95.57 | 98.14 | 98.23 |
| Corn | 19 | 218 | 48.28 | 44.88 | 32.37 | 87.80 | 89.47 | 80.14 | 94.58 | 98.19 | 98.60 |
| Grass-P | 39 | 444 | 90.31 | 78.21 | 77.16 | 91.80 | 96.19 | 91.73 | 96.48 | 94.53 | 96.63 |
| Grass-T | 59 | 671 | 92.53 | 83.53 | 94.45 | 97.60 | 94.91 | 98.14 | 92.03 | 97.71 | 98.79 |
| Grass-P-M | 3 | 25 | 73.91 | 75.00 | 32.17 | 94.40 | 96.78 | 94.35 | 100 | 90.43 | 94.78 |
| Hay-W | 39 | 439 | 96.17 | 89.66 | 98.62 | 100 | 99.32 | 99.00 | 97.74 | 99.15 | 99.93 |
| Oats | 2 | 18 | 43.89 | 42.86 | 22.22 | 72.22 | 79.18 | 85.00 | 30.00 | 77.78 | 95.56 |
| Soybean-N | 78 | 894 | 71.88 | 60.53 | 62.46 | 84.65 | 91.10 | 89.10 | 90.78 | 95.27 | 96.70 |
| Soybean-M | 197 | 2258 | 80.32 | 68.67 | 81.51 | 95.53 | 93.99 | 95.89 | 90.71 | 98.37 | 99.01 |
| Soybean-C | 48 | 545 | 67.29 | 48.31 | 47.98 | 90.64 | 91.84 | 85.99 | 95.36 | 95.99 | 97.32 |
| Wheat | 17 | 188 | 95.49 | 85.15 | 91.23 | 99.41 | 92.66 | 99.59 | 99.55 | 97.90 | 98.05 |
| Woods | 102 | 1163 | 94.18 | 90.09 | 88.82 | 99.40 | 97.10 | 99.17 | 94.14 | 99.36 | 99.94 |
| Buildings-G-T-D | 31 | 355 | 49.54 | 48.55 | 57.88 | 91.75 | 91.66 | 88.11 | 92.07 | 97.11 | 98.57 |
| Stone-S-T | 8 | 85 | 89.08 | 95.06 | 86.21 | 95.18 | 89.05 | 88.39 | 91.18 | 88.16 | 95.63 |
| OA | – | – | 78.60 | 67.42 | 74.38 | 93.72 | 93.53 | 93.91 | 92.49 | 97.30 | 98.29 |
| AA | – | – | 73.62 | 64.94 | 64.84 | 92.53 | 92.58 | 92.09 | 89.92 | 95.15 | 97.49 |
| Kappa | – | – | 75.51 | 62.93 | 70.31 | 92.83 | 92.62 | 93.04 | 91.41 | 96.93 | 98.06 |

For the EMAP, the attribute profiles are extracted from the first three principle components with different attributes:

- 1) Area of the regions a ;
- 2) length of the diagonal of the bounding box δ ; and
- 3) standard deviation ε .

The threshold values of a are set to be 10, 15, and 20. The threshold values of δ are set as 50, 100, and 500. The threshold values of ε are set as 10, 30, and 50. Furthermore, the MFL method is implemented using the code which is available on Dr. Li's homepage.¹ The EPF method is implemented using the code which is available on Dr. Kang's homepage.² For the quantitative comparison and analysis, the overall accuracy (OA), the average accuracy (AA), and the Kappa coefficient are adopted to evaluate the classification results.

1) Parameters Setting: In the experiments, the parameters of the proposed GFDN methods are empirically selected and kept unchanged for the four test images. As for the Gabor filter, the eight orientations θ of $0 \sim 180$ for the proposed GFDN method are considered, $[0, \frac{\pi}{8}, \frac{\pi}{4}, \frac{3\pi}{8}, \frac{\pi}{2}, \frac{5\pi}{8}, \frac{3\pi}{4}, \frac{7\pi}{8}]$. The scale U and the size d of Gabor filter are set as 5 and 55, respectively, because they can give both high classification accuracies and an acceptable computing burden. For the deep network part, the SSAE model consists of two layers of sparse AE, in which each layer has 100 hidden units with 400 epochs of fine-tuning and iterations. Typically, the weight decay penalty λ is close to 0 and the default value is $1e^{-4}$. The sparsity parameter ρ is set to be 0.05. In the following section, the influences of associated parameters on the performance of the proposed approach will be further analyzed.

2) Comparison of Different Classification Methods: The first experiment is conducted on the Indian Pines image. In this experiment, the real training samples are chosen randomly

which account for 8% of each class and the rest is used as the test samples (see the second and third columns in Table I). Moreover, virtual samples are constructed by low-correlated ($\rho_{x_i, x_j} < 0.7$) real labeled data from the randomly selected training samples. Then, real and virtual training samples are combined together to train the deep network. The visual classification maps associated with the corresponding OA scores for different methods are shown in Fig. 9. As shown in this figure, the SVM and SRC methods present noisy estimations in their classification maps because of only exploiting the spectral information. By incorporating the spatial information of the HSI, the EMP, JSRC, MFL, and EPF methods can exhibit smoother performances in their classification results. Due to the limited number of training samples and negligence of rich structural information, the SAE also presents noisy classification result. By contrast, the proposed GFDN approach not only delivers better performances but also achieves accurate estimations in the edge area. Table I presents the number of training and test samples and the classification accuracies for different methods. The classification accuracies represented in Table I are the average results over ten experiments with the same number of randomly selected training samples. From Table I, it can be observed that the proposed GFDN approach outperforms the other compared methods, in terms of OA, AA, and Kappa coefficient. The GFDN* reports the result obtained by the proposed method without using the virtual samples for training. By comparing the results of GFDN* and GFDN in Table I, it can be seen that the GFDN can get better classification accuracies than GFDN*. This indicates that virtual samples indeed contribute to the proposed deep learning network. Furthermore, it is interesting to find that the classification accuracy of the Oats can be increased from 77.78% to 95.96% with only one added virtual sample for this class. In our view, the major reason behind this phenomenon is two folds. First, it is observed that the proposed Gabor filtering based deep network without the virtual sample construction step

¹<http://www.lx.it.pt/~jun/>

²<http://xudongkang.weebly.com/>

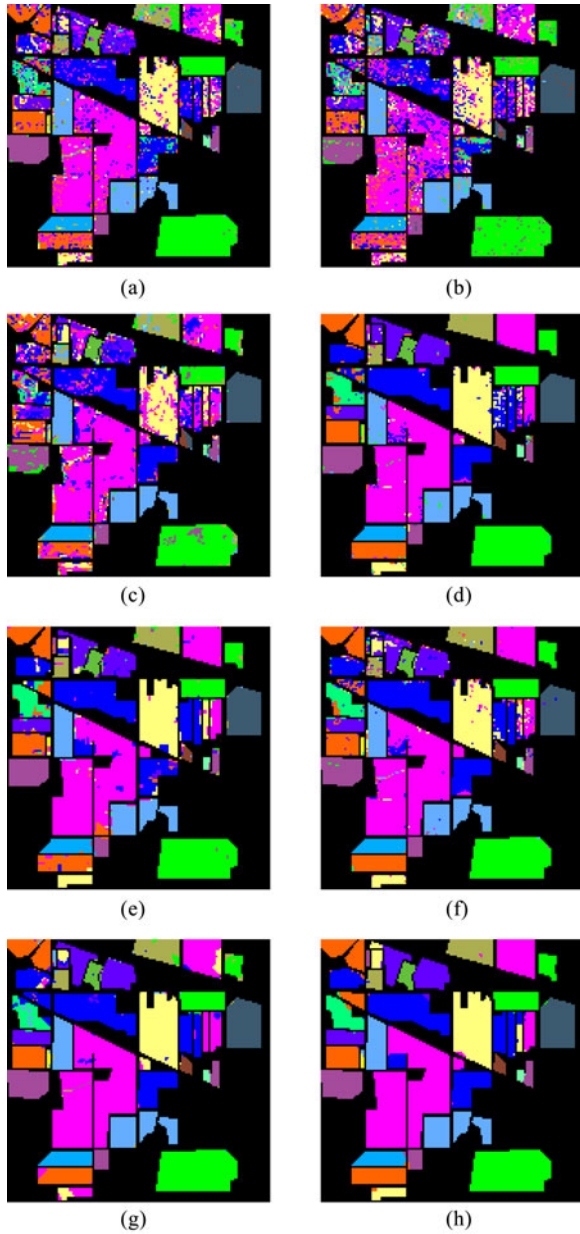


Fig. 9. Classification results (OA in %) of the Indian Pines image obtained by the (a) SVM [8], (b) SRC [12], (c) SAE [30], (d) EMAP [14], (e) JSRC [12], (f) MFL [47], (g) EPF [16], and (h) GFDN methods. (a) OA = 80.02, (b) OA = 67.42, (c) OA = 74.51, (d) OA = 93.86, (e) OA = 93.56, (f) OA = 94.06, (g) OA = 93.19, (h) OA = 98.62.

(GFDN*) already performs very well for the imbalanced training and test dataset, compared with the original SAE method. Second, although only one virtual sample is added for the Oats class, there are much more virtual samples for the other classes, such as the Corn-N and Soybean-M classes. These virtual samples all contribute in constructing more robust features of higher levels, and thus, also could lead to the increasement of accuracy for the Oats class.

The second and third experiments are carried on the University of Pavia image and the Salinas image, respectively. For the University of Pavia dataset, 200 samples are selected as the training samples for each class (see the second and third

columns of Table II). For the Salinas image, 2% of the labeled data is randomly selected as the real training samples and the remaining samples are used as the test set (see the second column in Table III). Similarly, we adopt the real labeled data per class that have been selected and are low-correlated ($\rho_{x_i, x_j} < 0.7$) to construct virtual samples in these two experiments. Afterward, the deep network is trained by real and virtual training samples. The classification results and quantitative metrics (averaged over ten experiments) obtained by the compared methods on the University of Pavia image and the Salinas image are represented in Figs. 10 and 11, and Tables II and III. It can be seen that, the proposed GFDN classification approach achieves better performances than other compared methods in terms of visual quality and objective metrics.

The last experiment is conducted on the Houston image. About 10% labeled pixels per class are randomly selected as the training samples and the remainder is used for test (see the second and third columns in Table IV). The virtual samples are constructed using the same way mentioned above. Table IV reports the quantitative classification results of various classification methods. As shown in this table, the proposed GFDN classification approach achieves the best performance in terms of the highest OA, AA, and Kappa.

All the experiments in this paper are performed on a laptop computer with an Intel Core i5-3210 CPU 2.50 GHz and 8 GB RAM. Table V reports the computational time of each step for the proposed GFDN method on the three test images. It can be observed that the main computational cost is occupied by the deep network training and fine-tuning process. The step for training the first layer takes 92.03, 97.95, and 99.26 s for the Indian Pines, University of Pavia, and Salinas images, respectively, which is the most time consuming step. Since the deep network training and fine-tuning processes are quite time consuming, one of our ongoing work is to adopt the graphics processing unit to greatly accelerate these processes.

C. Analysis of the Influence of Parameters

1) *Effect of the Scale and Size of Gabor filter:* In this section, the influence of the two parameters, i.e., the scale U and size d of filter, are analyzed. The experiment is performed on the University of Pavia dataset (see Fig. 12). As shown in Fig. 12, when the values of U and d are relatively small, the accuracies of the proposed method decrease significantly. It means that when U and d are very small, useful discriminative information will be lost in the Gabor filtering process. The proposed method can obtain satisfactory accuracies when the number of scale U is larger than 5 and the size d of filter is larger than 55. Moreover, as shown in Fig. 12(d), the computing time of the proposed method increases when the number of scale U and the size d of filter is increasing, which leads to a tradeoff between computing efficiency and classification accuracy. In this paper, $U = 5$ and $d = 55$ are set to be the default parameters because they can give both high classification accuracies and an acceptable computing burden.

2) *Effect of the Number of Units Per Layer and the Sparsity Parameter:* Here, the effect of the number of units per layer and

TABLE II

NUMBER OF TRAINING AND TEST SAMPLES OF THE UNIVERSITY OF PAVIA IMAGE AND CLASSIFICATION ACCURACIES (IN %) FOR THE SVM [8], SRC [12], SAE [30], EMAP [14], JSRC [12], MFL [47], EPF [16], GFDN*, AND GFDN METHODS

| Class | Training | Test | SVM | SRC | SAE | EMAP | JSRC | MFL | EPF | GFDN* | GFDN |
|--------------|----------|-------|-------|-------|------------|--------------|-------|-------|--------------|-------|--------------|
| Asphalt | 200 | 6431 | 80.89 | 63.07 | 87.73 | 95.56 | 84.19 | 97.48 | 98.74 | 98.35 | 98.47 |
| Meadows | 200 | 18449 | 81.07 | 79.66 | 94.91 | 98.88 | 96.26 | 98.74 | 99.62 | 96.36 | 97.99 |
| Gravel | 200 | 1899 | 77.60 | 68.14 | 93.00 | 97.93 | 99.10 | 90.80 | 97.18 | 98.97 | 99.87 |
| Trees | 200 | 2864 | 96.36 | 92.05 | 98.12 | 98.23 | 94.02 | 98.14 | 98.51 | 97.07 | 98.38 |
| Metal sheets | 200 | 1145 | 99.48 | 99.75 | 100 | 99.67 | 97.33 | 99.41 | 98.63 | 99.50 | 99.90 |
| Bare Soil | 200 | 4829 | 80.53 | 68.27 | 90.66 | 99.50 | 99.43 | 98.25 | 94.08 | 99.26 | 99.13 |
| Bitumen | 200 | 1130 | 85.43 | 86.73 | 92.61 | 99.76 | 98.31 | 98.25 | 96.33 | 99.76 | 99.96 |
| Bricks | 200 | 3482 | 75.06 | 72.47 | 91.47 | 97.59 | 96.08 | 96.44 | 93.76 | 98.62 | 98.62 |
| Shadows | 200 | 747 | 98.38 | 97.99 | 99.85 | 99.95 | 76.01 | 99.80 | 99.29 | 97.74 | 99.93 |
| OA | — | — | 82.32 | 76.53 | 93.29 | 98.30 | 94.41 | 97.90 | 97.94 | 97.58 | 98.51 |
| AA | — | — | 86.09 | 80.90 | 94.26 | 98.56 | 93.41 | 97.48 | 97.35 | 98.40 | 99.14 |
| Kappa | — | — | 77.48 | 69.37 | 91.11 | 97.72 | 92.55 | 97.19 | 97.24 | 96.79 | 98.01 |

TABLE III

NUMBER OF TRAINING AND TEST SAMPLES OF THE SALINAS IMAGE AND CLASSIFICATION ACCURACIES (IN %) FOR THE SVM [8], SRC [12], SAE [30], EMAP [14], JSRC [12], MFL [47], EPF [16], GFDN*, AND GFDN METHODS

| Class | Training | Test | SVM | SRC | SAE | EMAP | JSRC | MFL | EPF | GFDN* | GFDN |
|-------------|----------|-------|-------|-------|--------------|--------------|--------------|--------------|--------------|-------|--------------|
| weeds_1 | 41 | 1968 | 98.61 | 98.69 | 97.09 | 99.01 | 100 | 99.84 | 100 | 98.74 | 99.76 |
| weeds_2 | 75 | 3651 | 99.07 | 98.72 | 99.54 | 99.78 | 99.74 | 99.49 | 99.89 | 98.76 | 99.67 |
| Fallow | 40 | 1936 | 95.79 | 96.14 | 96.65 | 99.82 | 99.43 | 99.96 | 99.31 | 96.06 | 98.71 |
| Fallow-P | 28 | 1366 | 99.20 | 99.17 | 98.15 | 99.54 | 90.32 | 98.24 | 98.31 | 96.32 | 97.94 |
| Fallow-S | 54 | 2624 | 96.26 | 94.03 | 99.05 | 97.81 | 95.38 | 98.84 | 99.69 | 97.14 | 98.28 |
| Stubble | 80 | 3879 | 99.53 | 99.74 | 99.96 | 99.63 | 99.19 | 99.71 | 99.88 | 99.01 | 99.78 |
| Celery | 72 | 3507 | 99.47 | 99.41 | 98.89 | 99.75 | 99.75 | 99.83 | 99.83 | 99.12 | 99.63 |
| Grapes | 226 | 11045 | 86.71 | 72.83 | 85.14 | 93.55 | 90.77 | 90.44 | 85.13 | 97.93 | 98.53 |
| Soil | 125 | 6078 | 98.98 | 98.16 | 99.47 | 99.96 | 99.73 | 99.86 | 98.69 | 99.42 | 99.92 |
| Corn | 66 | 3212 | 93.08 | 89.64 | 91.60 | 96.18 | 97.25 | 98.20 | 97.04 | 98.25 | 98.61 |
| Lettuce-4wk | 22 | 1046 | 95.21 | 94.75 | 92.74 | 96.97 | 99.56 | 91.72 | 99.26 | 96.63 | 97.53 |
| Lettuce-5wk | 39 | 1888 | 99.63 | 99.49 | 99.30 | 99.98 | 97.24 | 99.81 | 98.80 | 98.56 | 98.98 |
| Lettuce-6wk | 19 | 897 | 97.36 | 97.25 | 97.90 | 98.58 | 95.77 | 99.21 | 96.78 | 95.95 | 96.67 |
| Lettuce-7wk | 22 | 1048 | 93.99 | 92.94 | 93.56 | 95.30 | 98.63 | 87.42 | 97.65 | 95.44 | 97.99 |
| Vinyard-U | 146 | 7122 | 68.45 | 62.52 | 70.60 | 94.09 | 85.73 | 71.32 | 87.75 | 96.65 | 98.40 |
| Vinyard-T | 37 | 1770 | 97.43 | 96.94 | 92.69 | 99.64 | 98.89 | 98.12 | 99.59 | 96.37 | 97.27 |
| OA | — | — | 91.56 | 87.41 | 91.38 | 97.22 | 95.13 | 93.36 | 94.58 | 97.91 | 98.86 |
| AA | — | — | 94.92 | 93.15 | 94.52 | 98.10 | 96.71 | 95.75 | 97.35 | 97.52 | 98.60 |
| Kappa | — | — | 90.57 | 85.99 | 90.38 | 96.90 | 94.57 | 92.58 | 93.95 | 97.67 | 98.72 |

the sparsity parameter ρ on the performance of the proposed GFDN method are analyzed. In this analysis, the numbers of training and test samples are selected to the same as those presented in Tables I–III. The number of units per layer are selected from 10 to 200. Fig. 13(a) illustrates the OA values of the proposed GFDN method under different numbers of units per layer on the Indian Pines, University of Pavia, and Salinas images. It can be seen that the OA values of the proposed GFDN method improve as the number of units per layer increases. However, when the number of units per layer is more than 100, the OAs of the proposed method may vary in a very small dynamic range. The reason is that if the the number of units per layer is very large, the weights in the learned layers obtained will be duplicative, which leading to redundant features representations, and thus decreases the classification performance. Furthermore, Fig. 13(b) analyzes the influences of the sparsity parameter ρ to the performance of the proposed GFDN method, in which ρ is varying from 0.001 to 0.1. It can be seen that, when the

sparsity parameter ρ goes from 0.001 to 0.05, the performance of the proposed method generally degrades. The reason behind this phenomenon is that if the sparsity parameter ρ is very small, $\hat{\rho}_j$ will deviate significantly from ρ and thus, resulting in large values of the penalty term. In this paper, ρ is suggested to be set as a value larger than 0.05.

3) *Effect of the Depth of Network and Maximum Epoch of Fine-tuning and Iteration:* The depth of the deep network also has an influence on the performance of the proposed GFDN method. Fig. 14 illustrates the overall classification accuracies of the proposed method under different numbers of network layers. It can be seen that, the OA values of the proposed GFDN method first increase and then decrease as the numbers of network layers increase. The main reason is that if the number of the network layers is very large, the whole deep network will have much more parameters, so that a larger number of training samples are required to train the deep network accurately. However, although virtual samples can be constructed, the training

TABLE IV
NUMBER OF TRAINING AND TEST SAMPLES OF THE HOUSTON IMAGE AND CLASSIFICATION ACCURACIES (IN %) FOR THE SVM [8], SRC [12], SAE [30], EMAP [14], JSRC [12], MFL [47], EPF [16], GFDN*, AND GFDN METHODS

| Class | Training | Test | SVM | SRC | SAE | EMAP | JSRC | MFL | EPF | GFDN* | GFDN |
|-----------------|----------|------|-------|--------------|-------|--------------|--------------|------------|------------|------------|--------------|
| Health grass | 126 | 1125 | 98.34 | 94.52 | 95.78 | 97.59 | 99.42 | 95.89 | 97.24 | 98.44 | 98.28 |
| Stressed grass | 126 | 1128 | 98.06 | 97.95 | 96.47 | 99.02 | 97.75 | 97.93 | 98.07 | 96.61 | 98.14 |
| Synthetic grass | 70 | 627 | 99.78 | 99.71 | 90.86 | 99.90 | 99.15 | 100 | 100 | 100 | 100 |
| Tree | 125 | 1119 | 98.99 | 99.80 | 94.33 | 99.31 | 99.51 | 96.32 | 99.08 | 95.76 | 96.98 |
| Soil | 125 | 1117 | 98.54 | 96.88 | 97.43 | 99.02 | 99.79 | 99.83 | 99.02 | 99.00 | 99.74 |
| Water | 33 | 292 | 96.99 | 99.51 | 66.03 | 96.82 | 97.83 | 98.08 | 100 | 98.05 | 98.42 |
| Residential | 127 | 1141 | 93.72 | 91.54 | 86.01 | 96.77 | 88.72 | 91.63 | 93.88 | 97.42 | 97.88 |
| Commercial | 125 | 1119 | 92.08 | 91.11 | 81.95 | 93.53 | 95.65 | 80.01 | 95.05 | 92.66 | 94.76 |
| Road | 126 | 1126 | 87.29 | 77.59 | 84.76 | 94.01 | 94.89 | 82.62 | 95.00 | 94.89 | 97.39 |
| Highway | 123 | 1104 | 93.14 | 82.94 | 85.30 | 98.58 | 97.63 | 97.84 | 96.73 | 98.44 | 99.66 |
| Railway | 124 | 1111 | 91.51 | 80.72 | 82.65 | 97.33 | 93.74 | 92.86 | 97.96 | 98.55 | 99.35 |
| Parking lot1 | 124 | 1109 | 91.48 | 75.36 | 86.45 | 94.75 | 96.18 | 85.48 | 86.12 | 95.42 | 97.18 |
| Parking lot2 | 47 | 422 | 50.83 | 72.80 | 79.41 | 78.53 | 59.32 | 77.11 | 90.00 | 77.8 | 92.13 |
| Tennis court | 43 | 385 | 98.26 | 96.71 | 84.00 | 99.79 | 97.01 | 98.31 | 98.34 | 99.66 | 99.97 |
| Running track | 66 | 594 | 98.43 | 99.30 | 98.08 | 99.78 | 99.94 | 98.99 | 99.85 | 99.97 | 100 |
| OA | — | — | 93.57 | 89.77 | 88.65 | 96.75 | 95.51 | 92.56 | 96.12 | 96.85 | 98.01 |
| AA | — | — | 92.50 | 90.43 | 87.30 | 96.31 | 94.44 | 92.86 | 96.42 | 96.84 | 97.99 |
| Kappa | — | — | 93.03 | 88.94 | 87.71 | 96.48 | 95.14 | 91.95 | 95.80 | 96.59 | 97.85 |

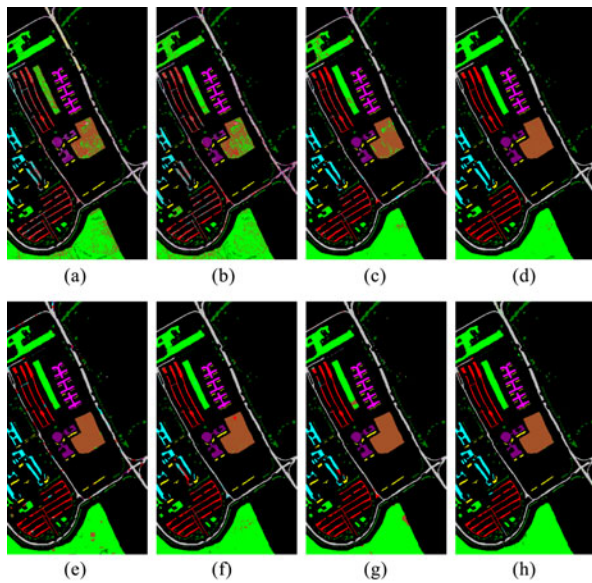


Fig. 10. Classification results (OA in %) of the University of Pavia image obtained by the (a) SVM [8], (b) SRC [12], (c) SAE [30], (d) EMAP [14], (e) JSRC [12], (f) MFL [47], (g) EPF [16], and (h) GFDN methods. (a) OA = 82.33, (b) OA = 77.67, (c) OA = 92.96, (d) OA = 98.18, (e) OA = 94.39, (f) OA = 97.65, (g) OA = 97.94, (h) OA = 98.67.

samples are still very limited if the numbers of network layers increase.

Besides the number of layers, the effect of maximum epoch is also analyzed on the Indian Pines dataset (see Table VI). When the training epoch is very small, the deep network can't be adequately trained and thus, leading to the decrease of classification accuracy. Although large epoch of fine-tuning and iterations can achieve relatively good results, the computational cost is very high. In this paper, the maximum epoch is suggested to be set as 400.

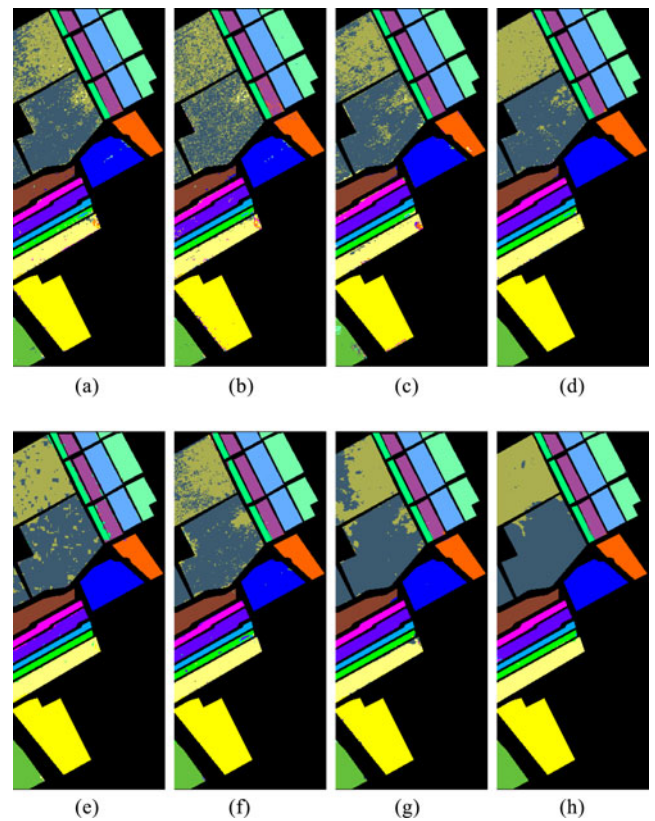


Fig. 11. Classification results (OA in %) of the Salinas image obtained by the (a) SVM [8], (b) SRC [12], (c) SAE [30], (d) EMAP [14], (e) JSRC [12], (f) MFL [47], (g) EPF [16], and (h) GFDN methods. (a) OA = 91.69, (b) OA = 87.76, (c) OA = 91.52, (d) OA = 97.15, (e) OA = 95.49, (f) OA = 93.46, (g) OA = 94.49, (h) OA = 98.90.

D. Effect of Feature Extraction Methods

In this section, the effect of each feature component to the performance of the proposed method is first evaluated in

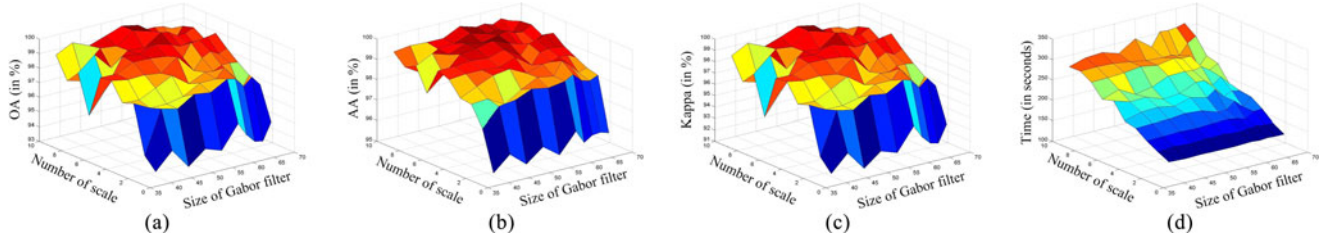


Fig. 12. Experimental results for the University of Pavia image. From left to right record the (a) overall classification accuracy, (b) average classification accuracy, (c) Kappa coefficient of agreement, and (d) computing time of the proposed method with respect to different numbers of scale U and different sizes d of Gabor filter.

TABLE V
RUM TIME (IN SECONDS) FOR EACH STEP OF THE PROPOSED GFDN METHOD
ON THE INDIAN PINES, UNIVERSITY OF PAVIA, AND SALINAS IMAGES

| Step | Indian Pines | University of Pavia | Salinas |
|--------------------------|--------------|---------------------|---------|
| Gabor feature creation | 1.89 | 10.60 | 6.08 |
| First layer training | 92.03 | 97.95 | 99.26 |
| Second layer training | 33.27 | 49.94 | 35.71 |
| Deep network fine-tuning | 61.34 | 58.87 | 64.14 |
| Softmax classification | 0.09 | 0.34 | 0.49 |

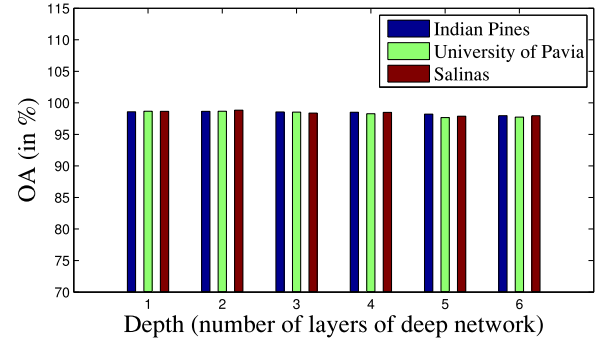


Fig. 14. Effect of depth of network on the Indian Pines, University of Pavia, and Salinas Images.

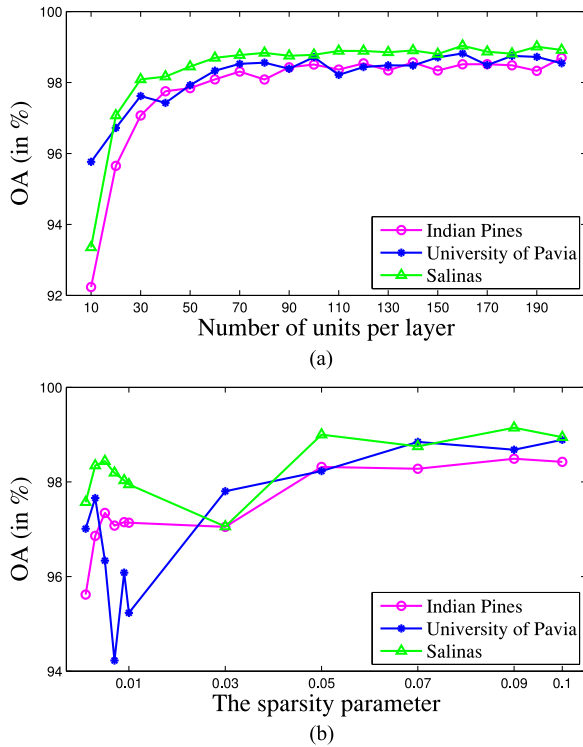


Fig. 13. Effect of the number of units per layer and the sparsity parameter when applying the proposed GFDN method to the three test images. (a) Effect of the number of units per layer. (b) Effect of the sparsity parameter.

Table VII. As shown in this table, the proposed method with the stacked spectral and Gabor features shows the best performance in terms of classification accuracies. It demonstrates that both the Gabor and spectral features make important roles

in improving the classification performance of the proposed method.

Then, in order to analysis the influences of different feature extraction methods on the performance of the proposed GFDN method, three widely used feature extraction methods of APs [14], EPs [35] and LBP [37] are used comparison. The experiment is performed on the Indian Pines, University of Pavia, and Salinas datasets. Table VIII reports the classification accuracies of the compared feature extraction methods. Among these methods, the EPs are evaluated using the EPs of the Indian Pines and University of Pavia images are provided by Li *et al.* [35]. For the APs, relevant parameters are the same as the EMAP method mentioned above. The LBP method is implemented using the code provided by Aptoula *et al.* [37], in which the default parameter setting is adopted. It can be seen that, the Gabor filtering method performs the best of all for the Indian Pines and Salinas datasets. Although the accuracy of EPs is slightly higher than the proposed method on the University of Pavia dataset, it actually performs much worse on the Indian Pines dataset.

E. Effect of Different Number of Principle Components

In this section, the influence of the number of principle components on the performance of the proposed method is evaluated in Fig. 15. It can be seen that, when the number of principle components is only one or two, the performance of the proposed method decreases significantly. However, as the number of principle components increases from 3 to 6, the classification accuracy only increases slightly for the Salinas dataset and

TABLE VI
EFFECT OF MAXIMUM EPOCH OF FINE-TUNING AND ITERATION ON THE INDIAN PINES IMAGE

| Metrics index | Maximum epoch of fine-tuning and iteration | | | | | | | | | | |
|---------------|--|-------|-------|--------|--------|--------|--------|--------|--------|--------|--------|
| | 50 | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | 1000 |
| OA (%) | 96.74 | 97.65 | 97.70 | 97.95 | 98.32 | 98.39 | 98.34 | 98.32 | 98.43 | 98.57 | 98.33 |
| Time (s) | 16.70 | 33.74 | 84.67 | 135.19 | 190.56 | 239.76 | 316.62 | 350.47 | 361.44 | 395.46 | 456.48 |

TABLE VII
CLASSIFICATION ACCURACIES (IN %) OBTAINED BY THE PROPOSED METHOD WITH THE ORIGINAL SPECTRAL FEATURES, GABOR FEATURES AND STACKED FEATURES

| Metrics index | Indian Pines | | | University of Pavia | | | Salinas | | |
|---------------|--------------|-------|--------------|---------------------|-------|--------------|----------|-------|--------------|
| | Spectral | Gabor | GFDN | Spectral | Gabor | GFDN | Spectral | Gabor | GFDN |
| OA | 73.06 | 96.04 | 98.29 | 81.17 | 94.70 | 98.51 | 90.76 | 95.62 | 98.86 |
| AA | 64.64 | 93.25 | 97.49 | 84.56 | 95.77 | 99.14 | 94.05 | 94.28 | 98.60 |
| Kappa | 69.15 | 95.48 | 98.06 | 75.87 | 93.02 | 98.01 | 89.72 | 95.13 | 98.72 |

The experiment is performed on the Indian Pines, Salinas and University of Pavia Images, respectively.

TABLE VIII
CLASSIFICATION ACCURACIES (IN %) OF THE INDIAN PINES, SALINAS AND UNIVERSITY OF PAVIA IMAGES WITH THE APs [14], EPs [35], LBP [37] AND GABOR [41] METHODS FOR PREPROCESSING

| Metrics index | Indian Pines | | | | University of Pavia | | | | Salinas | | |
|---------------|--------------|-------|-------|--------------|---------------------|--------------|-------|-------|---------|-------|--------------|
| | APs | EPs | LBP | Gabor | APs | EPs | LBP | Gabor | APs | LBP | Gabor |
| OA | 93.78 | 93.11 | 94.03 | 98.29 | 97.98 | 98.94 | 91.48 | 98.51 | 96.01 | 93.85 | 98.86 |
| AA | 90.41 | 89.11 | 86.63 | 97.49 | 98.53 | 99.18 | 92.82 | 99.14 | 97.14 | 92.06 | 98.60 |
| Kappa | 92.89 | 92.13 | 93.19 | 98.06 | 97.31 | 98.58 | 88.96 | 98.01 | 95.56 | 93.13 | 98.72 |

The experiment is performed on the Indian Pines, University of Pavia, and Salinas Images, respectively.

decrease for the University of Pavia and Indian Pines datasets. The main reason is that the first three principal components can preserve most of the edge information of the hyperspectral image, and thus, is already able to well model the spatial information of the hyperspectral image. However, the following principal components may contain more spatial information while bringing more image noise, and thus, may have a negative effect on the classification performance. Furthermore, a small number of principal components could reduce the computing cost of the proposed method, and thus, the number of principal components is fixed as 3 in this paper.

F. Effect of Different Numbers of Training Samples

In this section, the influence of the number of training and test samples on the compared methods is analyzed on the Indian Pines, University of Pavia, and Salinas images. For the Indian Pines and Salinas images, different percentages (from 2% to 15% per class for the Indian Pines image and from 0.25% to 3% per class for the Salinas image) of the labeled data are randomly selected as training samples, and then, the corresponding virtual samples are generated. For the University of Pavia image, various numbers (from 10 to 200 pixels for each class) of pixels are randomly selected as real training samples, and then

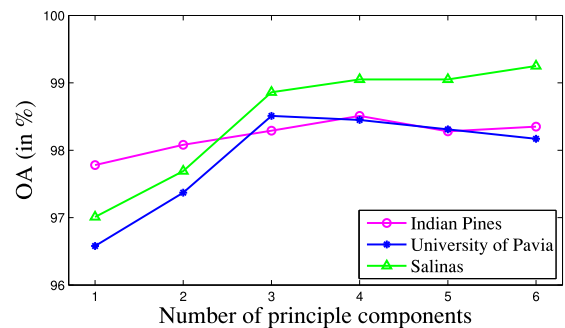


Fig. 15. Influence of the number of principle components on the performance of the proposed method.

corresponding virtual samples are constructed from the selected training samples.

Fig. 16 illustrates the overall classification accuracy (averaged over ten runs) of each method with different numbers of training samples. It can be observed that the performances of all the compared methods will improve as the number of training samples increases. Furthermore, the proposed GFDN method consistently presents excellent performances with different numbers of training and test samples.

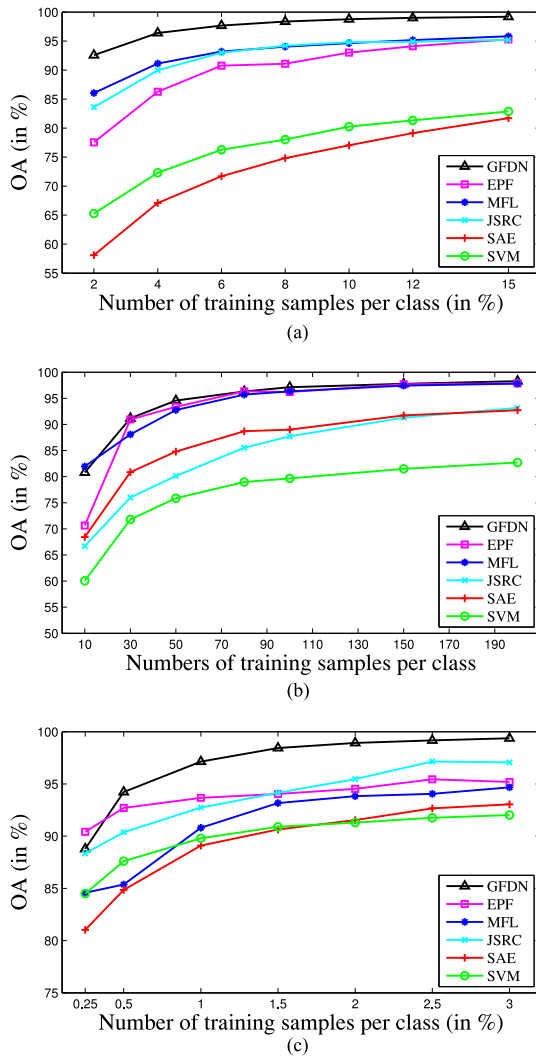


Fig. 16. Influence of the number of training samples per class on the GFDN, EPF, MFL, JSRC, SAE, and SVM for the (a) Indian Pines image, (b) University of Pavia image, and (c) Salinas image.

V. CONCLUSION

This paper presents a GFDN method for HSI classification. Through integrating Gabor filtering and SAE in a combined classification framework, the performance of hyperspectral image classification is able to be improved. Experiments performed on four real HSI images demonstrate the effectiveness of the proposed GFDN, in terms of both visual quality on the classification map and quantitative metrics. The major disadvantage of the proposed method is that it requires a number of correctly labeled samples to train the network. In the future, whether the network could be trained using a massive number of training samples which contain both accurate and inaccurate samples could be further researched.

REFERENCES

[1] J. R. Kaufman, M. T. Eismann, and M. Celenk, "Assessment of spatial-spectral feature-level fusion for hyperspectral target detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2534–2544, Jun. 2015.

[2] X. Kang, X. Zhang, S. Li, K. Li, J. Li, and J. A. Benediktsson, "Hyperspectral anomaly detection with attribute and edge-preserving filters," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 10, pp. 5600–5611, Oct. 2017.

[3] X. Kang, Y. Huang, S. Li, H. Lin, and J. A. Benediktsson, "Extended random walker for shadow detection in very high resolution remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, to be published, doi: [10.1109/TGRS.2017.2755773](https://doi.org/10.1109/TGRS.2017.2755773).

[4] J. M. Bioucas-Dias *et al.*, and "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 354–379, Apr. 2012.

[5] 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.

[6] X. Kang, S. Li, and J. A. Benediktsson, "Feature extraction of hyperspectral images with image fusion and recursive filtering," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 6, pp. 3742–3752, Jun. 2014.

[7] X. Kang, X. Xiang, S. Li, and J. A. Benediktsson, "PCA-based edge-preserving features for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, to be published, doi: [10.1109/TGRS.2017.2743102](https://doi.org/10.1109/TGRS.2017.2743102).

[8] 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.

[9] M. Dalponte, H. O. rka, T. Gobakken, D. Gianelle, and E. Nsset, "Tree species classification in boreal forests with hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 5, pp. 2632–2645, May 2013.

[10] 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.

[11] F. Ratle, G. Camps-Valls, and J. Weston, "Semisupervised neural networks for efficient hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 5, pp. 2271–2282, May 2010.

[12] 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.

[13] A. Castrodad, Z. Xing, J. B. Greer, E. Bosch, L. Carin, and G. Sapiro, "Learning discriminative sparse representations for modeling, source separation, and mapping of hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4263–4281, Nov. 2011.

[14] M. D. Mura, J. A. Benediktsson, B. Waske, and L. Bruzzone, "Extended profiles with morphological attribute filters for the analysis of hyperspectral data," *Int. J. Remote Sens.*, vol. 31, no. 22, pp. 5975–5991, Dec. 2010.

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

[16] 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.

[17] 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.

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

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

[20] 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.

[21] X. Kang, S. Li, L. Fang, and J. A. Benediktsson, "Intrinsic image decomposition for feature extraction of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 4, pp. 2241–2253, Apr. 2015.

[22] B. Sun, X. Kang, S. Li, and J. A. Benediktsson, "Random-walker-based collaborative learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 1, pp. 212–222, Jan. 2017.

[23] X. Kang, S. Li, L. Fang, M. Li, and J. A. Benediktsson, "Extended random walker-based classification of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 1, pp. 144–153, Jan. 2015.

- [24] L. Zhang, L. Zhang, D. Tao, and X. Huang, "Tensor discriminative locality alignment for hyperspectral image spectral-spatial feature extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 1, pp. 242–256, Jan. 2013.
- [25] R. Ji, Y. Gao, R. Hong, Q. Liu, D. Tao, and X. Li, "Spectral-spatial constraint hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 3, pp. 1811–1824, Mar. 2014.
- [26] R. Mendez-Rial and J. Martin-Herrero, "Efficiency of semi-implicit schemes for anisotropic diffusion in the hypercube," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2389–2398, May 2012.
- [27] G. Hinton *et al.* "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.
- [28] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1891–1898.
- [29] M. Chen, L. Zhang, and J. P. Allebach, "Learning deep features for image emotion classification," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2015, pp. 4491–4495.
- [30] 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.
- [31] 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.
- [32] 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.
- [33] H. Liang and Q. Li, "Hyperspectral imagery classification using sparse representations of convolutional neural network features," *Remote Sens.*, vol. 8, no. 2, p. 99, Jan. 2016.
- [34] Y. Li, H. Zhang, and Q. Shen, "Spectral-spatial classification of hyperspectral imagery with 3D convolutional neural network," *Remote Sens.*, vol. 9, no. 1, p. 67, Jan. 2017.
- [35] P. Ghamisi, R. Souza, J. A. Benediktsson, X. X. Zhu, L. Rittner, and R. A. Lotufo, "Extinction profiles for the classification of remote sensing data," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 5631–5645, Oct. 2016.
- [36] P. Ghamisi, B. Hfle, and X. X. Zhu, "Hyperspectral and lidar data fusion using extinction profiles and deep convolutional neural network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 6, pp. 3011–3024, Jun. 2017.
- [37] 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.
- [38] E. Aptoula, M. C. Ozdemir, and B. Yanikoglu, "Deep learning with attribute profiles for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 12, pp. 1970–1974, Dec. 2016.
- [39] A. Ng, "Sparse autoencoder," *CS294A Lect. Notes*, vol. 72, pp. 1–19, 2011.
- [40] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, "A limited memory algorithm for bound constrained optimization," *SIAM J. Sci. Comput.*, vol. 16, no. 5, pp. 1190–1208, Sep. 1995.
- [41] L. L. Shen, L. Bai, and M. Fairhurst, "Gabor wavelets and general discriminant analysis for face identification and verification," *Image Vis. Comput.*, vol. 25, no. 5, pp. 553–563, May 2007.
- [42] R. Alterson and M. Spetsakis, "Object recognition with adaptive gabor features," *Image Vis. Comput.*, vol. 22, no. 12, pp. 1007–1014, Oct. 2004.
- [43] S. E. Grigorescu, N. Petkov, and P. Kruizinga, "Comparison of texture features based on gabor filters," *IEEE Trans. Image Process.*, vol. 11, no. 10, pp. 1160–1167, Oct. 2002.
- [44] C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition," *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 467–476, Apr. 2002.
- [45] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Comput.*, vol. 15, no. 6, pp. 1373–1396, Mar. 2006.
- [46] J. Wang, *Pearson Correlation Coefficient*. New York, NY, USA: Springer, 2013, pp. 1671–1671.
- [47] J. Li *et al.*, "Multiple feature learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 3, pp. 1592–1606, Mar. 2015.
- [48] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, pp. 27:1–27:27, 2011.



Xudong Kang (S'13–M'15–SM'17) received the B.Sc degree from Northeast University, Shenyang, China, in 2007, and the Ph.D. degree from Hunan University, Changsha, China, in 2015, both in electrical engineering.

In 2015, he joined the College of electrical Engineering, Hunan University, Changsha, China. His research interests include hyperspectral feature extraction, image classification, image fusion, and anomaly detection.

Dr. Kang received the Second Prize in the Student Paper Competition in IGARSS 2014. In IGARSS 2017, he was selected as the Best Reviewer for IEEE Geoscience and Remote Sensing Letters in 2016.



Chengchao Li received the B.Sc degree in electrical engineering from Hubei Engineering University, Xiaogan, China, in 2015. He is currently working toward the M.S. Degree in electrical engineering at Hunan University, Changsha, China.

His research interest include hyperspectral image classification.



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, all in electrical engineering.

In 2001, he joined the College of Electrical and Information Engineering, Hunan University. From May 2001 to October 2001, he was a Research Associate in the Department of Computer Science, Hong Kong University of Science and Technology, Kowloon, Hong Kong. From November 2002 to November 2003, he was a Postdoctoral Fellow in the Royal Holloway College, University of London, Surrey, U.K. From April 2005 to June 2005, he was a Visiting Professor in the Department of Computer Science, Hong Kong University of Science and Technology. In 2013, he was granted the National Science Fund for Distinguished Young Scholars in China. He is currently a Full Professor in the College of Electrical and Information Engineering, Hunan University. He is also a Chang-Jiang Scholar Professor appointed by the Ministry of Education of China. He has authored/coauthored more than 180 refereed papers. His research interests include compressive sensing, sparse representation, image processing, and pattern recognition.

Dr. Li is an Associate Editors of the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT, Members of the Editorial Board of the Information Fusion, Sensing and Imaging. He received two 2nd-Grade National Awards at the Science and Technology Progress of China, in 2004 and 2006.



Hui Lin received the B.S. degree in electrical engineering and M.S. degree in management engineering from Hunan University, Changsha, China, in 1985 and 1999, respectively. He is currently an Associate Professor in the College of Electrical and Information Engineering, Hunan University. He has authored or co-authored more than 20 refereed papers. His research interests include system engineering and pattern recognition.