Helmholtz-Zentrum Dresden-Rossendorf (HZDR)

HELMHOLTZ ZENTRUM
DRESDEN ROSSENDORF

Multiscale Densely-Connected Fusion Networks for Hyperspectral Images Classification

Xie, J.; He, N.; Fang, L.; Ghamisi, P.;

Originally published:

February 2020

IEEE Transactions on Circuits and Systems for Video Technology 31(2021)1, 246-259

DOI: https://doi.org/10.1109/TCSVT.2020.2975566

Perma-Link to Publication Repository of HZDR:

https://www.hzdr.de/publications/Publ-31958

Release of the secondary publication
on the basis of the German Copyright Law § 38 Section 4.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.2975566, IEEE Transactions on Circuits and Systems for Video Technology

Multiscale Densely-Connected Fusion Networks for Hyperspectral Images Classification

Jie Xie, *Student Member, IEEE,* Nanjun He, *Student Member, IEEE,* Leyuan Fang, *Senior Member, IEEE,* and Pedram Ghamisi, *Senior Member, IEEE*

***Abstract*—Convolutional neural network (CNN) has demonstrated to be a powerful tool for hyperspectral images (HSIs) classification. Previous CNN-based HSI classification methods only adopt the fixed-size patches to train the CNN model, and such single scale patches may not reflect the complex spatial structural information in the HSIs. In addition, although different layers of CNN can extract features of multiple scales, the traditional CNN model can only utilize features from the highest level for the classification task. These features, however, do not fully consider the strong complementary yet correlated information among different layers. To address these issues, in this paper, a multiscale densely-connected convolutional network (MS-DenseNet) framework is proposed to sufficiently exploit multiple scales information for the HSIs classification. Specifically, for each pixel, the MS-DenseNet, first, extracts its surrounding patches of multiple scales. These patches can separately constitute multiple scale training and testing samples. Within each specific scale sample, instead of using the forward convolutional layers, the MS-DenseNet adopts the dense blocks, which can connect each layer to other layers in a feed-forward fashion and thus can exploit the information among different layers for training and testing. Furthermore, since high correlations exist in patches of different scales, the MS-DenseNet introduces several dense blocks to fuse the multiscale information among different layers for the final HSI classification. Experimental results on several real HSIs demonstrate the superiority of the proposed MS-DenseNet over single scale-based CNN classification model and several well-known classification methods.**

***Index Terms*—Hyperspectral images (HSIs) classification, multiscale, convolutional neural networks, densebolck.**

I. Introduction

HYPERSPECTRAL images (HSIs) consist of hundreds of observation bands among the electromagnetic spectrum with high spectral resolution [1] and have attracted considerable attention in the remote sensing community. With such rich spectral information, HSIs can be effectively utilized to differentiate the categories of different land covers. In addition, HSIs have also played an important role in many practical applications, such as environmental monitoring [2], object detection [3], military defense [4], and among others [5], [6].

The objective of HSIs classification is to assign each spectral pixel to one specific class. In the last decades, a vast of methods have been developed to address the HSIs classification task. Among them, the support vector machines (SVMs) [7] are a powerful classifier, which can learn an optimal hyperplane to classify the HSIs. As another widely used HSIs classifier, multinomial logistic regression [8] uses the logistic function to compute the posterior probability for the classification. The random forest [9] develops several trees from randomly sampled subspaces and then combines the outputs with a majority voting, which has also been applied for HSI classification [10]. In general, although the spectral information can be effectively utilized in the aforementioned classification methods, the spatial contexts of the HSIs are not considered and thus their classification maps usually appear noisy. To enhance the classification performance, recent approaches have attempted to incorporate the spatial information of the HSI into the classification task (e.g., composite kernel [11], superpixel-based sparse representation [12], [13], and random fields [14], [15]). In addition, other kinds of classification methods focus on the design of effective spatial-spectral feature extraction techniques (e.g., mathematical morphology [16], loopy belief propagation [17] and Random Walker [18]). Although the above-mentioned methods can provide promising classification results, they adopt the "shallow" models, and thus has limited ability to handle the highly nonlinear HSI data.

All of the above methods are based on hand-crafted features, which rely heavily on the experience of experts. Deep learning can automatically achieve effective feature extraction by means of hierarchical layers, thereby extracting high-level and abstract features for the input nonlinear data. It has achieved great success in many applications, including image classification [19], object detection [20], and super-resolution [21]. Recently, deep learning models have also been extended to HSIs classification [22]. In [23], Chen *et al.* used a deep stacked auto-encoder (SAE) to extract features from the spectral domain of HSIs for the classification task. In [24], Tao *et al.* introduced a modified auto-encoder model called as multiscale sparse SAE to construct features from unlabeled data in HSIs. The learned features were input into the SVMs for classification. In [25], Chen *et al.* further adopted the singular restricted Boltzmann machine (RBM) and a multiple-layer deep network to learn the spectral-based features from HSIs for classification. The above-mentioned networks (e.g., SAE and RBM) are equipped with fully connected layers, and thus a lot of parameters are required to be estimated in those networks, causing very high computational cost. In addition, the above methods only adopt the 1-D spectral vector to train

This paper was supported by the National Natural Science Foundation of China under Grant No.61922029, the National Natural Science Foundation under Grant No. 61771192, and the Science and Technology Planning Project of Hunan Province NO. 2019RS2016.

J. Xie, N. He, and L. Fang are with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China (e-mail: xj_xj@hnu.edu.cn; henanjun@hnu.edu.cn; fangleyuan@gmail.com).

P. Ghamisi is with Helmholtz-Zentrum Dresden-Rossendorf, Helmholtz Institute Freiberg for Resource Technology, Exploration Chemnitzer Str. 40, D-09599 Freiberg, Germany (e-mail: p.ghamisi@gmail.com).

Copyright ©2020 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending an email to pubs-permissions@ieee.org.
1051-8215 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Helmholtz-Zentrum Dresden-Rossendorf. Downloaded on December 22,2020 at 07:52:34 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.2975566, IEEE Transactions on Circuits and Systems for Video Technology

networks and thus do not utilize the 2-D spatial structures in the HSI.

Chen *et al.* first introduces the convolutional neural network (CNN) into the HSIs classification in [26]. In this method, first, the principal component analysis (PCA) is used to reduce the spectral dimension of the HSI and the fixed-size spatial patches are extracted as the training samples. Then, several convolutional layers are adopted to automatically extract spatial feature maps from the training samples and different layers can extract the features from different scales. The better experimental results verify that the spatial information contributes to the performance of CNN-based HSIs classification methods. In addition, the CNN uses the local connections and shared weights to greatly reduce the number of parameters, which can speed the training and testing process. A self-improving CNN model is proposed in [27] to adaptively select several informative spectral bands to train the CNN model. A CNN-based pixel-pairs model is proposed in [28] to combine the center pixel and each of the surrounding pixels in a patch to create pixel-pairs, which is expected to extract more spatial information in the CNN model. Therefore, the CNN can jointly exploit the spatial-spectral information within the patches, which can provide a very high classification accuracy. However, the above-mentioned CNN models only adopt the fixed-size patches and thus have the limited capacity to reflect the complex structures in HSIs. Furthermore, such CNN networks simply pass the feature maps from one convolutional layer to the next layer in a single-pass forward way. Different layers can extract features of different scales, and thus, this multiple scale information within the CNN model is not effectively utilized to train the network.

To address the aforementioned issues, a novel framework termed as multiscale densely-connected convolutional network (MS-DenseNet) is proposed to integrate multiscale information within and among the networks for HSIs classification. Firstly, for each pixel, different sizes of surrounding patches are extracted to construct the corresponding training and testing samples, which can reflect the HSI structures in different scales. Within samples of one specific scale, instead of using the forward convolutional layers in CNN, the dense block is introduced to connect each layer to other layers in a feed-forward fashion and thus can exploit the multiple scale information among different layers for training and testing. In addition, since the patches among different scales contain high correlated and complementary information, the MS-DenseNet progressively combines the multiple-scale information among different networks for the final HSI classification.

The main contributions of this study are summarized below.

1) To exploit multiscale information in both of the hyperspectral image and feature maps of different layers in the CNN model, the MS-FC-DenseNet is proposed to fuse the information of multiscale samples after the last pooling layer.

2) The MS-DenseNet is designed to fuse feature maps of different scale samples after the first dense block, and this fusion strategy enables the MS-DenseNet to have fewer parameters and better classification performance than the MS-FC-DenseNet.

3) A series of the comprehensive analysis and comparison of the proposed network design have been implemented on five real publicly hyperspectral images in term of the classification performance (i.e., overall accuracy (OA), average accuracy (AA), and Kappa coefficient (Kappa)).

The rest of this paper is organized as follows. Section II reviews the traditional CNN model and CNN based HSIs classification. Section III introduces the MS-DenseNet for HSIs classification. Section IV shows experimental results on several real hyperspectral data sets. Section V concludes this paper and suggests some future works.

II. Related Works

A. CNN Model

Recently, the CNN has been widely applied to several images processing applications, especially for the natural images classification [29], [30]. A CNN model can automatically extract the effective features from the input images by adopting a series of transforming layers. The commonly used layers include: 1) convolutional layers; 2) pooling layers; 3) fully connected layers; and 4) soft max classification layer. Details of the above layers are described as follows.

1) Convolutional Layer: Convolutional layer is crucial to extract features from input data, which consists of a number of convolutional kernels (also called filters). The convolutional layer utilizes the kernels to convolve the input images (or feature maps), which can create different feature maps. Specifically, let $\boldsymbol{X_i} = \{x_{1,1,1,i}, \ldots, x_{m,h,c,i}, \ldots, x_{M,H,C,i}\}$ be the input data in the i-th convolutional layer and its size is $M \times H \times C$, where M and H are the height and the width of the input data, respectively, and C represents the number of the channel in the feature maps or input images. Assume the convolutional layer has J kernels and $\boldsymbol{W_{j,i}} = \{w_{1,1,1,j,i}, \ldots, w_{k,k,c,j,i}, \ldots, w_{K,K,C,j,i}\}$ represents the j-th kernel ($1 \leqslant j \leqslant J, 1 \leqslant K \leqslant M$ and $1 \leqslant K \leqslant H$) in the i-th layer. The output of the i-th convolutional layer $\boldsymbol{Y_i} = \{\boldsymbol{Y_{1,i}}, \ldots, \boldsymbol{Y_{j,i}}, \ldots, \boldsymbol{Y_{J,i}}\}$ is obtained by:

$$\boldsymbol{Y_{j,i}} = \boldsymbol{X_i} \otimes \boldsymbol{W_{j,i}}, \tag{1}$$

where $\otimes$ represents convolution operation and $\boldsymbol{Y_{j,i}}$ is the j-th feature map in the i-th layer. The above mapping can also be defined as $\boldsymbol{Y_i} = \boldsymbol{f}(\boldsymbol{X_i})$. As the number of convolutional layer increases, the receptive field will become larger, which means that each pixel in the later feature maps can cover more spatial information of wider region. In other words, for different convolutional layers, the corresponding extracted feature maps can reflect spatial structures of different scales (from fine to coarse scales). However, in the convolutional training and testing process, the information only flows between two adjacent layers.

2) Pooling Layer: In the convolutional layer, multiple kernel convolution operations will greatly increase the number of feature maps, thus creating very high computational burden for the following steps. Therefore, a pooling layer is usually adopted after the convolutional layer to reduce the spatial size of the feature maps and the extracted features become more abstract. Specifically, the pooling layer fuses the nearby spatial information (within a window of size $b \times b$) in the feature maps by usually adopting the max or average operations.

1051-8215 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Helmholtz-Zentrum Dresden-Rossendorf. Downloaded on December 22,2020 at 07:52:34 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.2975566, IEEE Transactions on Circuits and Systems for Video Technology

B. Fully Connected Layer

After several convolutional and pooling layers, fully connected (FC) layers are adopted which have full connections to all features in the previous layer and create a vector of size $K \times 1$. Finally, such vector is fed into the softmax layer, which can determine the probability of each image belonging to each class and the probability is denoted as [31]:

$$P_k = \frac{e^{z_{k,i}}}{\sum_{n=1}^{K} e^{z_{k,i}}}, k = 1, \ldots, K, \quad (2)$$

where K is the number of classes and $z_{k,i}$ is the k-th value of the input $\boldsymbol{Z} = \{z_{1,i}, \ldots, z_{k,i}, \ldots, z_{K,i}\}$. Then, the class with the maximal probability is used as the estimated label θ_n for each image. Based on the estimated labels, the loss function L_f can be obtained via a combination of logistic loss and an additional weight decay term for regularization,

$$L_f = -\sum_{Z} g_k(z) log(P_k) + \lambda(\left\|\boldsymbol{W}^{(\cdot)}\right\|_{\boldsymbol{F}}^{2} + \left\|\boldsymbol{S}^{(\cdot)}\right\|_{\boldsymbol{F}}^{2}), \quad (3)$$

where $g_k(z)$ is a vector with one for the true label and zero entries for the others, the $\boldsymbol{W}^{(\cdot)}$ represents the set of all parameters in the convolutional layers, the $\boldsymbol{S}^{(\cdot)}$ represents the set of all parameters in the fully connected layers, and λ represents the weight decay coefficient of the CNN model.

To minimize the loss function, the backward propagation algorithm is used to optimize the aforementioned parameters ($\boldsymbol{W}^{(\cdot)}$ and $\boldsymbol{S}^{(\cdot)}$). Specifically, it propagates the predicted error L_f from the last layer to the first layer and modifies parameters according to the propagated error at each layer. In general, the stochastic gradient descent (SGD) algorithm is applied to achieve this goal. Based on the gradient of parameters, the loss function can be updated to be optimal.

C. CNN based HSIs Classification

Based on its powerful ability to extract representative features, CNN has recently been applied for the HSIs classification [32], [33]. Specifically, let $\boldsymbol{X}$ be an HSI with the size of $M \times H \times C$, where C is the number of spectral bands. For each training pixel, we extract its fixed neighboring pixels within a $b \times b$ block to construct the corresponding patch x_{Train}^{i} of size $b \times b \times C$. Then, these labeled training samples are inputted to CNN. As mentioned in the above subsection, once the loss function reaches convergence, the CNN model, which includes optimal parameters, will be obtained. For the testing, we also extract the fixed patch x_{Test}^{i} of size $b \times b \times C$ for each test pixel, and then the trained CNN model is applied on the test patches to determine the class label of the corresponding test pixels.

III. The Proposed MS-DenseNet Method for HSIs Classification

In the previous CNN-based HSIs classification methods described in Section II B, the extracted fixed-size patches are used as the training samples. In general, single scale fixed-size patches cannot well reflect the complex structures of the HSI. This is due to the fact that large homogeneous regions require large patches, whereas detailed structural regions need small patches. In addition, as the number of convolutional layers increases, different layers can extract features from fine (more detail) to coarse (more abstract) scales. However, traditional CNN networks simply pass the feature maps from one convolutional layer to the next layer in a single forward manner, and thus the multiple scale information in the CNN model is not effectively exploited to train the network.

To address the above issues, the MS-DenseNet method is proposed in this section to sufficiently utilize multiscale information for the HSIs classification. Firstly, we discuss how to create the multiscale samples used for MS-DenseNet training and testing. Then, we introduce the dense blocks to connect the multiscale information among different convolutional layers and combine information of multiscale training samples for HSI classification. The flowchart of the proposed MS-DenseNet is illustrated in Fig. 1.

A. The Creation of Multiscale Samples

Since the HSIs contain hundreds of spectral bands, the PCA dimension reduction algorithm is first applied on the spectral dimension of the original HSIs and the first L numbers of principle components are used as the feature image (of size $M \times H \times L$) [34], [35]. For each pixel $\mathbf{y_i}$ in the feature image, we extract its $\boldsymbol{T}$ neighboring patches $\{\boldsymbol{Y_1}, \cdots, \boldsymbol{Y_t}, \cdots, \boldsymbol{Y_T}\}$ of different scales. Then, the extracted patches are constructed as the corresponding multiscale samples $\{\boldsymbol{U_1}, \cdots, \boldsymbol{U_t}, \cdots, \boldsymbol{U_T}\}$, where $\boldsymbol{U_t}$ denotes all the extracted patches from the t-th scale.

B. Multiscale Fusion via Dense block

Since there exists multiscale information among different convolutional layers, we first introduce the dense block to connect each layer to other layers in a feed-forward fashion. Specifically, for the i-th convolutional layer in dense block, it receives the feature maps $\{\boldsymbol{Y_1}, \ldots, \boldsymbol{Y_{i-1}}\}$ from all previous convolutional layers and concatenates them as the input:

$$\boldsymbol{x_i} = Concate[\boldsymbol{Y_1}, \ldots, \boldsymbol{Y_{i-1}}]. \quad (4)$$

where $Concate[\cdot]$ denotes the concatenation operation. The example of one dense block with four convolutional layers are shown in Fig. 2. As described in [36], to facilitate the downsampling process, the whole network will be divided into several dense blocks.

Given training samples of multiple scales, one simple fusion strategy is first to use training samples of one specific scale to train one network with several dense blocks. Then, networks of different scales are fused in the final FC layer. Since the network (MS-FC-DenseNet) in each scale requires several dense block (see Fig. 3), such a fusion strategy will make the whole multiscale fusion network very huge, thus will create high computational cost. Alternatively, we introduce a novel fusion strategy to reduce the number of required dense blocks. Specifically, for training samples of each specific scale, we first utilize one dense block to create the feature maps (see Fig. 1). Then, the feature maps of different scales are fused together (using the concatenation) and used as the input into the next dense block one. In this way, instead of fusing all feature maps in the final FC layer with many dense blocks,

1051-8215 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Helmholtz-Zentrum Dresden-Rossendorf. Downloaded on December 22,2020 at 07:52:34 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.2975566, IEEE Transactions on Circuits and Systems for Video Technology

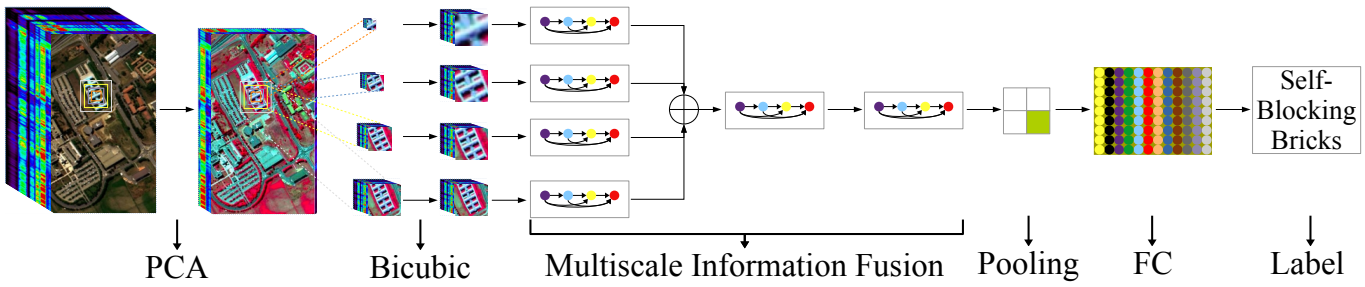

Fig. 1. The proposed framework (MS-DenseNet) to fuse multiscale information for HSIs classification.

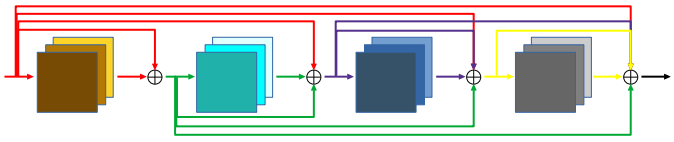

Fig. 2. A four convolutional layer dense block. Each layer takes all feature maps as the input.

we fuse the feature maps after the first dense block layer, which can reduce the number of dense blocks. Note that, since feature maps from different scales are of different sizes, they cannot be directly concatenated together. Therefore, for training and testing samples of different scales, we utilize the Bicubic algorithm to interpolate the samples of different scales to the size of the largest scale samples. After the last dense block step, we further add one pooling layer and fully connected layer to create one feature vector. This vector is then fed into the softmax layer for the classification. After the multiple-scale network is trained, for each testing, we will input its corresponding extracted patches of multiple scales into the trained network and finally will obtain one class label for each center pixel.

IV. Experimental Results

A. Data sets Description

To verify the effectiveness of the proposed MS-DenseNet framework, it is tested on five real hyperspectral data sets: The Washington DC image, the University of Pavia image, the Houston University image, the Indian Pines image and the Salinas image.

The first data set is the Washington DC image, which was collected on the spectral information technology application center of Virginia over the Washington DC Mall. After discarding 19 bands in the spectral range 0.9-1.4 mm, this image contains 191 bands in the 0.4 to 2.4 mm region of the visible and infrared spectrum. Each band is of size 280×307. Fig. 7(a) and (b) show the three-band false color image and the reference samples of the Washington DC data set, that contains six different classes.

The second data set is the University of Pavia image, which was acquired with the reflective optics system imaging spectrometer (ROSIS-03) sensor over the campus of the University of Pavia, Italy. It contains 115 spectral bands ranging from 0.43 to 0.86 μm and has the spatial size of 610×340 pixels with a spatial resolution of 1.3 m. Before the classification, 12 spectral bands were removed due to the existence of high amount of noise. A three-band false color image and the corresponding reference data are illustrated in Fig. 8(a) and (b), that contains nine classes.

The third data set is the Houston University image, which covers the Houston University campus and its neighboring area. This image was distributed in the 2013 GRSS Data Fusion Contest. It contains 144 spectral bands and each band is of size 349×1905. The spatial resolution of this image is 2.5 m per pixel and the spectral coverage ranges from 380 nm to 1050 nm. Heavy shadows contained in the observed data were removed and a sub-region with the size of 349×1300 was retained for classification. Fig. 9(a) and (b) show the color composite of the University of Pavia image and the reference data, which considers fifteen classes of interest.

The fourth data set is the Indian Pines image, which was captured by the airborne visible infrared imaging spectrometer (AVIRIS) sensor over the Indian Pines test site in northwestern Indiana. After removing 20 bands covering the region of water absorption, this image is of size $145 \times 145 \times 200$ with a spatial resolution of 20 m per pixel and its spectral coverage ranges from 0.2 to 2.4 μm. Sixteen classes of interest are considered for the image and Fig. 10(a) and (b) show the false color composite of the Indian Pines image and the corresponding ground truth data.

The fifth data set is the Salinas image, which was acquired by AVIRIS sensor over Salinas Valley, California. This scene image has 224 spectral band and it has a spatial dimension of 512×217 with a spatial resolution of 3.7 m per pixel. As with Indian Pines image, 20 water absorption bands are reduced before the classification. A three-band false color composite of the Salinas image and its ground truth data are seen in Fig. 11(a) and (b), that contains sixteen different classes.

B. Quantitative Metrics

To evaluate the effect of the parameters in the proposed method, the OA is adopted in the following experiments. In addition, AA, and Kappa are also introduced to measure the performances of different methods. The full definitions of the quantitative metrics, i.e., OA, AA, and Kappa are as follows.

Let $\boldsymbol{N_g} = \{N_g^1, \cdots, N_g^k, \cdots, N_g^K\}$ be the number of samples in the ground truth, where the K represents the number of the categories and N_g^k the number of samples in the k-th categories. Assume $\boldsymbol{N_p} = \{N_p^1, \cdots, N_p^k, \cdots, N_p^K\}$, $\boldsymbol{N_c} =$

1051-8215 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Helmholtz-Zentrum Dresden-Rossendorf. Downloaded on December 22,2020 at 07:52:34 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.2975566, IEEE Transactions on Circuits and Systems for Video Technology

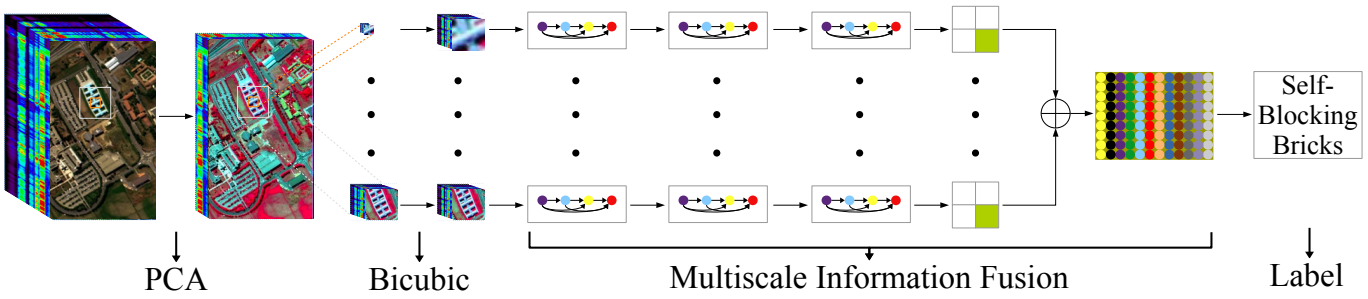

Fig. 3. The structure of the MS-FC-DenseNet.

$\{N_c^1, \cdots, N_c^k, \cdots, N_c^K\}$ and the method predicts N_p^k samples for the k-th categories including N_c^k correct samples. The OA is defined as:

$$\mathrm{OA} = \boldsymbol{N_c}/\boldsymbol{N}, \tag{5}$$

the AA is defined as:

$$\mathrm{AA} = \frac{\sum_{k=1}^{K} N_c^k / N_g^k}{K}, \tag{6}$$

and the Kappa is defined as:

$$\mathrm{Kappa} = \frac{\mathrm{OA} - \frac{\sum_{k=1}^{K} N_p^k \times N_g^k}{\boldsymbol{N_g} \times \boldsymbol{N_g}}}{1 - \frac{\sum_{k=1}^{K} N_p^k \times N_g^k}{\boldsymbol{N_g} \times \boldsymbol{N_g}}}. \tag{7}$$

All results are calculated by averaging the results obtained in ten repeated experiments with ten groups of randomly selected training samples.

C. Parameter Analysis

The parameter setting in the proposed fusion scheme for the HSI classification is given as follows. To construct an effective fusion network for HSIs classification, MS-DenseNet consists of six dense blocks, in which four of them are utilized to extract multiple-scale information from different scale samples (in Fig. 1) and the others are used to extract middle and high level features from the fused feature maps. The number of convolutional layers in the first four dense blocks is set to be 7 and the number of convolutional kernels is set to be 16. The numbers of convolutional layers in the other two dense blocks are set to be 6. To obtain richer information, the number of convolutional kernels is set to be 32 and 64 in the middle and high level dense blocks, respectively. In addition, effects of the number of multiple scale samples and principle component (PC) will be analyzed by evaluating OAs on three data sets.

First, the effects of different single-scale patch sizes to the performance of CNN method are analyzed on three real data sets (i.e. Washington DC image, University of Pavia image and Houston University image). As shown in Fig. 4, the classification results vary when the CNN network considers different sizes of raw patch as input, and larger patch sizes cannot guarantee better results. As described in [37], a small patch can well represent the detailed structural region, whereas large patch can better reflect the large homogeneous region. In this paper, four different patch sizes ($\{\boldsymbol{U_1}, \boldsymbol{U_2}, \boldsymbol{U_3}, \boldsymbol{U_4}\}$) are utilized in our framework, which are set to 3, 9, 15 and 31, respectively. By comparing with different numbers of multiple-scale patches fused in the final model (in Fig. 5), four patches of different sizes are enough to obtain satisfactory classification performance for all three data sets used in the experiments, which also consider the tradeoff between the network complexity and classification accuracy.

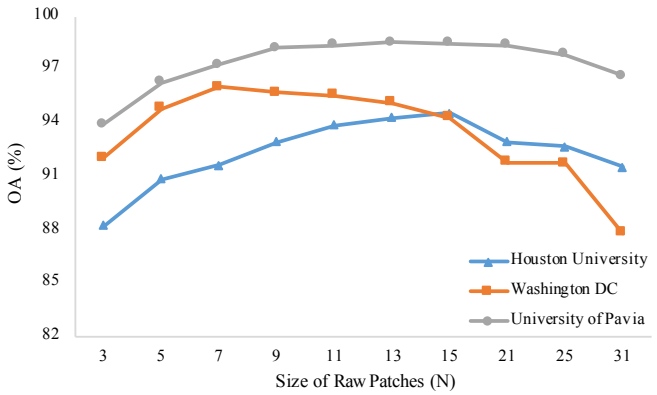

Fig. 4. Effects of different single-scale patch sizes to the performance of CNN framework.

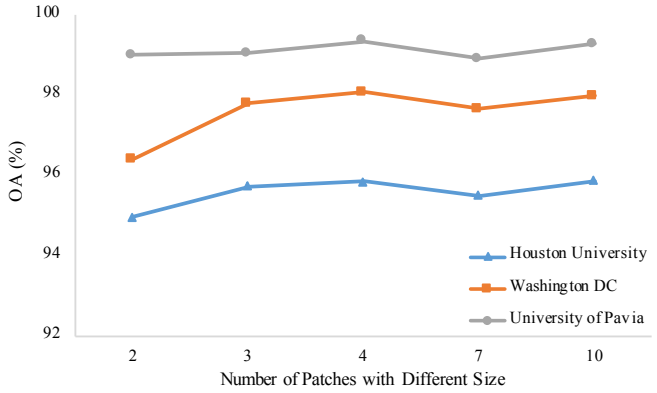

Fig. 5. Effects of different single-scale patch sizes to the performance of CNN framework.

Then, the effect of the number of principal components, i.e., (L = 1, 2, 3, 5, 10), is analyzed. The experiment is performed on three real data sets (i.e. Washington DC image, University of Pavia image and Houston University image) and tested on multiple scale networks. As shown in Fig. 6, a small number of principal components, i.e., (L= 1) and

1051-8215 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Helmholtz-Zentrum Dresden-Rossendorf. Downloaded on December 22,2020 at 07:52:34 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.2975566, IEEE Transactions on Circuits and Systems for Video Technology

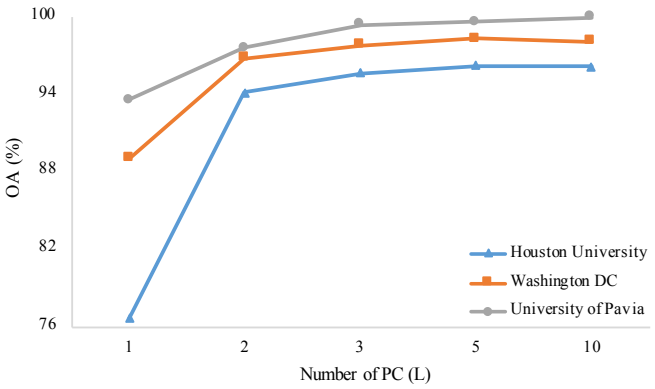

Fig. 6. Effects of different single-scale patch sizes to the performance of CNN framework.

TABLE I
COMPARISON BETWEEN THE MS-FC-DENSENET AND THE MS-DENSENET.

Methods		MS-FC-DenseNet	MS-DenseNet
Washington DC image 300 training samples 9599 testing samples	Parameter amount	9.5 millions	4.5 millions
	Overall accuracy	97.63%	97.74%
	Training time	2523.82 seconds	1403.32 seconds
	Testing time	13.57 seconds	9.52 seconds
University of Pavia image 450 training samples 42326 testing samples	Parameter amount	9.5 millions	4.5 millions
	Overall accuracy	99.10%	99.32%
	Training time	2523.82 seconds	1403.32 seconds
	Testing time	53.12 seconds	36.06 seconds
Houston University image 450 training samples 29608 testing samples	Parameter amount	9.5 millions	4.5 millions
	Overall accuracy	95.52%	95.91%
	Training time	2523.82 seconds	1403.32 seconds
	Testing time	37.67 seconds	25.18 seconds
Salinas image 800 training samples 53329 testing samples	Parameter amount	9.5 millions	4.5 millions
	Overall accuracy	99.41%	99.50%
	Training time	2523.82 seconds	1403.32 seconds
	Testing time	66.77 seconds	44.37 seconds
Indian Pines image 520 training samples 9729 testing samples	Parameter amount	9.5 millions	4.5 millions
	Overall accuracy	95.59%	97.66%
	Training time	2523.82 seconds	1403.32 seconds
	Testing time	13.87 seconds	9.17 seconds

(L= 2), are not able to represent raw data well, whereas a large number of principal components, i.e., (L= 5) and (L= 10), generally show limited improvement (or even decrease) in terms of classification accuracy. The reason is that the first three principal components usually contain most of the effective information [38]. Therefore, the number of principal components L is set as 3 in the following experiments.

Finally, the comparison between the MS-FC-DenseNet and the MS-DenseNet, in terms of accuracy, complexity and efficiency, are performed on five real data sets (i.e. Washington DC image, University of Pavia image, Houston University image, Salinas image and Indian Pines image). As can be observed from Table I, compared with the MS-FC-DenseNet, the MS-DenseNet shows better classification performance on five data sets with fewer parameters. Furthermore, fewer parameters result in less training time and less testing time. It is worth noting that, from Table I, since the same training strategies (including the same batch size, the same number of iterations, and the same input size) are adopted on all images, the parameter amount and the training time on all images of each method (MS-FC-DenseNet or MS-DenseNet) are the same.

D. Comparison of Different Methods

The classification results obtained by the proposed multiple-scale information fusion method are visually and quantitatively compared with several well-known methods, including SVM [39], extended morphological profiles (EMP) [40], pixel-wise sparse representation classification (SRC) [41], superpixel-based classification via multiple kernels (SC-MK) [11], image decomposition for feature extraction (IIDF) [42], PCA-Based edge-preserving features (PCA-E) [43], multiscale superpixel-based sparse representation (MSSRC) [44], correlation coefficient and joint sparse representation (CCJSR) [45], multiscale dense networks (MSDNet) [46], deep feature fusion network (DFFN) [47] and semisupervised and active learning (SOAL) [18]. The SVM and SRC methods do not utilize spatial information for HSIs classification, while the remaining ones are among spectral-spatial classification methods. For the EMP, the spectral-spatial feature is extracted by the morphological method. The SC-MK, IIDF and PCA-E are feature-fusion methods, which extract and fuse multiple types of spectral-spatial features. The MSSRC and CCJSR are the modified versions of the SRC, which make better use of the spectral similarity among different pixels by correlation coefficient and multiscale superpixel, respectively. The MSDNet and DFFN are two CNN-based methods, which fuse the feature maps in the different layers to accelerate the classification performance. The SOAL is a method of semisupervised and active learning based on random walker. Furthermore, two other fusion networks are also adopted for comparison: multiscale convolutional network (MS-Net) and MS-FC-DenseNet (described in section III B). Instead of using the dense block, the MS-Net only adopts the convolutional layers to fuse samples of different sizes, and the number of feature maps in each layer is consistent with MS-DenseNet. The parameters of SVM, EMP, SRC, SC-MK, PCA-E, MSSRC, CCJSR, MSDNet, DFFN, and SOAL are set to the default values as reported in their references [11], [18], [39]–[41], [43]–[47]. All the classification methods are repeated ten times to reduce the bias and the average values are reported in the Tables II-VI.

The first experiment is performed on the Washington DC data set. With 50 samples per class (Nr = 50) being randomly selected for training, the OA, AA, and Kappa, which are calculated from different methods, are shown in Table II. It can be observed that the OA, AA, and Kappa yielded by the proposed MS-DenseNet framework are better than those from other methods. Fig. 7 shows the classification maps of the Washington DC image by various classification methods.

The second experiment is performed on the University of Pavia data set. Table III shows the OA, AA, and Kappa of different methods with 50 samples per class (Nr = 50) being selected for training. Similar to the above experiment, the best results are also obtained by the proposed methods, which demonstrates the effectiveness of the proposed MS-DenseNet framework. Fig. 8 shows the classification maps of the University of Pavia image by fourteen classification methods.

The third experiment is performed on the Houston University data set. With 30 training samples per class (Nr = 30), OA,

1051-8215 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Helmholtz-Zentrum Dresden-Rossendorf. Downloaded on December 22,2020 at 07:52:34 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.2975566, IEEE Transactions on Circuits and Systems for Video Technology

TABLE II
CLASSIFICATION PERFORMANCE OF THE WASHINGTON DC DATA SET WITH THE SVM, EMP, SRC, SC-MK, IIDF, PCA-E, MSSRC, CCJSR, MSDNET, DFFN, SOAL, MS-NET ,AS WELL AS THE PROPOSED MS-FC-DENSENET AND MS-DENSENET.

Class	SVM	EMP	SRC	SC-MK	IIDF	PCA-E	MSSRC	CCJSR	MSDNet	DFFN	SOAL	MS-Net	MS-FC-DenseNet	MS-DenseNet
Roofs	92.69	89.45	80.23	95.17	96.93	99.15	92.47	**99.96**	95.12	88.29	96.22	94.01	96.28	95.75
Road	96.97	97.63	96.14	97.69	97.38	97.00	97.00	94.99	97.76	94.41	**100.00**	96.44	98.73	99.29
Grass	76.31	87.76	88.83	92.11	89.00	92.15	95.79	82.92	92.51	89.46	96.59	93.81	96.47	**97.11**
Trail	91.95	97.38	96.94	98.37	99.06	98.21	98.57	93.57	97.32	96.69	97.11	97.53	99.47	**99.65**
Trees	95.35	96.71	95.14	97.36	96.83	97.50	94.79	97.69	97.79	95.11	98.77	97.93	**99.42**	99.16
Shadow	**98.82**	96.67	98.17	96.63	98.68	87.93	81.98	91.02	89.45	90.41	87.76	93.80	98.09	98.12
OA	91.59	93.37	90.21	96.03	96.19	96.01	93.64	93.88	95.20	92.58	96.43	95.31	97.63	**97.74**
AA	92.01	94.27	92.58	96.22	96.31	95.32	93.43	93.36	94.99	92.40	96.07	95.59	98.08	**98.18**
Kappa	89.59	91.80	88.57	95.04	95.27	95.05	92.35	92.68	94.11	90.13	95.54	94.18	97.10	**97.20**

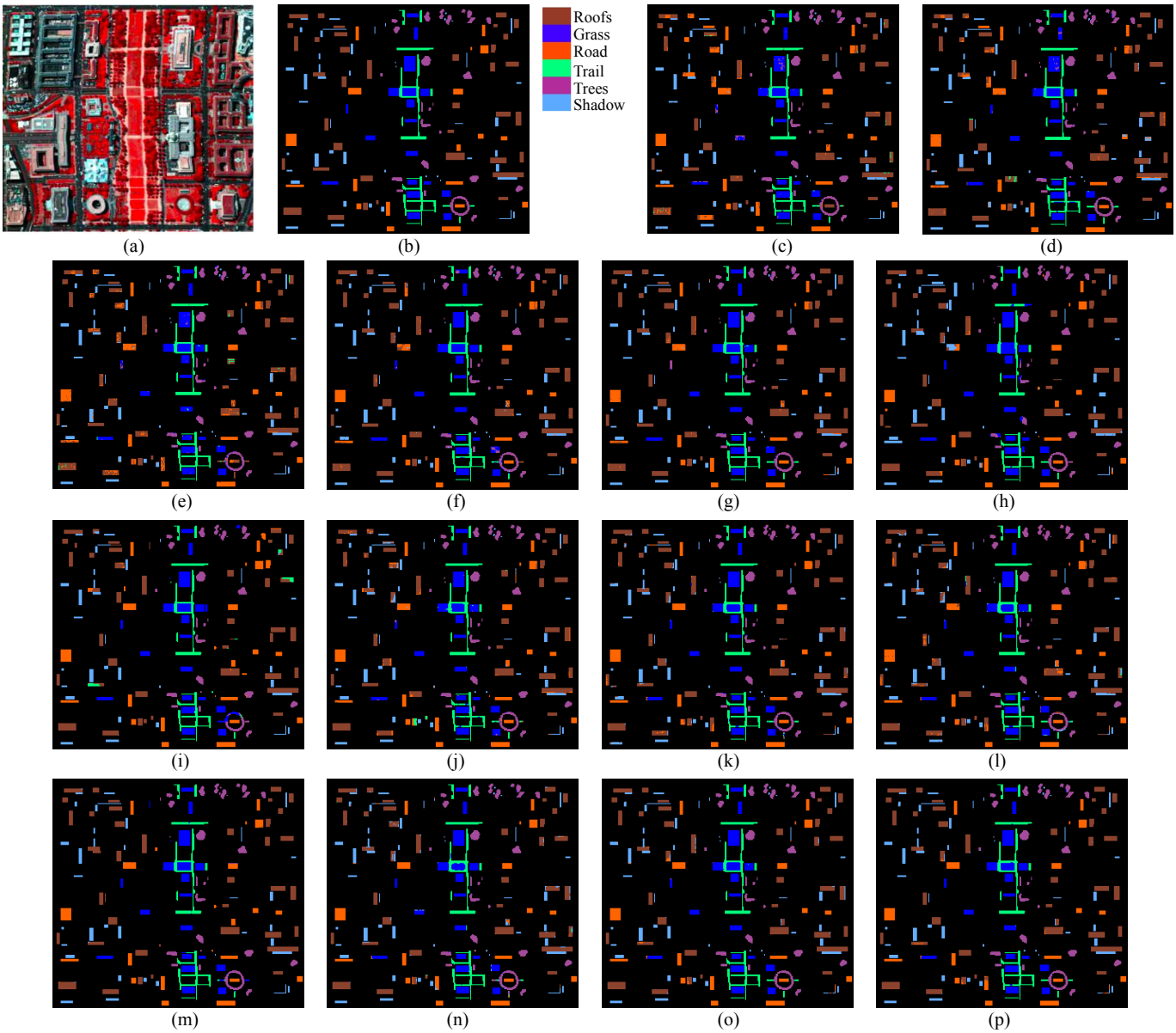

Fig. 7. Washington DC data set. (a) Three-band color image. (b) Corresponding reference data and maps from different compared methods. (c) SVM (OA=91.46%), (d) EMP (OA=94.35%), (e) SRC (OA=90.19%), (f) SC-MK (OA=96.16%), (g) IIDF OA=97.37%), (h) PCA-E (OA=95.43%), (i) MSSRC (OA=93.79%), (j) CCJSR (OA=94.82%), (k) MSDNet (OA=95.61%), (l) DFFN (OA=93.42%), (m) SOAL (OA=98.03%), (n) MS-Net (OA=95.17%), (o) MS-FC-DenseNet (OA=97.54%), (p) MS-DenseNet (OA=98.11%).

1051-8215 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Helmholtz-Zentrum Dresden-Rossendorf. Downloaded on December 22,2020 at 07:52:34 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.2975566, IEEE Transactions on Circuits and Systems for Video Technology

TABLE III
CLASSIFICATION PERFORMANCE OF THE UNIVERSITY OF PAVIA DATA SET WITH THE SVM, EMP, SRC, SC-MK, IIDF, PCA-E, MSSRC, CCJSR, MSDNET, DFFN, SOAL, MS-NET ,AS WELL AS THE PROPOSED MS-FC-DENSENET AND MS-DENSENET.

Class	SVM	EMP	SRC	SC-MK	IIDF	PCA-E	MSSRC	CCJSR	MSDNet	DFFN	SOAL	MS-Net	MS-FC-DenseNet	MS-DenseNet
Asphalt	88.41	96.84	69.03	98.13	93.14	98.21	76.48	95.99	98.72	90.88	**99.93**	97.53	97.75	96.82
Meadows	92.89	95.28	92.07	99.81	98.72	99.64	94.37	90.57	99.40	98.77	**100.00**	95.52	99.03	99.78
Gravel	67.21	87.08	55.66	91.54	99.15	96.63	99.60	60.16	99.86	99.91	99.43	99.90	**100.00**	99.97
Trees	95.31	91.58	73.18	93.01	93.56	94.72	85.78	78.34	89.62	91.99	82.02	97.96	98.74	**98.80**
Metal sheets	98.06	99.70	98.63	98.54	**100.00**	97.53	95.26	75.85	99.86	99.93	**100.00**	99.98	**100.00**	**100.00**
Bare soil	95.53	63.12	43.85	99.84	99.85	99.99	**100.00**	50.64	**100.00**	**100.00**	**100.00**	99.76	**100.00**	**100.00**
Bitumen	79.30	92.80	68.30	**100.00**	99.29	96.45	**100.00**	61.84	**100.00**	99.97	**100.00**	99.92	**100.00**	**100.00**
Bricks	79.67	93.30	70.71	96.44	87.83	92.74	96.92	57.18	98.92	97.49	99.92	98.44	99.61	**99.93**
Shadows	99.94	99.35	93.72	99.89	99.75	95.17	62.71	71.83	97.12	97.33	98.86	99.97	**100.00**	**100.00**
OA	83.58	87.75	71.69	97.85	96.62	98.04	91.68	74.67	98.65	97.54	98.83	97.33	99.10	**99.32**
AA	88.48	91.01	73.91	97.47	96.81	96.79	90.12	71.37	98.17	97.36	97.80	98.78	99.46	**99.48**
Kappa	78.05	84.38	63.70	97.15	95.52	97.40	89.00	67.18	98.13	96.11	98.33	96.67	98.81	**99.09**

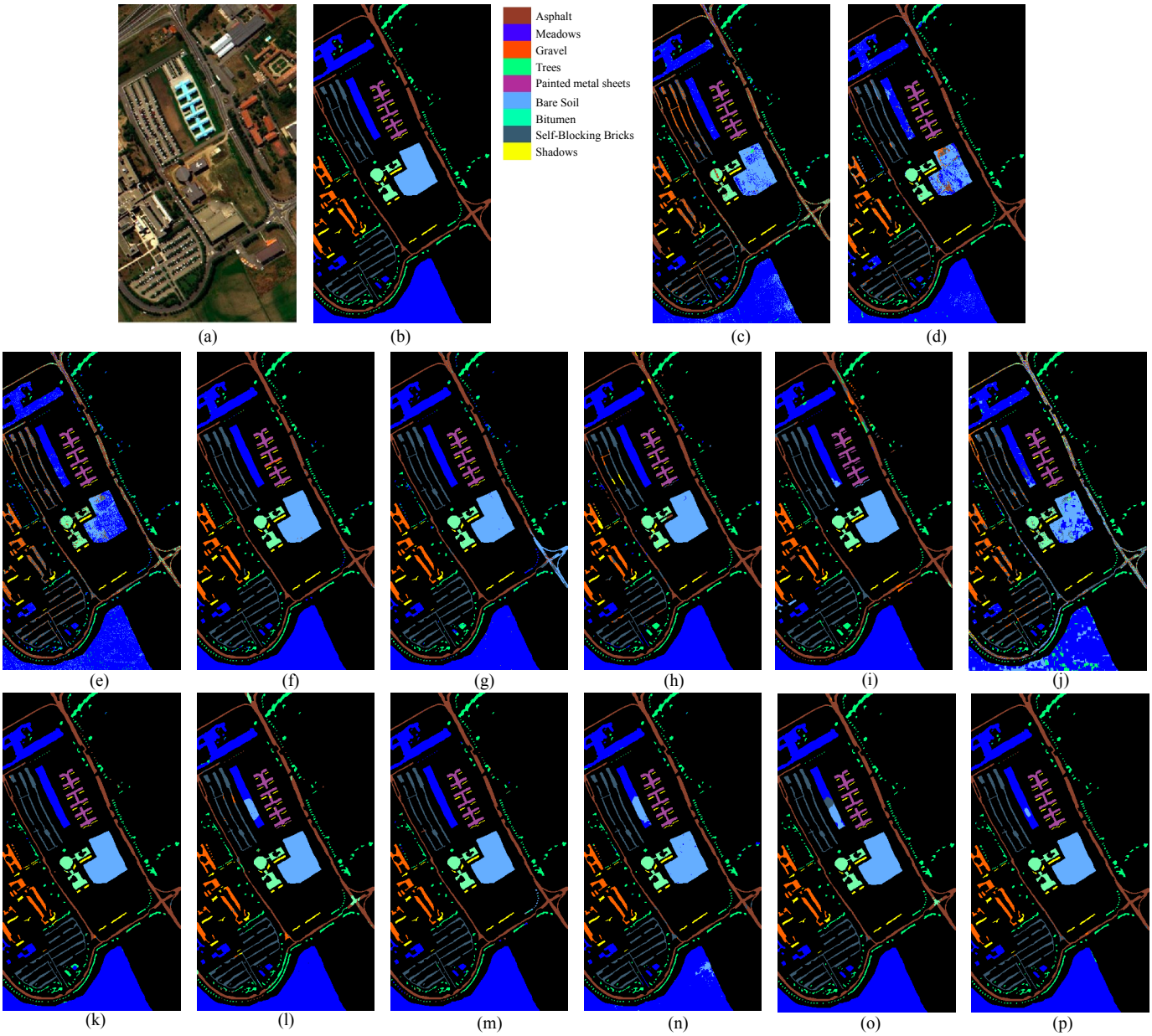

Fig. 8. University of Pavia data set. (a) Three-band color image. (b) Corresponding reference data and maps from different compared methods. (c) SVM (OA=88.92%), (d) EMP (OA=91.05%), (e) SRC (OA=77.66%), (f) SC-MK (OA=98.05%), (g) IIDF (OA=98.75%), (h) PCA-E (OA=99.03%), (i) MSSRC (OA=93.71%), (j) CCJSR (OA=76.35%), (k) MSDNet (OA=98.83%), (l) DFFN (OA=97.72%), (m) SOAL (OA=98.83%), (n) MS-Net, (OA=97.35%), (o) MS-FC-DenseNet (OA=98.92%), (p) MS-DenseNet (OA=99.09%).

1051-8215 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Helmholtz-Zentrum Dresden-Rossendorf. Downloaded on December 22,2020 at 07:52:34 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.2975566, IEEE Transactions on Circuits and Systems for Video Technology

AA, and Kappa of all fourteen methods are shown in Table IV. It can be observed that the proposed MS-DenseNet and MS-FC-DenseNet still obtains the best results in the terms of the OA, AA, and Kappa. Specifically, the classification accuracies in terms of OA metric by the MS-FC-DenseNet and MS-DenseNet frameworks are 95.52% and 95.91%, respectively, while the rest competetive methods have the OA of less than 94%. Furthermore, the accuracies of the eighth class obtained by all methods are not satisfactory. The reason is that within class variability is high which cannot be fully represented using a limited number of training samples. Fig. 9 shows the classification maps of the Houston University data set obtained by different methods.

The fouth experiment is performed on the Indian Pines data set. Table V shows the OA, AA, and Kappa of different methods with 5% samples per class (Nr = 5%) being selected for training. The Indian Pines data set is an imbalanced data set with fewer than 50 samples in three categories. Except for the SOAL, the proposed MS-DenseNet framework has better classification performance than other methods. But, the SOAL is an active learning-based methods which is hardly affected by the number of training samples. Fig. 10 shows the classification maps of the Indian Pines data set obtained by all fourteen methods.

The fifth experiment is performed on the Salinas data set. With 50 training samples per class (Nr = 50), OA, AA, and Kappa of all fourteen methods are shown in Table VI. Except for the SOAL, the classification result in terms of OA metric of the MS-DenseNet framework is better than classification results from other methods. From Fig 11 and Table VI, compared with the SOAL, more misclassification results from the categories of the vinyard_untrained and the grapes_untrained are obtained by the proposed method. Fig. 11 shows the classification maps of the Salinas data set obtained by different methods.

E. Effect of Different Number of Training Samples

In this section, the influence of different training sets on the performance of the MS-DenseNet is analyzed using three test images (i.e. Washington DC image, University of Pavia image and Houston University image), as shown in Fig. 12. As can be observed, more training samples can improve the performance of the proposed framework in classification tasks and the robustness of the proposed MS-DenseNet framework. From Fig. 12, Table V and VI, the proposed method delivers slight inferior performance to the SOAL method on the Indian Pines data set, the Salinas data set, and some sets of training samples of the Washington DC data set, the University of Pavia data set and the Houston University data set. This is because the SOAL is a semisupervised and active learning-based method, which can iteratively detect and then add new samples for training. Therefore, it is hardly affected by the training samples and thus can achieve very good results. Because the proportion number of the labeled samples of the Indian Pines image (10249 labeled samples/21025 samples) and the Salinas image (54129 labeled samples/111104 samples) are more than those of the University of Pavia image (42776 labeled samples/207400 samples), the Washington DC image (9899 labeled samples/85960 samples) and the Houston University image (15029 labeled samples/664845 samples) [48], and thus the Indian pines image and the Salinas image are comparatively simple to be classified. Even so, the proposed MS-DenseNet can consistently outperform other compared methods in most different sets of training samples when conducting experiments on the Washington DC data set, the University of Pavia data set and the Houston University data set.

V. Conclusion Results and Discussion

In this paper, a novel densely-connected convolutional network framework is proposed to integrate the multiple scale information of HSIs for the classification task. Specifically, both multiple scale information in multiple scale patches and different layers are utilized to improve the classification accuracy. The surrounding patches of multiple scales for each pixel are extracted to reflect complex spatial structures. In addition, the dense block is then adopted to fuse complementary yet correlated information among different layers and jointly extract multiple scale features. Experimental results demonstrated that, compared with other state-of-the-art frameworks recently presented in the literature, the proposed MS-DenseNet framework can obtain better classification accuracies on three real HSIs.

In this paper, the input of the MS-DenseNet is some principal components produced by the PCA method from raw data due to the limitation of computer hardware, which may not well consider correlations among spectral bands. Moreover, a limited number of training samples cannot fully reflect effective features to achieve an accurate classification task when the within class variance is high. Therefore, how to classify the HSIs without preprocessing and enhance the characterization ability of the training samples to reflect information will be the focuses of our future works.

Acknowledgment

The authors would like gratefully thank the Editors and the Anonymous Reviewers for their valuable comments and suggestions, which greatly helped us to enhance the technical quality and presentation of this paper.

References

[1] J. Fan, T. Chen, and S. Lu, "Superpixel guided deep-sparse-representation learning for hyperspectral image classification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 11, pp. 3163–3173, Nov. 2018.

[2] R. Ribeiro, G. Cruz, J. Matos, and A. Bernardino, "A data set for airborne maritime surveillance environments," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 9, pp. 2720–2732, Sep. 2019.

[3] E. Honkavaara, M. A. Eskelinen, I. Pölönen, H. Saari, H. Ojanen, R. Mannila, C. Holmlund, T. Hakala, P. Litkey, T. Rosnell, N. Viljanen, and M. Pulkkanen, "Remote sensing of 3-d geometry and surface moisture of a peat production area using hyperspectral frame cameras in visible to short-wave infrared spectral ranges onboard a small unmanned airborne vehicle (UAV)," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 9, pp. 5440–5454, Sep. 2016.

[4] J. Pontius, M. Martin, L. Plourde, and R. Hallett, "Ash decline assessment in emerald ash borer-infested regions: A test of tree-level, hyperspectral technologies," *Remote Sens. Environ.*, vol. 112, no. 5, pp. 2665–2676, May. 2008.

1051-8215 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Helmholtz-Zentrum Dresden-Rossendorf. Downloaded on December 22,2020 at 07:52:34 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.2975566, IEEE Transactions on Circuits and Systems for Video Technology

TABLE IV
CLASSIFICATION PERFORMANCE OF THE HOUSTON UNIVERSITY DATA SET WITH THE SVM, EMP, SRC, SC-MK, IIDF, PCA-E, MSSRC, CCJSR, MSDNET, DFFN, SOAL, MS-NET ,AS WELL AS THE PROPOSED MS-FC-DENSENET AND MS-DENSENET.

Class	SVM	EMP	SRC	SC-MK	IIDF	PCA-E	MSSRC	CCJSR	MSDNet	DFFN	SOAL	MS-Net	MS-FC-DenseNet	MS-DenseNet
Healthy grass	94.87	96.40	92.22	92.15	93.70	94.26	87.61	96.41	92.39	90.97	85.43	93.61	95.34	**96.64**
Stressed grass	95.61	91.58	97.83	87.47	97.03	84.57	85.13	87.76	94.77	**98.03**	86.12	96.54	97.06	97.43
Synthetic grass	99.70	94.72	99.73	**100.00**	**100.00**	**100.00**	**100.00**	99.75	99.62	99.11	**100.00**	99.80	**100.00**	99.87
Trees	96.13	93.11	91.01	89.54	98.88	93.27	95.72	95.86	83.53	80.64	85.67	95.72	98.84	**99.68**
Soil	91.48	92.07	96.91	94.54	95.51	**100.00**	99.02	87.87	99.67	98.10	**100.00**	98.65	99.98	99.5
Water	99.13	98.98	94.51	96.27	98.51	86.26	87.23	**100.00**	99.58	97.23	87.30	98.87	98.51	98.71
Residential	84.15	88.06	68.58	85.32	93.67	73.25	58.48	87.01	94.91	92.56	88.46	93.38	97.65	**99.35**
Commercial	77.42	77.23	71.24	76.43	86.43	83.07	74.52	**93.01**	82.87	77.18	90.86	74.16	78.82	80.08
Road	70.99	69.43	67.43	82.34	85.85	77.14	86.51	85.86	74.06	68.98	**94.85**	85.87	88.87	90.29
Highway	81.45	82.30	86.38	92.67	94.59	94.92	95.09	75.63	**100.00**	96.67	99.16	95.38	98.01	94.97
Railway	82.96	69.43	64.20	93.93	96.59	95.71	92.87	81.25	97.52	97.77	**98.34**	87.30	96.35	97.86
Parking lot1	73.58	71.21	65.80	89.66	91.33	**97.91**	91.53	78.07	96.76	92.86	96.26	88.50	97.46	97.22
Parking lot2	54.84	86.10	37.31	76.72	77.13	89.47	77.56	79.93	**98.47**	98.58	81.14	91.95	95.35	95.26
Tennis court	94.54	93.42	98.79	**100.00**	**100.00**	**100.00**	**100.00**	81.83	**100.00**	**100.00**	**100.00**	99.83	**100.00**	**100.00**
Running track	99.49	99.71	98.98	99.87	99.29	**100.00**	98.48	99.87	98.81	95.56	**100.00**	99.63	**100.00**	**100.00**
OA	86.04	85.06	81.38	89.51	93.53	89.89	87.80	87.16	92.90	90.83	93.03	92.13	95.52	**95.91**
AA	86.43	86.92	82.06	90.46	93.90	91.32	88.65	88.67	94.20	92.28	92.91	93.28	96.15	**96.46**
Kappa	84.89	83.84	79.86	88.68	93.01	89.07	86.95	86.04	92.33	90.15	92.42	91.49	95.16	**95.58**

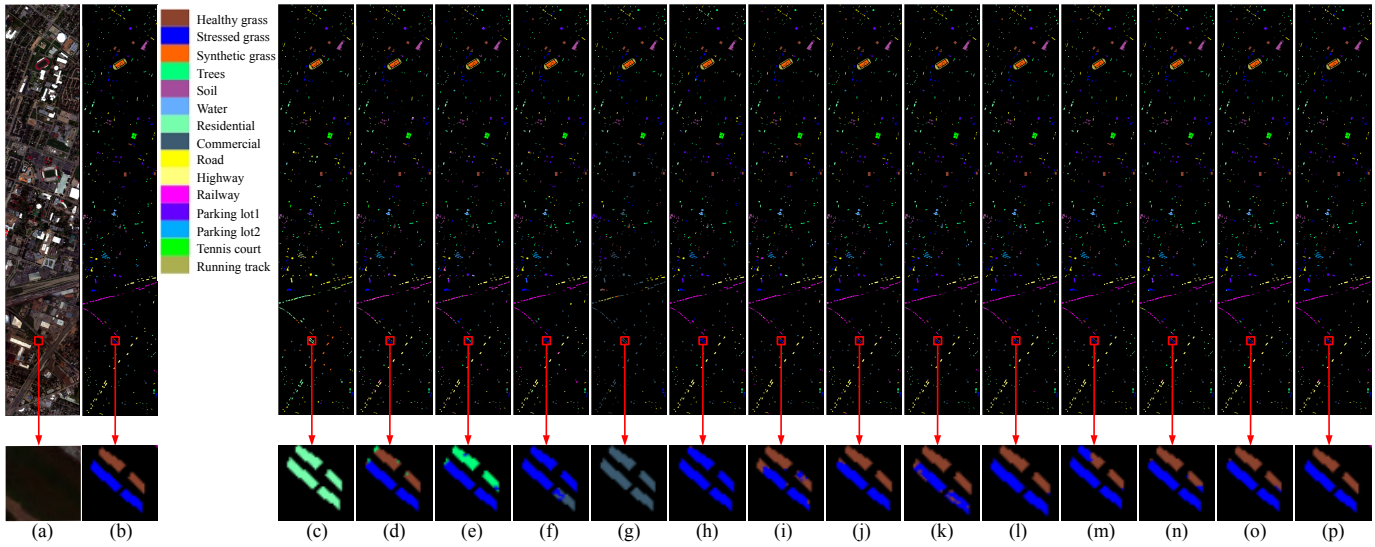

Fig. 9. Houston University data set. (a) Three-band color image. (b) Corresponding reference data and maps from different compared methods. (c) SVM (OA=86.04%), (d) EMP, (OA=85.06%), (e) SRC (OA=81.21%), (f) SC-MK (OA=89.77%), (g) IIDF (OA=93.42%), (h) PCA-E (OA=89.94%), (i) MSSRC (OA=87.23%), (j) CCJSR (OA=92.50%), (k) MSDNet (OA=93.11%), (l) DFFN (OA=91.01%), (m) SOAL (OA=94.57%), (n) MS-Net (OA=92.12%), (o) MS-FC-DenseNet (OA=95.35%), (p) MS-DenseNet (OA=96.12%).

TABLE V
CLASSIFICATION PERFORMANCE OF THE INDIAN PINES DATA SET WITH THE SVM, EMP, SRC, SC-MK, IIDF, PCA-E, MSSRC, CCJSR, MSDNET, DFFN, SOAL, MS-NET ,AS WELL AS THE PROPOSED MS-FC-DENSENET AND MS-DENSENET.

Class	SVM	EMP	SRC	SC-MK	IIDF	PCA-E	MSSRC	CCJSR	MSDNet	DFFN	SOAL	MS-Net	MS-FC-DenseNet	MS-DenseNet
Alfalfa	43.37	77.56	25.97	98.28	**100.00**	**100.00**	96.40	**100.00**	93.06	97.58	**100.00**	90.70	98.92	87.67
Corn-notill	73.09	76.71	45.23	96.67	89.97	95.42	93.72	89.97	93.67	93.42	**98.89**	96.61	91.30	96.26
Corn-mintill	72.79	80.01	41.63	93.66	86.84	96.88	99.37	90.57	97.54	97.75	99.24	99.11	92.26	**99.87**
Corn	45.50	62.82	37.72	87.58	88.69	97.94	**99.76**	85.76	85.01	94.49	96.89	92.00	81.33	90.18
Grass-pasture	91.15	81.57	77.46	88.38	97.08	96.78	**98.69**	96.87	93.05	91.83	98.03	95.85	95.63	95.50
Grass-trees	77.38	85.71	90.58	**100.00**	96.86	98.75	99.96	89.94	98.31	97.02	99.86	98.70	99.28	**100.00**
Grass-pasture-mowed	38.61	86.88	88.85	89.97	71.03	86.38	**99.90**	65.68	96.19	80.68	96.15	84.62	92.31	96.82
Hay-windrowed	83.95	99.56	93.56	**100.00**	99.34	**100.00**	**100.00**	98.75	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
Oats	31.98	35.47	37.23	**100.00**	48.18	94.04	**100.00**	84.67	78.99	94.65	**100.00**	68.42	73.68	**100.00**
Soybean-notill	71.50	81.28	64.85	91.87	93.28	96.25	98.53	87.88	95.64	92.54	**100.00**	95.56	93.93	94.19
Soybean-mintill	72.71	86.61	69.34	98.71	95.62	98.48	92.97	93.54	99.27	96.31	**99.96**	97.64	97.47	98.26
Soybean-clean	51.91	73.29	37.69	91.97	89.12	94.41	93.44	88.82	97.55	98.49	**99.29**	96.45	97.16	97.92
Wheat	81.90	99.71	91.11	99.85	98.73	97.72	99.67	95.53	95.92	96.82	**100.00**	94.33	**100.00**	**100.00**
Woods	91.56	98.67	86.24	99.11	98.51	99.77	99.75	98.39	97.38	96.91	**100.00**	99.25	99.08	**100.00**
Buildings-Grass-Trees-Drives	46.68	95.21	32.90	98.84	97.43	98.94	98.75	93.63	93.48	98.82	98.63	99.68	99.73	**100.00**
Stone-Steel-Towers	95.74	86.32	84.48	98.21	98.43	99.02	99.48	97.45	90.03	87.41	98.86	96.64	**100.00**	**100.00**
OA	73.95	86.61	64.86	96.25	94.03	97.49	96.60	92.50	96.64	95.82	**99.49**	97.40	95.59	97.66
AA	66.86	81.71	62.80	95.82	90.57	96.92	98.15	91.09	94.01	94.67	**99.11**	94.10	94.51	97.29
Kappa	70.47	84.74	59.79	95.65	93.27	97.15	95.96	91.44	96.15	95.25	**99.41**	97.04	94.97	97.46

1051-8215 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Helmholtz-Zentrum Dresden-Rossendorf. Downloaded on December 22,2020 at 07:52:34 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.2975566, IEEE Transactions on Circuits and Systems for Video Technology

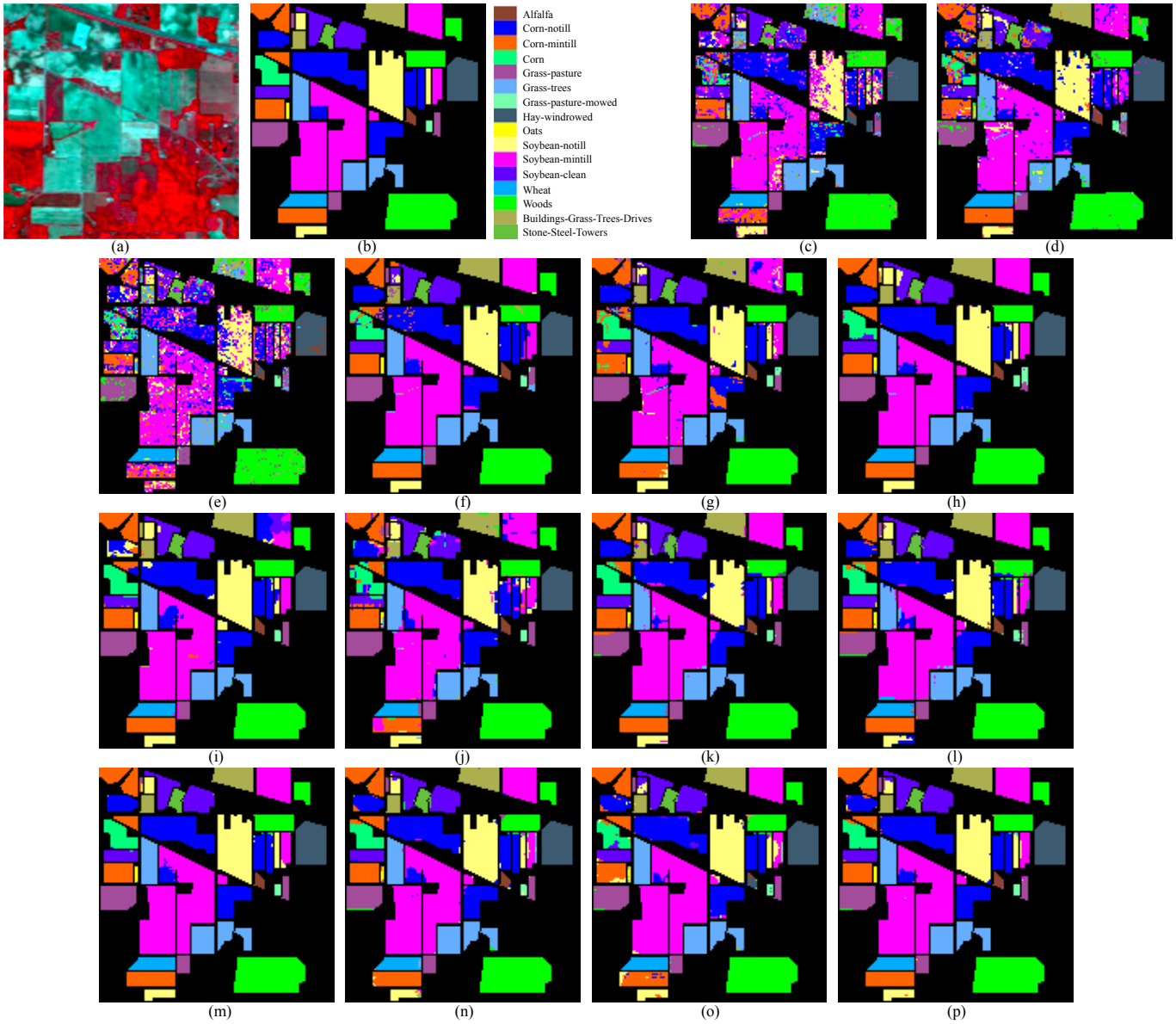

Fig. 10. Indian Pines data set. (a) Three-band color image. (b) Corresponding reference data and maps from different compared methods. (c) SVM (OA=75.30%), (d) EMP, (OA=86.37%), (e) SRC (OA=66.19%), (f) SC-MK (OA=96.31%), (g) IIDF (OA=94.21%), (h) PCA-E (OA=97.73%), (i) MSSRC (OA=93.47%), (j) CCJSR (OA=92.71%), (k) MSDNet (OA=96.60%), (l) DFFN (OA=95.91%), (m) SOAL (OA=99.32%), (n) MS-Net (OA=97.47%), (o) MS-FC-DenseNet (OA=95.57%), (p) MS-DenseNet (OA=98.65%).

TABLE VI
CLASSIFICATION PERFORMANCE OF THE SALINAS DATA SET WITH THE SVM, EMP, SRC, SC-MK, IIDF, PCA-E, MSSRC, CCJSR, MSDNET, DFFN, SOAL, MS-NET ,AS WELL AS THE PROPOSED MS-FC-DENSENET AND MS-DENSENET.

Class	SVM	EMP	SRC	SC-MK	IIDF	PCA-E	MSSRC	CCJSR	MSDNet	DFFN	SOAL	MS-Net	MS-FC-DenseNet	MS-DenseNet
Brocoli_green_weeds_1	**100.00**	94.18	98.64	**100.00**	**100.00**	**100.00**	**100.00**	99.95	**100.00**	99.79	**100.00**	**100.00**	**100.00**	**100.00**
Brocoli_green_weeds_2	98.13	87.09	98.23	**100.00**	99.95	**100.00**	**100.00**	**100.00**	99.95	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
Fallow	94.48	86.49	97.53	99.08	99.99	99.46	99.50	97.84	98.67	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
Fallow_rough_plow	99.37	99.55	99.59	99.18	97.65	96.69	98.93	99.22	99.54	**99.93**	**99.93**	99.11	99.03	99.40
Fallow_smooth	98.60	96.50	93.23	98.34	**99.99**	99.67	98.85	97.04	99.63	99.89	99.48	99.92	99.31	99.96
Stubble	**100.00**	98.85	99.32	99.51	**100.00**	**100.00**	99.47	99.77	99.95	99.72	**100.00**	99.97	**100.00**	**100.00**
Celery	99.65	96.83	98.90	99.72	99.96	99.68	99.41	99.07	99.97	99.97	**100.00**	99.55	99.04	99.97
Grapes_untrained	77.62	76.46	63.41	94.97	98.69	98.27	96.33	85.83	98.38	96.95	**99.69**	90.76	99.18	98.81
Soil_vinyard_develop	99.32	97.53	97.32	99.79	99.97	99.74	100.00	99.22	**100.00**	99.52	**100.00**	99.66	99.62	99.98
Corn_senesced_green_weeds	85.55	96.93	85.93	97.42	98.07	99.67	93.36	95.54	99.94	**99.99**	99.91	99.47	98.10	98.51
Lettuce_romaine_4wk	92.91	98.87	99.02	98.01	99.94	99.60	**100.00**	97.70	**100.00**	99.91	**100.00**	99.51	**100.00**	**100.00**
Lettuce_romaine_5wk	99.67	82.78	99.76	99.49	99.95	99.67	**100.00**	97.19	99.95	99.17	**100.00**	99.63	**100.00**	**100.00**
Lettuce_romaine_6wk	96.14	98.07	96.35	98.24	96.63	98.79	96.69	96.63	99.14	99.88	99.56	99.42	99.28	**100.00**
Lettuce_romaine_7wk	97.42	95.64	96.67	94.42	96.66	96.96	96.16	97.92	99.86	**100.00**	99.72	99.61	**100.00**	99.71
Vinyard_untrained	65.17	75.92	59.27	97.38	99.01	98.04	99.42	64.74	98.01	99.27	**99.97**	98.96	99.20	99.02
Vinyard_vertical_trellis	99.81	97.42	97.70	**100.00**	**100.00**	99.69	96.77	99.19	**100.00**	**100.00**	98.95	98.29	99.18	99.83
OA	88.79	88.42	84.68	98.02	99.31	99.08	98.36	90.35	99.28	98.46	**99.85**	97.69	99.41	99.50
AA	93.99	92.44	92.55	98.47	99.15	99.12	99.43	95.43	99.56	99.62	**99.83**	98.99	99.50	99.70
Kappa	87.51	87.15	82.98	97.82	99.22	99.01	98.23	89.24	99.19	98.24	**99.83**	97.43	99.38	99.44

1051-8215 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Helmholtz-Zentrum Dresden-Rossendorf. Downloaded on December 22,2020 at 07:52:34 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.2975566, IEEE Transactions on Circuits and Systems for Video Technology

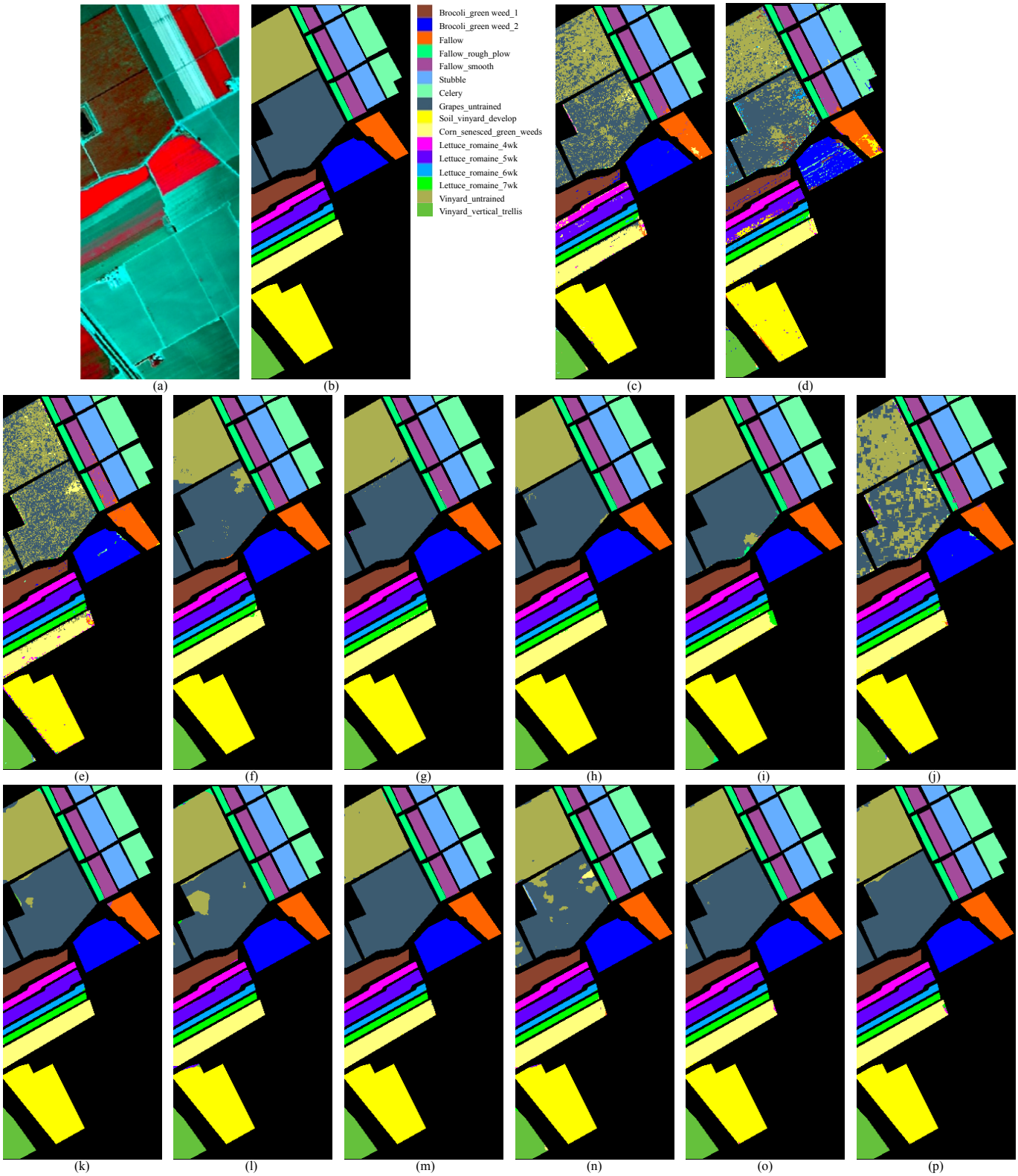

Fig. 11. Salinas data set. (a) Three-band color image. (b) Corresponding reference data and maps from different compared methods. (c) SVM (OA=88.69%), (d) EMP, (OA=88.27%), (e) SRC (OA=84.87%), (f) SC-MK (OA=98.24%), (g) IIDF (OA=99.58%), (h) PCA-E (OA=99.61%), (i) MSSRC (OA=98.86%), (j) CCJSR (OA=89.41%), (k) MSDNet (OA=99.22%), (l) DFFN (OA=98.01%), (m) SOAL (OA=99.78%), (n) MS-Net (OA=97.69%), (o) MS-FC-DenseNet (OA=99.71%), (p) MS-DenseNet (OA=99.51%).

1051-8215 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Helmholtz-Zentrum Dresden-Rossendorf. Downloaded on December 22,2020 at 07:52:34 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.2975566, IEEE Transactions on Circuits and Systems for Video Technology

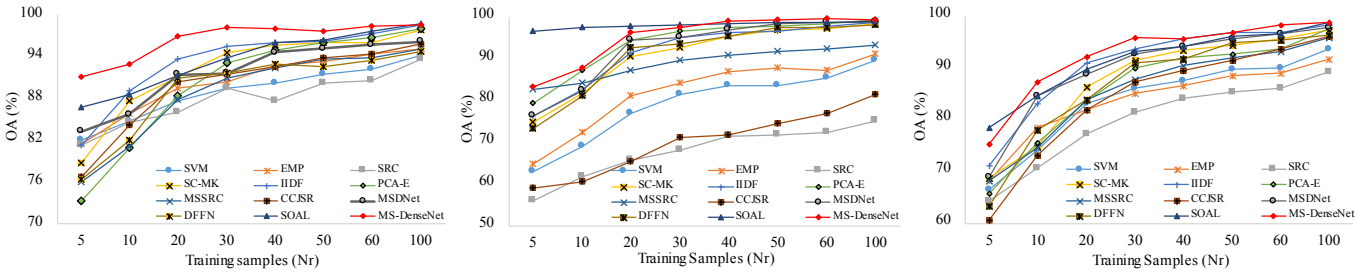

Fig. 12. Classification results of different training samples on the (a) Washington DC, (b) University of Pavia and (c) Houston University. Nr ϵ [5, 10, 20, 30, 40, 50, 60,100]

[5] L. Wang, Z. Xiong, G. Shi, W. Zeng, and F. Wu, "Simultaneous depth and spectral imaging with a cross-modal stereo system," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 3, pp. 812–817, Mar. 2018.

[6] K. Zhang, X. Xu, B. Han, L. R. Mansaray, Q. Guo, and J. Huang, "The influence of different spatial resolutions on the retrieval accuracy of sea surface wind speed with c-2po models using full polarization c-band sar," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 9, pp. 5015–5025, Sep. 2017.

[7] R. A. Farrugia and C. J. Debono, "A robust error detection mechanism for h.264/avc coded video sequences based on support vector machines," vol. 18, no. 12, pp. 1766–1770, Dec. 2008.

[8] P. Zhong and R. Wang, "Jointly learning the hybrid crf and mlr model for simultaneous denoising and classification of hyperspectral imagery," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 7, pp. 1319–1334, Jul. 2014.

[9] W. Wang, C. Wang, S. Liu, T. Zhang, and X. Cao, "Robust target tracking by online random forests and superpixels," vol. 28, no. 7, pp. 1609–1622, Jul. 2018.

[10] J. Xia, P. Ghamisi, N. Yokoya, and A. Iwasaki, "Random forest ensembles and extended multiextinction profiles for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 1, pp. 202–216, Jan. 2018.

[11] 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.

[12] 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.

[13] W. Fu, S. Li, and L. Fang, "Spectral-spatial hyperspectral image classification via superpixel merging and sparse representation," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, 2015, pp. 4971–4974.

[14] P. Zhong and R. Wang, "Modeling and classifying hyperspectral imagery by crfs with sparse higher order potentials," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 2, pp. 688–705, Feb. 2011.

[15] M. Golipour, H. Ghassemian, and F. Mirzapour, "Integrating hierarchical segmentation maps with mrf prior for classification of hyperspectral images in a bayesian framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 2, pp. 805–816, Feb. 2016.

[16] M. Graña and D. Chyzhyk, "Image understanding applications of lattice autoassociative memories," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 9, pp. 1920–1932, Sep. 2016.

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

[18] 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.

[19] J. Xie, N. He, L. Fang, and A. Plaza, "Scale-free convolutional neural network for remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6916–6928, Sep. 2019.

[20] S. López-Tapia, R. Molina, and N. P. de la Blanca, "Deep cnns for object detection using passive millimeter sensors," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 9, pp. 2580–2589, Sep. 2019.

[21] Y. Wang, L. Wang, H. Wang, and P. Li, "Ran: Resolution-aware network for image super-resolution," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 9, pp. 1259–1269, May. 2019.

[22] A. Santara, K. Mani, P. Hatwar, A. Singh, A. Garg, K. Padia, and P. Mitra, "Bass net: Band-adaptive spectral-spatial feature learning neural network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 9, pp. 5293–5301, Sep. 2017.

[23] 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.

[24] 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.

[25] 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.

[26] 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.

[27] P. Ghamisi, Y. Chen, and X. X. Zhu, "A self-improving convolution neural network for the classification of hyperspectral data," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 10, pp. 1537–1541, Oct. 2016.

[28] 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.

[29] X. Li, M. Ye, Y. Liu, and C. Zhu, "Adaptive deep convolutional neural networks for scene-specific object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 9, pp. 2538–2552, Sep. 2019.

[30] M. H. Rafiei and H. Adeli, "A new neural dynamic classification algorithm," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 12, pp. 3074–3083, Dec. 2017.

[31] 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.

[32] S. Mei, J. Ji, J. Hou, X. Li, and Q. Du, "Learning sensor-specific spatial-spectral features of hyperspectral images via convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 8, pp. 4520–4533, Aug. 2017.

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

[34] W. Song, J. Zhu, Y. Li, and C. Chen, "Image alignment by online robust pca via stochastic gradient descent," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 7, pp. 1241–1250, Jul. 2016.

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

[36] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2017, pp. 2261–2269.

[37] 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.

[38] L. Jiao, M. Liang, H. Chen, S. Yang, H. Liu, and X. Cao, "Deep fully convolutional network-based spatial distribution prediction for

1051-8215 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Helmholtz-Zentrum Dresden-Rossendorf. Downloaded on December 22,2020 at 07:52:34 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.2975566, IEEE Transactions on Circuits and Systems for Video Technology

hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 10, pp. 5585–5599, Oct. 2017.
[39] 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.
[40] 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.
[41] 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.
[42] 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.
[43] X. Kang, X. Xiang, S. Li, and J. A. Benediktsson, "PCA-based edge-preserving features for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 7140–7151, Dec. 2017.
[44] S. Zhang, S. Li, W. Fu, and L. Fang, "Multiscale superpixel-based sparse representation for hyperspectral image classification," *Remote Sens.*, vol. 9, no. 2, p. 139, Feb. 2017.
[45] B. Tu, X. Zhang, X. Kang, G. Zhang, J. Wang, and J. Wu, "Hyperspectral image classification via fusing correlation coefficient and joint sparse representation," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 3, pp. 340–344, Mar. 2018.
[46] G. Huang, D. Chen, T. Li, F. Wu, L. van der Maaten, and K. Q. Weinberger, "Multi-scale dense networks for resource efficient image classification," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–14.
[47] W. Song, S. Li, L. Fang, and T. Lu, "Hyperspectral image classification with deep feature fusion network," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3173–3184, Jun. 2018.
[48] S. Li, Q. Hao, G. Gao, and X. Kang, "The effect of ground truth on performance evaluation of hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 12, pp. 7195–7206, Dec 2018.

Jie Xie (S'18) received the B.Sc. degree from Hunan University of Science and Technology, Xiangtan, China, in 2015. He is currently pursuing the Ph.D. degree of control science and engineering in Hunan University, Changsha, China. His current research interests include hyperspectral image processing, remote sensing images processing, and deep learning.

Nanjun He (S'17) received the B.S. degree from the Central South University of Forestry and Technology, Changsha, China, in 2013. He is currently pursuing the Ph.D. degree with the Laboratory of Vision and Image Processing, Hunan University, Changsha. From 2017 to 2018, he was a Visiting Ph.D. Student with the Hyperspectral Computing Laboratory, University of Extremadura, Cáceres, Spain, supported by the China Scholarship Council. His research interests include remote sensing image classification and remote sensing object detection.

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

From September 2011 to September 2012, he was a visiting Ph.D. student with the Department of Ophthalmology, Duke University, Durham, NC, USA, supported by the China Scholarship Council. From August 2016 to September 2017, he was a Postdoc Researcher with the Department of Biomedical Engineering, Duke University, Durham, NC, USA. He is currently a Professor with the College of Electrical and Information Engineering, Hunan University. His research interests include sparse representation and multi-resolution analysis in remote sensing and medical image processing. He was a recipient of one 2nd-Grade National Award at the Nature and Science Progress of China in 2019.

Pedram Ghamisi (S'12–M'15–SM'18) received the B.Sc. degree in civil (survey) engineering from the Tehran South Campus of Azad University, Tehran, Iran, and the M.Sc. degree (Hons.) in remote sensing from the K. N. Toosi University of Technology, Tehran, in 2012. In 2013/2014, he spent seven months at the School of Geography, Planning and Environmental Management, The University of Queensland, Brisbane, QLD, Australia. He received the Ph.D. degree in electrical and computer engineering from the University of Iceland, Reykjavik, Iceland, in 2015.

After receiving his Ph.D. degree, he was a Post-Doctoral Research Fellow with the University of Iceland. In 2015, he received the prestigious Alexander von Humboldt Fellowship in 2015 and started his work as a Post-Doctoral Research Fellow with the Technical University of Munich (TUM), Munich, Germany, and Heidelberg University, Heidelberg, Germany, in October 2015. He was also a Research Scientist with the German Aerospace Center (DLR), Remote Sensing Technology Institute (IMF), Oberpfaffenhofen, Germany, from October 2015 to May 2018. In 2018, he won the prestigious High Potential Program and started his work as the Head of the Machine Learning Group, Helmholtz-Zentrum Dresden-Rossendorf (HZDR), Dresden, Germany. His research interests involve interdisciplinary research on remote sensing and machine (deep) learning, image and signal processing, and multisensory data fusion.

Dr. Ghamisi received the Best Researcher Award for M.Sc. students in K. N. Toosi University of Technology in the academic year 2010–2011. At the 2013 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Melbourne, July 2013, he received the IEEE Mikio Takagi Prize for winning the Student Paper Competition, competing with almost 70 submissions. In 2016, he was selected as the Talented International Researcher by Iran's National Elites Foundation. In 2017, he won the Data Fusion Contest 2017 organized by the Image Analysis and Data Fusion Technical Committee (IADF) of the Geoscience and Remote Sensing Society (IEEE-GRSS). He was also the Winner of the 2017 Best Reviewer Prize of the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS (GRSL). He serves as an Associate Editor for the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS (GRSL) and Remote Sensing.

1051-8215 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Helmholtz-Zentrum Dresden-Rossendorf. Downloaded on December 22,2020 at 07:52:34 UTC from IEEE Xplore. Restrictions apply.