Learning Multiscale Deep Features for High-Resolution Satellite Image Scene Classification

Qingshan Liu, *Senior Member, IEEE*, Renlong Hang, *Member, IEEE*, Huihui Song, and Zhi Li

***Abstract*— In this paper, we propose a multiscale deep feature learning method for high-resolution satellite image scene classification. Specifically, we first warp the original satellite image into multiple different scales. The images in each scale are employed to train a deep convolutional neural network (DCNN). However, simultaneously training multiple DCNNs is time-consuming. To address this issue, we explore DCNN with spatial pyramid pooling (SPP-net). Since different SPP-nets have the same number of parameters, which share the identical initial values, and only fine-tuning the parameters in fully connected layers ensures the effectiveness of each network, thereby greatly accelerating the training process. Then, the multiscale satellite images are fed into their corresponding SPP-nets, respectively, to extract multiscale deep features. Finally, a multiple kernel learning method is developed to automatically learn the optimal combination of such features. Experiments on two difficult data sets show that the proposed method achieves favorable performance compared with other state-of-the-art methods.**

***Index Terms*— Deep convolutional neural networks (DCNNs), feature fusion, multiple kernel learning (MKL), multiscale deep features, satellite image classification, spatial pyramid pooling.**

I. Introduction

REMOTE sensing image classification has been an active research topic in the past few decades, and most of the existing works primarily focus on pixelwise classification, which assigns label information to each pixel in a multispectral or hyperspectral image [1]–[5]. Although significant progress has been made in this area, pixels are not enough for the entire image understanding, because they have a few semantic meanings [6]. With the development of imaging techniques, a large amount of high spatial resolution satellite images become available [7]–[9], which opens new possibilities in remote sensing image analysis and classification.

However, satellite images with high spatial resolution pose many challenging issues in image classification. First, the enhanced resolution brings more details; thus, simple low-level features (e.g., intensity and textures) widely used in the case of low-resolution images are insufficient in capturing efficiently discriminative information [8]. For instance, Fig. 1(a) and (b) has similar color and texture features, but they belong to different categories (i.e., runway and freeway), which can be discriminated by adding the car information. Second, objects in the same type of scene might have different scales and orientations [10]. As shown in Fig. 1(c) and (d), the airplane in Fig. 1(d) is much smaller than that in Fig. 1(c), and their orientations are also different. Besides, high-resolution satellite images often consist of many different semantic classes, which makes further classification more difficult [11]. Taking Fig. 1(e) for example, the commercial scene comprises roads, buildings, trees, parking lots, and so on. Thus, developing effective feature representations is critical for solving these issues.

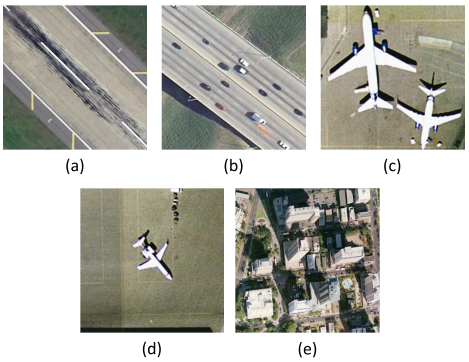


Fig. 1. Few examples of high-resolution satellite images. (a) Runway. (b) Freeway. (c) and (d) Airplane. (e) Commercial.

There are two popular feature representation models that are successfully used in satellite image classification. One is the bag of visual words (BOVWs) model [12]–[14], which generally includes three steps: 1) extracting man-made visual features, such as scale invariant feature transform (SIFT) [15] and histogram of oriented gradient [16] descriptors; 2) clustering features to form visual words (clustering centers) by using k-means or other clustering methods; and 3) mapping visual features to the closest word and generating a mid-level feature representation by word histograms. This model and its variants have been investigated in satellite image classification [11], [17]. However, it is an orderless collection of local descriptors, regardless of spatial information. To overcome this drawback, a spatial pyramid matching (SPM) method was proposed in [18], in which the image is first partitioned into increasingly fine subregions and then histograms of local features are extracted inside each subregion. Since satellite imagery generally does not have

Manuscript received April 5, 2016; revised January 18, 2017, May 1, 2017, and June 14, 2017; accepted August 13, 2017. Date of publication September 13, 2017; date of current version December 27, 2017. This work was supported in part by the Natural Science Foundation of China under Grant 61532009 and Grant 41501377 and in part by the Natural Science Foundation of Jiangsu Province, China under Grant 15KJA520001. *(Corresponding authors: Qingshan Liu; Renlong Hang.)*

The authors are with the Jiangsu Key Laboratory of Big Data Analysis Technology, School of Information and Control, Nanjing University of Information Science and Technology, Nanjing 210044, China (e-mail: qsliu@nuist.edu.cn; renlong_hang@163.com).

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

Digital Object Identifier 10.1109/TGRS.2017.2743243

0196-2892 © 2017 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.

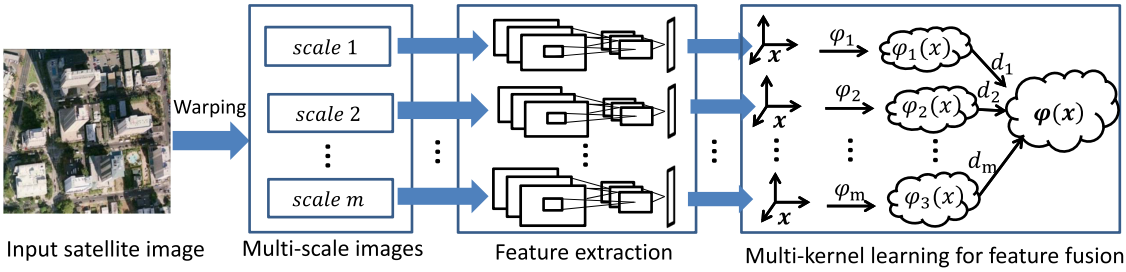


Fig. 2. Flowchart of the proposed method.

an absolute reference frame, the relative spatial arrangement of the image elements becomes very important. Accordingly, Yang and Newsam [9] proposed the spatial pyramid co-occurrence, which characterizes both the photometric and geometric information of an image. Unlike dividing the image into uniform cells in [9] and [18], Jiang *et al.* [19] proposed a randomized spatial partition to describe various image layouts.

Feature representation based on sparse coding (SC) is the other popular method for scene classification [20], [21]. Its basic idea is that the original signal can be sparsely reconstructed with respect to some fixed bases (dictionary) and the selected bases are enforced into as few categories as possible. In [7], a two-layer SC was proposed for satellite image classification. Sheng *et al.* [10] proposed to use SC to generate three mid-level representations based on SIFT, local ternary pattern histogram Fourier, and color histogram features, respectively. Recently, an unsupervised dictionary learning method has been proposed in [11], which achieves favorable performance in satellite image classification.

Although these methods have achieved promising results in satellite image classification, there still exist some shortcomings. For the BOVW models, a key step is how to extract low-level visual features. This process is generally handcrafted and heavily depends on experience and domain knowledge of designers. For the SC models, they can be considered as a single-layer feature learning architecture, which automatically selects a few vectors from a large pool of possible bases to encode an input signal [22], [23]. As discussed in [24], the shallow architectures have shown effectiveness in solving many simple or well-constrained problems, but their limited modeling and representational power are insufficient in complex scene cases like the high-resolution satellite images. Besides, SC focuses on searching for sparse representation of the original images, which may lose helpful discriminative information for the subsequent supervised classification.

Recently, deep learning, especially DCNN, has been widely used in natural image processing [23]. The core idea is to hierarchically learn high-level semantic features without human interactions. In 2012, Krizhevsky *et al.* [25] designed a DCNN architecture based on two graphics processing units with multiple convolutional and fully connected layers. This architecture achieved excellent classification results on the ImageNet 2012 Large Scale Visual Recognition Challenge. Afterward, a large amount of works about DCNN sprang up [26]–[31]. He *et al.* [31] proposed the DCNN with spatial pyramid pooling (SPP-net) to solve the size constraint problem of input images, which exists in most DCNN architectures. Benefiting from spatial pyramid pooling, SPP-net can be trained faster and achieves higher performance than DCNN. In the field of remote sensing image processing, DCNN has also attracted much attention [32]–[34].

In this paper, we employ SPP-net to automatically extract multiscale deep features of high-resolution satellite images. As shown in Fig. 1(c) and (d), the scales of objects in satellite images often vary. Traditional DCNNs are not able to sufficiently explore this information, because they can extract only the deep features of images from a predefined scale (e.g., 224×224). We, therefore, attempt to construct multiple DCNNs with different input scales to address this issue. However, it is well known that training a deep model costs much time, not to mention training multiple models simultaneously. Benefiting from spatial pyramid pooling, SPP-net can generate a fixed-length representation regardless of image size/scale. In other words, SPP-nets with different input image scales can exactly share the same weight parameters [35]. Besides, for each SPP-net, fine-tuning the parameters in fully connected layers ensures an efficient network, thus greatly accelerating the training process. Hence, we choose SPP-net as our basic deep model. Because of the large numbers of parameters and scarcity in training samples, the SPP-net inevitably poses the overfitting problem. We, therefore, take advantage of the training results using the ImageNet data set. Afterward, we use the trained SPP-nets to extract multiscale deep features. In the classification stage, we attempt to optimize the fusion weights of multiscale deep features and classifier parameters via the multiple kernel learning (MKL) method, making the learned fusion weights optimal for classification.

II. Methodology

The flowchart of the proposed method is shown in Fig. 2. The whole procedure consists of three steps: 1) warping the original satellite images into multiple scales; 2) extracting multiscale deep features using multiple SPP-nets; and 3) fusing multiscale deep features via a multikernel learning method. In Sections II-A–II-C, we will introduce the last two steps in detail.

A. SPP-Net Architecture

SPP-net was first proposed in [31] to address the size issue of input images. Here, we use it to automatically learn multiscale deep features of high-resolution satellite images. Specifically, we combine the prevalent seven-layer architecture

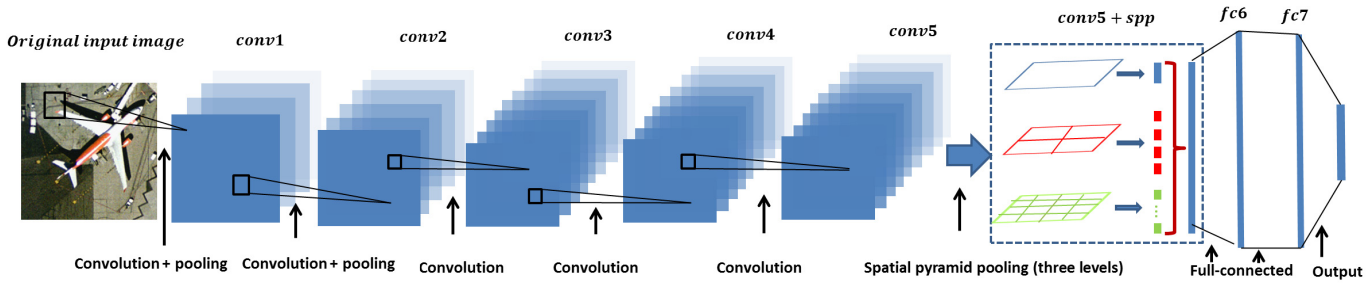


Fig. 3. Architecture of the SPP-net.

in [25] with spatial pyramid pooling (SPP). The designed architecture is shown in Fig. 3. The network contains five successive convolutional layers and two fully connected layers. They are concisely referred to as conv$i$, $i = \{1, \ldots, 5\}$ and fc$j$, $j = \{6, 7\}$. For instance, conv1 denotes that the first layer is a convolutional layer; fc6 indicates that the sixth layer is a fully connected layer. The first two convolutional layers are followed by max-pooling operators. They operate in a sliding-window manner and output feature maps representing the spatial layout of the responses. Before the first fully connected layer, SPP is exploited to pool the features from the last convolutional layer. Similar to SPM [18], we partition the feature maps into increasingly fine subregions, and pool the responses inside each subregion (throughout this paper, we use max pooling). Assume that the size of each feature map after the last convolutional layer is $a \times a$ pixels and each feature map is partitioned into $n \times n$ subregions. Then, SPP can be considered as convolution operators in a sliding-widow manner with window size win $= \lceil a/n \rceil$ and stride str $= \lfloor a/n \rfloor$, where $\lceil \cdot \rceil$ and $\lfloor \cdot \rfloor$ denote ceiling and floor operators, respectively. Fig. 3 demonstrates a three-level SPP configuration by setting $n \times n$ as 1×1, 2×2, and 4×4, respectively. The final output of SPP is to concatenate these three-level pooling results into a vector. This simple pooling operator largely reduces the number of parameters needed to be trained between the last convolution layer and the first fully connected layer. Thus, it is faster to train SPP-net than the traditional DCNNs. Besides, SPP extracts multiresolution information from the last convolutional layer, which improves the final classification results. Despite the varying sizes of input images, which lead to the varying sizes of feature maps at each convolutional layer, the lengths of input vectors to the first fully connected layer remain the same. This property ensures that the number of parameters remains unchanged. Therefore, the multiple SPP-nets are capable of sharing the same initial parameters.

B. Training Method

The above network contains more than 30 millions of parameters. Training such a network needs a large amount of samples. A canonical data set widely used in the DCNN architectures is ImageNet, consisting of millions of images. However, only hundreds of samples are available for high-resolution satellite image classification, which is far less than ImageNet. The most intuitional method to enlarge the number of training samples is the cropping or flipping operator [25], [31]. Nevertheless, it is still far enough to train an efficient network. Recently, Oquab *et al.* [36] proposed to transfer image representations learned with DCNN on large data sets to other visual recognition tasks with limited training data. Motivated by this paper, we propose to first pretrain the network in [25] using auxiliary ImageNet 2012 data set (Source task), and then fine-tune our SPP-nets by employing the training samples from satellite images (Target task).

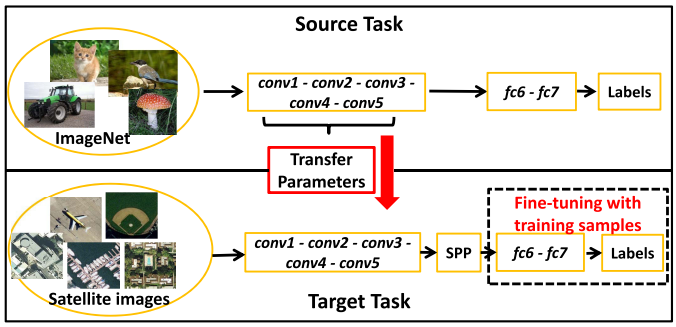


Fig. 4. Detailed training process of our method.

The training procedure of the source task is carried out via the open source Caffe DCNN library [37]. Specifically, a multinomial logistic regression function is optimized using the stochastic gradient descent algorithm based on the back propagation method [38]. The batch size and momentum are set to 256 and 0.9, respectively. The training is regularized by a weight decay of 0.0005 and dropout operators for the two fully connected layers (dropout ratio is set to 0.5). The initial learning rate is set to be 0.01. This value is fixed and used to update iteratively the weights. At each iteration, we calculate the classification accuracy of the validation set. When the accuracy stops increasing, we divide the learning rate by 10, and this new value is used to update the weights. The whole process is repeated until convergence. In our experiments, the learning rate reduces three times prior to termination (after 370k iterations) and the weights in each layer are initialized from a zero-mean Gaussian distribution with standard deviation $\sigma = 0.01$. After the pretraining of source task, the weight parameters learned in the five convolutional layers are then transferred to the target task and kept fixed. For the target task, we only need to fine-tune the last three layers (i.e., two fully connected layers and the output layer). The whole process is demonstrated in Fig. 4. It is worth noting that

the source task is pretrained only once, and the source task along with the target task shares the same initial parameters, which means that the learned parameters from the source task are directly transferred to the multiple SPP-nets. For each SPP-net, the parameters of fully connected layers are fine-tuned by the training samples of satellite images, while other parameters remain the same. After training the networks, the multiscale images are fed into their corresponding networks to extract multiscale features.

C. Feature Fusion

With the extracted multiscale deep features, an intuitive way of integrating these features is to concatenate them into a vector. This method is based on the assumption that all features have the same contribution to the subsequent classification, which obviously is not true in most cases. Besides, the formed high-dimensional feature space not only increases the computational burden but also induces the overfitting problem. MKL has been proved to be an effective method to combine different features for remote sensing image classification [39]–[42]. In this paper, the extracted multiscale deep representations can be considered as different features of an image. Therefore, we employ MKL to integrate these multiscale features.

Assume that the extracted multiscale features for the ith sample are denoted as $\mathbf{X^{(i)}} = \{\mathbf{x_1^{(i)}}, \ldots, \mathbf{x_m^{(i)}}, \ldots, \mathbf{x_M^{(i)}}\}$, where $\mathbf{x_m^{(i)}}$ is a row vector representing the features extracted from the mth SPP-net and M is the total number of SPP-nets. Our goal is to automatically learn the fusion weights $\{d_1, \ldots, d_m, \ldots, d_M\}$, making the generated feature $\hat{\mathbf{X}}^{\mathbf{(i)}} = \{d_1\mathbf{x_1^{(i)}}, \ldots, d_m\mathbf{x_m^{(i)}}, \ldots, d_M\mathbf{x_M^{(i)}}\}$ optimal for the subsequent support vector machine (SVM) classifier. Given N training samples $\{(\hat{\mathbf{X}}^{\mathbf{(i)}}, y^{(i)}), i = 1, 2, \ldots, N\}$ where $y^{(i)} \in \{-1, +1\}$, SVM generally maps each sample $\hat{\mathbf{X}}^{\mathbf{(i)}}$ to a higher dimensional Hilbert space $\mathcal{H}$ using a nonlinear mapping function ϕ, and therein constructs a linear hyperplane $\langle \boldsymbol{\omega}, \phi(\hat{\mathbf{X}})\rangle + b = 0$ where the operator $\langle \cdot, \cdot \rangle$ represents the inner product. The hyperplane is a maximum margin hyperplane, for which the distance from the hyperplane to the closest sample is maximum. It is well known that minimizing the norm of the parameters $1/2\|\boldsymbol{\omega}\|^2$ under the constraint $y^{(i)}(\langle \phi(\hat{\mathbf{X}}), \boldsymbol{\omega}\rangle + b) \geq 1$ maximizes the margin. Such a minimization of the weights provides a naturally regularized solution, which favors smooth models of optimal complexity and avoids overfitting the data. The dual problem of SVM can be written as

$$\begin{aligned} \max \ W(\alpha^{(i)}, \alpha^{(j)}) &= \sum_{i=1}^{N} \alpha^{(i)} - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha^{(i)} \alpha^{(j)} y^{(i)} y^{(j)} \\ &\quad \times \langle \phi(\hat{\mathbf{X}}^{(i)}), \phi(\hat{\mathbf{X}}^{(j)}) \rangle \\ \text{s.t. } 0 \leq \alpha^{(i)} &\leq C, \quad i = 1, 2, \ldots, N, \quad \sum_{i=1}^{N} \alpha^{(i)} y^{(i)} = 0 \end{aligned} \tag{1}$$

where $\alpha^{(i)}$ and $\alpha^{(j)}$ are Lagrange multipliers, and C is a regularization parameter, which determines the tradeoff between the margin and the error on training data.

It is worth noting that directly computing ϕ is nontrivial, but we can calculate the dot product for any two samples in the Hilbert space $\mathcal{H}$ via a kernel trick, which can be expressed as $\langle \phi(\hat{\mathbf{X}}^{\mathbf{(i)}}), \phi(\hat{\mathbf{X}}^{\mathbf{(j)}}) \rangle = K(\hat{\mathbf{X}}^{\mathbf{(i)}}, \hat{\mathbf{X}}^{\mathbf{(j)}})$, where $K(\cdot, \cdot)$ denotes a kernel function. Besides, because of $\hat{\mathbf{X}}^{\mathbf{(i)}} = \{d_1\mathbf{x_1^{(i)}}, \ldots, d_m\mathbf{x_m^{(i)}}, \ldots, d_M\mathbf{x_M^{(i)}}\}$, we have $K(\hat{\mathbf{X}}^{\mathbf{(i)}}, \hat{\mathbf{X}}^{\mathbf{(j)}}) = \sum_{m=1}^{M} d_m^2 K_m(\mathbf{x_m^{(i)}}, \mathbf{x_m^{(j)}})$ [43]. Replacing it in (1), we can get the following objective function:

$$\begin{aligned} \max \ W(\alpha^{(i)}, \alpha^{(j)}, d_m) &= \sum_{i=1}^{N} \alpha^{(i)} - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha^{(i)} \alpha^{(j)} y^{(i)} \\ &\quad \times y^{(j)} \sum_{m=1}^{M} d_m^2 K_m\left(\mathbf{x_m^{(i)}}, \mathbf{x_m^{(j)}}\right) \\ \text{s.t. } 0 \leq \alpha^{(i)} &\leq C, \quad i = 1, 2, \ldots, N, \quad \sum_{i=1}^{N} \alpha^{(i)} y^{(i)} = 0. \end{aligned} \tag{2}$$

Equation (2) can be considered as a typical MKL problem. To simultaneously optimize the fusion weight d_m, and Lagrange multipliers α_i and α_j, we adopt the Simple MKL algorithm, which was first proposed in [44]. Because the objective function in (2) is convex and differentiable, d_m^2 is optimized by using a gradient ascend method. The gradient equals to the derivative of W

$$\frac{\partial W}{\partial d_m^2} = -\frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha^{(i)} \alpha^{(j)} y^{(i)} y^{(j)} K_m\left(\mathbf{x_m^{(i)}}, \mathbf{x_m^{(j)}}\right). \tag{3}$$

Then, d_m^2 is updated as follows:

$$d_m^2 = d_m^2 + \gamma \frac{\partial W}{\partial d_m^2} \tag{4}$$

where γ is the step length. Note that the gradient is updated only when the objective value decreases during the iterative process. This updating procedure is repeated until the stopping criterion is satisfied.

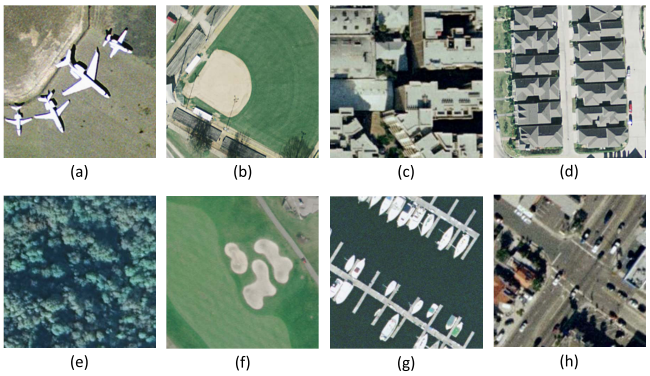


Fig. 5. Some image examples in 21-class land-use data set. (a) Airplane. (b) Baseball diamond. (c) Buildings. (d) Dense residential. (e) Forest. (f) Golf course. (g) Harbor. (h) Intersection.

III. Experiments

To evaluate the effectiveness of the proposed method, we compare it with several state-of-the-art approaches on two widely used data sets: *21-Class Land-Use* [9] data set and *19-Class Satellite Scene* [7], [8] data set.

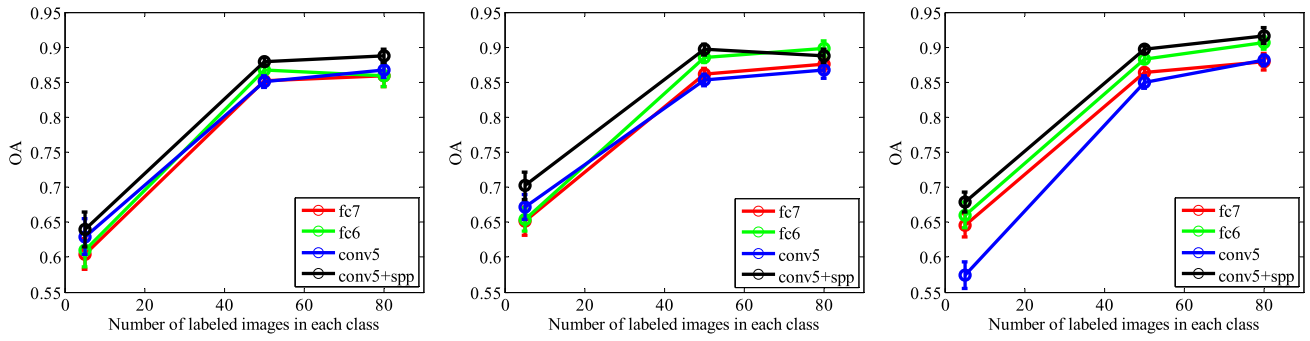


Fig. 6. OAs and standard deviations of SPP-nets at three different scales under different number of training samples on the 21-class land-use data set. (a) 128 × 128 scale. (b) 192 × 192 scale. (c) 256 × 256 scale.

TABLE I

OAs (%) AND STANDARD DEVIATIONS OF SPP-NETS WITH DIFFERENT LAYER FEATURES UNDER DIFFERENT NUMBER OF TRAINING SAMPLES ON 21-CLASS LAND-USE DATA SET

Scales	Number of images	conv5	conv5+spp	fc6	fc7
128 × 128	5	62.93 ± 2.55	$\mathbf{63.96 \pm 2.52}$	60.99 ± 2.37	60.41 ± 2.20
	50	85.12 ± 0.81	$\mathbf{87.98 \pm 0.50}$	86.79 ± 0.64	85.18 ± 0.50
	80	86.81 ± 1.18	$\mathbf{88.81 \pm 0.94}$	85.98 ± 1.65	85.98 ± 1.65
192 × 192	5	67.20 ± 1.81	$\mathbf{70.27 \pm 1.96}$	65.45 ± 1.68	65.14 ± 2.02
	50	85.37 ± 0.82	$\mathbf{89.77 \pm 0.79}$	88.66 ± 0.70	86.25 ± 0.81
	80	86.81 ± 1.18	88.81 ± 0.94	$\mathbf{89.88 \pm 1.16}$	87.64 ± 0.92
256 × 256	5	57.40 ± 1.92	$\mathbf{67.89 \pm 1.44}$	65.99 ± 1.84	64.58 ± 1.64
	50	84.99 ± 0.88	$\mathbf{89.70 \pm 0.52}$	88.35 ± 0.65	86.44 ± 0.52
	80	88.17 ± 0.78	$\mathbf{91.67 \pm 1.11}$	90.62 ± 0.89	87.95 ± 1.15

A. Twenty-One-Class Land-Use Data Set

1) Data Description: This data set was manually extracted from aerial orthoimagery downloaded from the United States Geological Survey National Map. It consists of 21 different land-use and land-cover classes, including *agricultural, airplane, baseball diamond, beach, buildings, chaparral, dense residential, forest, freeway, golf course, harbor, intersection, medium density residential, mobile home park, overpass, parking lot, river, runway, sparse residential, storage tanks*, and *tennis courts*. Each class contains 100 RGB images with a pixel resolution of 1 ft (i.e., 0.3 m) and an image size of 256 × 256 pixels. Fig. 5 shows some image examples from the 21 classes.

2) Experimental Setup: In each experiment, besides the original scale, the images are warped into two different scales, including 128 × 128 and 192 × 192 pixels. For (2), we select linear kernels. For training and testing, the images in each class are randomly split into two sets. In the training stage, we use the training set to fine-tune the SPP-nets and train the linear SVMs, where the SVMs are implemented using the LIBSVM package, and one-against-all strategy is adopted to address the multiclass issue. The performance of the classifiers is then evaluated on the testing set. In order to reduce the effect of random selection, we repeat each algorithm on ten different training/testing splits of the data set and report means and standard deviations of the obtained accuracies.

3) Each Layer Performance: To assess which layer is the best for our task, similar to [45], we analyze and compare the results of the last four feature layers. For simplicity, we name them conv5, conv5+spp, fc6, and fc7. Fig. 6 shows the mean overall accuracies (OAs) and standard deviations using features from different layers at three scales versus different number of training samples. From this figure, we can conclude that the OAs are improved as the number of training samples increases. Besides, fc6 is better than conv5 and fc7 in most cases. However, with the spatial pyramid pooling, conv5+spp improves the performance significantly as compared with conv5, and achieves better results than fc6. Table I demonstrates the detailed quantitative results using 5, 50, and 80 training samples from each class, respectively. The bold fonts indicate the best results with respect to different number of training samples under one scale. In common with Fig. 6, the features from conv5+spp layers achieve the highest accuracies in most of the cases. Thus, we use the features from conv5+spp for the subsequent multiscale feature fusion via MKL.

4) Efficiency of MKL: To examine the performance of our proposed MKL fusion method, we compare it with the single-scale method and the traditional fusion method, i.e., stacking the multiscale deep features into one vector (SV). The classification results using different number of training samples with features from conv5+spp layers are demonstrated in Fig. 7, where conv5+spp-128, conv5+spp-192, and conv5+spp-256 represent that the scales of input images are 128 × 128, 192 × 192, and 256 × 256 pixels, respectively. From this figure, we can observe that the SV method achieves higher classification accuracies than the single-scale features, which can be explained that multiscale deep features represent

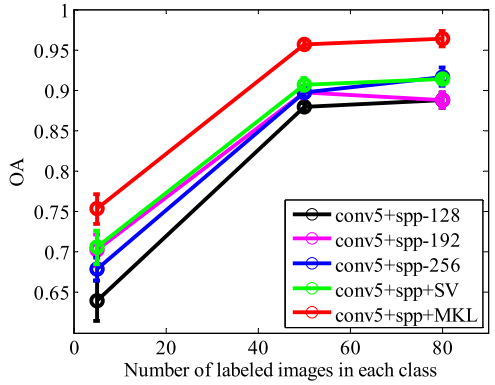


Fig. 7. OAs and standard deviations of MKL versus SV and the single scales using conv5+spp features under different number of training samples on the 21-class land-use data set. Standard deviations are shown as error bars in the vertical direction.

TABLE II

DETAILED CLASSIFICATION RESULTS COMPARISON BETWEEN SINGLE-SCALE FEATURES AND TWO DIFFERENT MULTISCALE FEATURE FUSION METHODS ON THE 21-CLASS LAND-USE DATA SET

Number of images	5	50	80
conv5+spp-128	63.96 ± 2.52	87.98 ± 0.50	88.81 ± 0.94
conv5+spp-192	70.27 ± 1.96	89.77 ± 0.79	88.81 ± 0.94
conv5+spp-256	67.89 ± 1.44	89.70 ± 0.52	91.67 ± 1.11
conv5+spp+SV	70.57 ± 2.06	90.73 ± 0.76	91.38 ± 0.46
conv5+spp+MKL	$\mathbf{75.33 \pm 1.86}$	$\mathbf{95.72 \pm 0.50}$	$\mathbf{96.38 \pm 0.92}$

TABLE III

OVERALL CLASSIFICATION ACCURACY (%) COMPARISON ON THE 21-CLASS LAND-USE DATA SET

Number of images	5	50	80
SSEP [46]	65.34 ± 2.01	–	–
Partlets-based method [6]	–	88.76 ± 0.79	91.33 ± 1.11
SC+Pooling [11]	–	–	81.67 ± 1.23
BOVW [9]	–	–	71.68
SPCK++ [9]	–	–	77.38
SPMK [18]	–	–	74.00
MKL [47]	64.78 ± 1.62	88.68 ± 1.10	91.26 ± 1.17
UFL [48]	–	–	90.26 ± 1.51
DCNN [25]	–	82.29 ± 1.54	85.71 ± 1.23
GBRCN [33]	–	–	94.53
SPP-net	70.27 ± 1.96	89.77 ± 0.79	91.67 ± 1.11
SPP-net+SV	70.57 ± 2.06	90.73 ± 0.76	91.38 ± 0.46
SPP-net+MKL	$\mathbf{75.33 \pm 1.86}$	$\mathbf{95.72 \pm 0.50}$	$\mathbf{96.38 \pm 0.92}$

different abstracts of the original images and simultaneously using these complementary information thereby improves the classification results. Another obvious observation is that the MKL method significantly boosts the classification results as compared with SV. The reason can be attributed to the fact that MKL automatically learns the optimal combination among multiscale deep features, while SV simply assumes that the features in all scales play the same role. The quantitative results in Table II support the conclusions in Fig. 7, which further confirms the efficiency of our proposed fusion method.

5) Comparison With the State of the Arts: To demonstrate the superiority of the proposed method, we compare with several state-of-the-art approaches, including DCNN [25], gradient boosting random convolutional network (GBRCN) [33], semisupervised ensemble projection (SSEP) [46], Partlets-based method [6], SC+Pooling [11], BOVW [9], extended spatial pyramid co-occurrence kernel (SPCK++) [9], spatial pyramid match kernel (SPMK) [18],

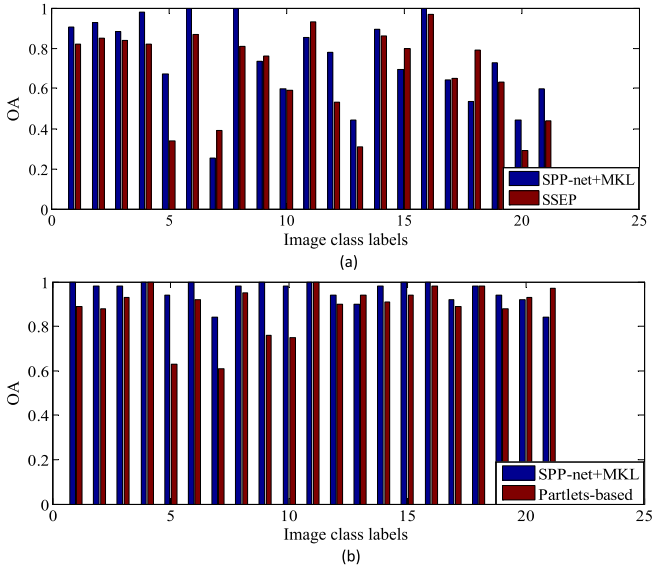


Fig. 8. Each class accuracy comparison between two methods on the 21-class land-use data set. (a) SSEP in [46] and SPP-net+MKL using five training samples. (b) Partlets-based method in [6] and SPP-net+MKL using 50 training samples.

MKL [47], and unsupervised feature learning (UFL) [48]. The classification results with different number of training samples are shown in Table III, where "–" denotes that there are no experiments. From this table, we can observe that SPP-net with the best single-scale feature achieves higher accuracies than most of comparison methods. This implies that the deep learning method learns more powerful features. Besides, the combination of multiscale deep features further improves the performance. Specifically, SPP-net+MKL boosts the performance dramatically by 15%, 8%, and 2% in comparison with the existing best results when the number of training samples is 5, 50, and 80, respectively. To the best of our knowledge, these results are the best on this data set, which adequately show the superiority of our proposed method. It is worth noting that SPP-net+MKL obtains higher OA (i.e., 96.38%) than GBRCN whose OA is 94.53% when we use 80 samples from each class as the training set. This indicates that sufficiently capturing the multiscale information in satellite images can improve the classification performance. In addition, we also compare SPP-net+MKL with two recently proposed state-of-the-art approaches by evaluating the accuracy in each class, which is shown in Fig. 8. From Fig. 8(a), we observe that SSEP gets a little better performance than SPP-net+MKL in six classes. This is because the SSEP method takes advantage of the sampling technique to indirectly increase the number of training samples, while SPP-net+MKL only uses the given training samples. Nevertheless, SPP-net+MKL achieves higher accuracies in the rest of 15 classes. Similarly, Fig. 8(b) demonstrates that SPP-net+MKL obtains higher performance in 19 classes compared with the Partlets-based method in [6]. For further analysis of the classification result achieved by SPP-net+MKL, we use the confusion matrix shown in Fig. 9 to illustrate one of the results in ten experiments when the number of training samples is five. The ith row and jth column element in the confusion matrix denotes the rate of test

TABLE IV

OAs (%) AND STANDARD DEVIATIONS OF SPP-NETS WITH DIFFERENT LAYER FEATURES UNDER DIFFERENT NUMBER OF TRAINING SAMPLES ON THE 19-CLASS SATELLITE SCENE DATA SET

Scales	Number of images	conv5	conv5+spp	fc6	fc7
128×128	5	73.01 ± 1.56	$\mathbf{76.56 \pm 0.92}$	75.82 ± 1.03	75.39 ± 1.12
	25	85.22 ± 0.97	$\mathbf{87.26 \pm 1.00}$	87.16 ± 0.89	87.16 ± 0.89
192×192	5	75.25 ± 1.10	$\mathbf{80.34 \pm 0.80}$	78.69 ± 0.92	77.60 ± 0.85
	25	87.68 ± 1.08	$\mathbf{90.13 \pm 0.90}$	89.66 ± 0.77	89.12 ± 0.85
256×256	5	74.80 ± 1.31	$\mathbf{80.46 \pm 1.47}$	78.84 ± 0.80	77.53 ± 1.25
	25	87.54 ± 1.25	$\mathbf{90.27 \pm 0.64}$	89.57 ± 0.76	88.94 ± 0.78

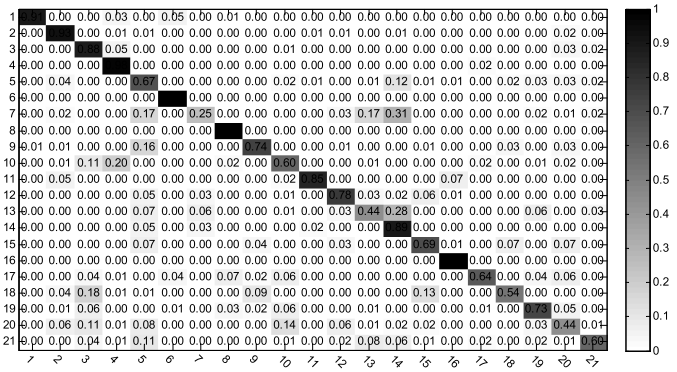


Fig. 9. Confusion matrix of SPP-net+MKL with five training samples in each class on the 21-class satellite scene data set. The rows and columns of the matrix denote actual and predicted classes, respectively. The class labels are as follows. 1: Agricultural. 2: Airplane. 3: Baseball diamond. 4: Beach. 5: Buildings. 6: Chaparral. 7: Dense residential. 8: Forest. 9: Freeway. 10: Golf course. 11: Harbor. 12: Intersection. 13: Medium residential. 14: Mobile home park. 15: Overpass. 16: Parking lot. 17: River. 18: Runway. 19: Sparse residential. 20: Storage tanks. 21: Tennis court.

Fig. 10. Some image examples of (first row) *dense residential*, (second row) *medium residential*, and (third row) *mobile home park*.

samples from the ith class classified to the jth class. It can be observed that the most difficult classes to discriminate contain *dense residential, Runway, medium residential*, and *storage tanks*, whose accuracies are all lower than 60%. For instance, as shown in Fig. 10, some images in the *dense residential* class have similar structures to those in the *medium residential* class and the *mobile home park* class. Therefore, these images in *dense residential* are easily misclassified as *medium residential* and *mobile home park*. However, when the number of training samples increases to 50, the accuracies of these classes improve significantly. This indicates that the number of training samples is a key factor for SPP-net+MKL.

B. Nineteen-Class Satellite Scene Data Set

1) Data Description and Experimental Setup: The second data set is composed of 19 classes of scenes, including *airport, beach, bridge, commercial area, desert, farmland, football field, forest, industrial area, meadow, mountain, park, parking, pond, port, railway station, residential area, river*, and *viaduct*. Each class has 50 images, with a size of 600×600 pixels. Such images are extracted from very large satellite images on Google Earth. Similar to the 21-class land-use data set, the original images are warped to three different scales: 128×128, 192×192, and 256×256. We construct two experiments. The first one is randomly choosing five images from each class as the training set, and the rest of the images are used as the testing set, following [10], [46], and [47]. The second experiment randomly chooses 25 images as the training set and the remaining as the testing set, following [10]. All the experiments are repeated ten times with different training/testing splits to obtain stable results. The final performance is reported as the mean and standard deviation of the results from ten runs.

TABLE V

DETAILED CLASSIFICATION RESULTS COMPARISON BETWEEN SINGLE SCALE FEATURES AND TWO DIFFERENT MULTISCALE FEATURE FUSION METHODS ON THE 19-CLASS SATELLITE SCENE DATA SET

Number of images	5	25
conv5+spp-128	76.56 ± 0.92	87.26 ± 1.00
conv5+spp-192	80.34 ± 0.80	90.13 ± 0.90
conv5+spp-256	80.46 ± 1.47	90.27 ± 0.64
conv5+spp+SV	80.92 ± 1.16	90.48 ± 0.87
conv5+spp+MKL	$\mathbf{85.22 \pm 1.22}$	$\mathbf{95.07 \pm 0.79}$

2) Each Layer Performance: Similar to the first data set, we evaluate the effect of different feature layers on the final performance. Fig. 11 shows the classification results at three different scales using conv5, conv5+spp, fc6, and fc7 features. From this figure, we can observe that conv5+spp achieves the highest OAs as well as in the first data set compared with the other three features, which is also demonstrated in Table IV. Besides, we observe that the OAs on this data set are higher than that on the first data set under the same number of training samples. This is because this data set is easier to discriminate and the number of testing sets is smaller than that in the first data set. Fig. 12 and Table V compare the performance between single-scale conv5+spp features and two multiscale fusion methods. Obviously, the performance of SV is only a little better than that of 192×192 and 256×256 scales. However, MKL displays significant improvements in comparison with SV, which confirms the effectiveness of the MKL fusion method.

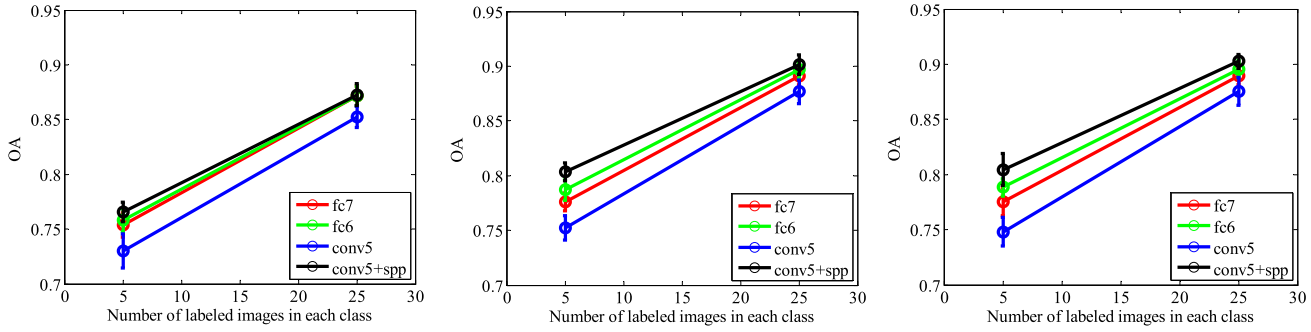


Fig. 11. OAs and standard deviations of SPP-nets at three different scales under different number of training samples on the 19-class satellite scene data set. (a) 128 × 128 scale. (b) 192 × 192 scale. (c) 256 × 256 scale. Standard deviations are shown as error bars in the vertical direction.

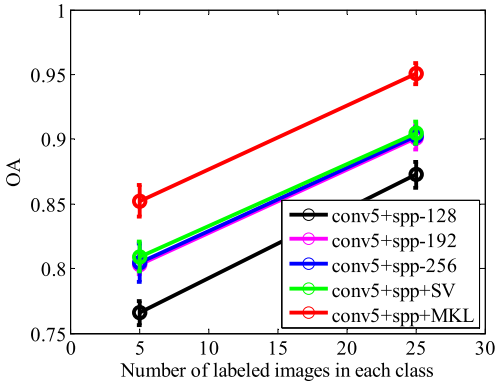


Fig. 12. OAs and standard deviations of MKL versus SV and the single scales using conv5+spp features under different number of training samples on the 19-class satellite scene data set. Standard deviations are shown as error bars in the vertical direction.

TABLE VI
OVERALL CLASSIFICATION ACCURACY (%) COMPARISON ON THE 19-CLASS SATELLITE SCENE DATA SET

Number of images	5	25
SSEP [46]	73.82 ± 1.52	–
SCMF [10]	78.32	90.05
MKL [47]	67.32 ± 2.90	–
SPP-net	80.46 ± 1.47	90.27 ± 0.64
SPP-net+SV	80.92 ± 1.16	90.48 ± 0.87
SPP-net+MKL	$\mathbf{85.22 \pm 1.22}$	$\mathbf{95.07 \pm 0.79}$

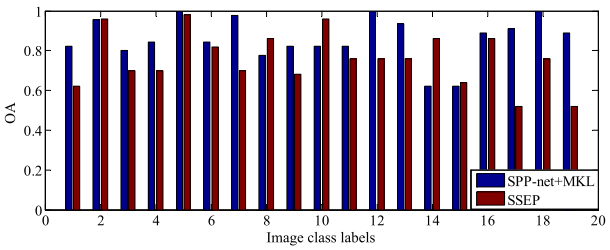


Fig. 13. Each class accuracy comparison between SPP-net+MKL and SSEP in [46] using five training samples.

3) Comparison With the State of the Arts: In order to comprehensively analyze the superiority of the proposed method, we compare it with three state-of-the-art approaches ever tested on this data set. They are SSEP [46], sparse codes of multiple features (SCMF) [10], and MKL [47]. The comparison results are illustrated in Table VI, from which we can observe that the proposed SPP-net+MKL significantly improves the accuracy from 78.32 to 85.22 and 90.05 to 95.07 when the numbers of training samples are 5 and 25, respectively. Besides, we compare each class accuracy with the latest approach SSEP in [46]. SPP-net+MKL achieves higher accuracies in 14 classes, as shown in Fig. 13.

IV. CONCLUSION

This paper proposed to automatically extract multiscale deep features from the satellite images by using SPP-net. This net comprises five convolutional layers and two fully connected layers, where the last convolutional layer is followed by the spatial pyramid pooling operator. It is well known that the performance of deep models heavily depends on the large number of training samples, while only hundreds of samples are available in most of the satellite image classification cases. Therefore, we focused on solving the problem of training multiple effective SPP-nets simultaneously. To this end, we pretrained the DCNN model by using the auxiliary ImageNet data set, which is different from satellite images, and then transferred the parameters in the five convolutional layers to the SPP-nets. Finally, the fully connected layers of each SPP-net were fine-tuned by their corresponding training samples. It is of great interest to see that this training approach leads to very promising classification results that outperform most of the existing results on the same data sets. Furthermore, a MKL method was adopted to fuse the multiscale deep features. The experiments on two classical satellite data sets have demonstrated that the proposed method dramatically improves the classification results compared with several state of the arts.

REFERENCES

[1] 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.
[2] A. Plaza *et al.*, "Recent advances in techniques for hyperspectral image processing," *Remote Sens. Environ.*, vol. 113, pp. S110–S122, Sep. 2009.
[3] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, "Advances in spectral-spatial classification of hyperspectral images," *Proc. IEEE*, vol. 101, no. 3, pp. 652–675, Mar. 2013.
[4] R. Hang, Q. Liu, H. Song, and Y. Sun, "Matrix-based discriminant subspace ensemble for hyperspectral image spatial–spectral feature fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 2, pp. 783–794, Feb. 2016.
[5] Q. Liu, Y. Sun, R. Hang, and H. Song, "Spatial–spectral locality-constrained low-rank representation with semi-supervised hypergraph learning for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 9, pp. 4171–4182, Sep. 2017.

[6] G. Cheng, J. Han, L. Guo, Z. Liu, S. Bu, and J. Ren, "Effective and efficient midlevel visual elements-oriented land-use classification using VHR remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4238–4249, Aug. 2015.

[7] D. Dai and W. Yang, "Satellite image classification via two-layer sparse coding with biased image representation," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 1, pp. 173–176, Jan. 2011.

[8] G.-S. Xia, W. Yang, J. Delon, Y. Gousseau, H. Sun, and H. Maître, "Structural high-resolution satellite image indexing," in *Proc. 7th ISPRS TC Symp. 100 Years*, vol. 38. 2010, pp. 298–303.

[9] Y. Yang and S. Newsam, "Spatial pyramid co-occurrence for image classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1465–1472.

[10] G. Sheng, W. Yang, T. Xu, and H. Sun, "High-resolution satellite scene classification using a sparse coding based multiple feature combination," *Int. J. Remote Sens.*, vol. 33, no. 8, pp. 2395–2412, 2012.

[11] A. M. Cheriyadat, "Unsupervised feature learning for aerial scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 1, pp. 439–451, Jan. 2014.

[12] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Oct. 2003, pp. 1470–1477.

[13] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. Eur. Conf. Comput. Vis.*, vol. 1. 2004, pp. 1–2.

[14] L. Fei-Fei and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2. Jun. 2005, pp. 524–531.

[15] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[16] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1. Jun. 2005, pp. 886–893.

[17] Y. Yang and S. Newsam, "Comparing SIFT descriptors and Gabor texture features for classification of remote sensed imagery," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2008, pp. 1852–1855.

[18] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2. 2006, pp. 2169–2178.

[19] Y. Jiang, J. Yuan, and G. Yu, "Randomized spatial partition for scene recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 730–743.

[20] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proc. IEEE*, vol. 98, no. 6, pp. 1031–1044, Jun. 2010.

[21] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning mid-level features for recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2559–2566.

[22] K. Yu, Y. Lin, and J. Lafferty, "Learning image representations from the pixel level via hierarchical sparse coding," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 1713–1720.

[23] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[24] L. Deng, "A tutorial survey of architectures, algorithms, and applications for deep learning," *APSIPA Trans. Signal Inf. Process.*, vol. 3, p. e2, Jan. 2014.

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

[26] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated recognition, localization and detection using convolutional networks," presented at the Int. Conf. Learn. Represent. (ICLR), Banff, AB, Canada, Apr. 2014.

[27] A. G. Howard. "Some improvements on deep convolutional neural network based image classification." Unpublished paper, 2013. [Online]. Available: https://arxiv.org/abs/1312.5402

[28] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 818–833.

[29] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. "Return of the devil in the details: Delving deep into convolutional nets." Unpublished paper, 2014. [Online]. Available: https://arxiv.org/abs/1405.3531

[30] K. Simonyan and A. Zisserman. "Very deep convolutional networks for large-scale image recognition." Unpublished paper, 2014. [Online]. Available: https://arxiv.org/abs/1409.1556

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 346–361.

[32] K. Makantasis, K. Karantzalos, A. Doulamis, and N. Doulamis, "Deep supervised learning for hyperspectral data classification through convolutional neural networks," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2015, pp. 4959–4962.

[33] F. Zhang, B. Du, and L. Zhang, "Scene classification via a gradient boosting random convolutional network framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 3, pp. 1793–1802, Mar. 2016.

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

[35] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[36] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1717–1724.

[37] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. ACM Int. Conf. Multimedia*, 2014, pp. 675–678.

[38] Y. LeCun *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989.

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

[40] Y. Gu, C. Wang, D. You, Y. Zhang, S. Wang, and Y. Zhang, "Representative multiple kernel learning for classification in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 7, pp. 2852–2865, Jul. 2012.

[41] C. Zhao, X. Gao, Y. Wang, and J. Li, "Efficient multiple-feature learning-based hyperspectral image classification with limited training samples," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 7, pp. 4052–4062, Jul. 2016.

[42] 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.

[43] M. Gönen and E. Alpaydın, "Multiple kernel learning algorithms," *J. Mach. Learn. Res.*, vol. 12, pp. 2211–2268, Jul. 2011.

[44] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet, "SimpleMKL," *J. Mach. Learn. Res.*, vol. 9, pp. 2491–2521, Nov. 2008.

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

[46] W. Yang, X. Yin, and G. S. Xia, "Learning high-level features for satellite image classification with limited labeled samples," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4472–4482, Aug. 2015.

[47] C. Cusano, P. Napoletano, and R. Schettini. "Remote sensing image classification exploiting multiple kernel learning." Unpublished paper, 2014. [Online]. Available: https://arxiv.org/abs/1410.5358

[48] F. Hu, G.-S. Xia, Z. Wang, L. Zhang, and H. Sun, "Unsupervised feature coding on local patch manifold for satellite image scene classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2014, pp. 1273–1276.

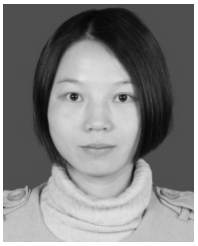
Qingshan Liu (M'05–SM'07) received the M.S. degree from Southeast University, Nanjing, China, in 2000, and the Ph.D. degree from the Chinese Academy of Sciences, Beijing, China, in 2003.

He was an Associate Professor with the National Laboratory of Pattern Recognition, Chinese Academy of Sciences. From 2010 to 2011, he was an Assistant Research Professor with the Department of Computer Science, Computational Biomedicine Imaging and Modeling Center, Rutgers University The State University of New Jersey, Piscataway, NJ, USA. From 2004 and 2005, he was an Associate Researcher with the Multimedia Laboratory, The Chinese University of Hong Kong, Hong Kong. He is currently a Professor with the School of Information and Control, Nanjing University of Information Science and Technology, Nanjing. His research interests include image and vision analysis.

Dr. Liu received the President Scholarship of the Chinese Academy of Sciences in 2003.

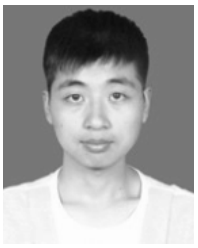
Renlong Hang (M'17) received the Ph.D. degree in meteorological information technology from the Nanjing University of Information Science and Technology, Nanjing, China, in 2017.

He is currently a Lecturer with the School of Information and Control, Nanjing University of Information Science and Technology. His research interests include machine learning and pattern recognition.



Huihui Song received the M.S. degree in communication and information system from the University of Science and Technology of China, Hefei, China, in 2011, and the Ph.D. degree in geography and resource management from The Chinese University of Hong Kong, Hong Kong, in 2014.

She is currently a Professor with the Jiangsu Key Laboratory of Big Data Analysis Technology, Nanjing University of Information Science and Technology, Nanjing, China. Her research interests include remote sensing image processing and image fusion.



Zhi Li received the M.S. degree in system science from the Nanjing University of Information Science and Technology, Nanjing, China, in 2016.

His research interests include deep learning and pattern recognition.