Skip-Connected Covariance Network for Remote Sensing Scene Classification

Nanjun He, *Student Member, IEEE*, Leyuan Fang, *Senior Member, IEEE*, Shutao Li, *Fellow, IEEE*, Javier Plaza, *Senior Member, IEEE*, and Antonio Plaza, *Fellow, IEEE*

***Abstract*— This paper proposes a novel end-to-end learning model, called skip-connected covariance (SCCov) network, for remote sensing scene classification (RSSC). The innovative contribution of this paper is to embed two novel modules into the traditional convolutional neural network (CNN) model, i.e., skip connections and covariance pooling. The advantages of newly developed SCCov are twofold. First, by means of the skip connections, the multi-resolution feature maps produced by the CNN are combined together, which provides important benefits to address the presence of large-scale variance in RSSC data sets. Second, by using covariance pooling, we can fully exploit the second-order information contained in such multi-resolution feature maps. This allows the CNN to achieve more representative feature learning when dealing with RSSC problems. Experimental results, conducted using three large-scale benchmark data sets, demonstrate that our newly proposed SCCov network exhibits very competitive or superior classification performance when compared with the current state-of-the-art RSSC techniques, using a much lower amount of parameters. Specifically, our SCCov only needs 10% of the parameters used by its counterparts.**

***Index Terms*— Covariance pooling, deep neural network, multi-layer feature, scene recognition.**

I. Introduction

REMOTE sensing scene classification (RSSC) has recently gathered considerable attention as it can be adopted in many practical applications, such as urban mapping and land-use classification [1]. Given a query image, the goal of the RSSC is to assign a unique label (e.g., airport, forest, and so on) to the image, based on its contents. Due to the variance of the distance between the sensor and the earth, RSSC often encounters the problem of large-scale variance (LSV) [2]–[4]. This is related to the fact that within the same scene category, the images can present very different scales (some examples illustrating this problem are given in Fig. 1). This makes RSSC become a very challenging problem.

Over the past decade, we have witnessed the renew of neural networks in the computer vision community, most notably in tasks, such as image classification and object detection [5], [6], face recognition [7], and scene recognition [8]. In addition, due to their excellent performance, deep neural networks have also been widely used in the remote sensing community, particularly in applications such as change detection [9]–[13], image super-resolution [14], hyperspectral image classification [15]–[19], high-resolution image classification [20], and radar image classification [21], among several others. The basic idea behind deep neural networks, such as the well-known convolutional neural network (CNN), is to represent the image with a deep hierarchical architecture (e.g., the Alexnet [22] and VGG16 [23]). By doing so, the deep neural network, especially the CNN models, can naturally extract feature maps with multi-resolution and pyramidal shape across different layers, which can be then utilized to address the LSV problem in image classification and object detection [24].

In this paper, we specifically tackle the LSV problem in RSSC by taking the characteristics of the CNN architecture into account. Specifically, we embed a skip-connection module into off-the-shelf CNN models, e.g., Alexnet and VGG16, to concatenate multi-resolution feature maps for classification purposes. In addition, a covariance pooling strategy is utilized to aggregate the concatenated multi-resolution feature maps from different layers. Compared to traditional max or average pooling strategies, which only use first-order statistics (i.e., max or mean) to integrate the feature maps, the covariance pooling offers the possibility to use the second-order statistics information (i.e., covariance) to pool the feature maps. As a result, more representative features can be learned. In order to demonstrate the effectiveness of our contribution, comprehensive experiments are presented in Section IV to demonstrate the aforementioned aspects. Resulting from our newly proposed methodology, a new end-to-end learning model called skip-connected covariance (SCCov) network is presented and discussed. Moreover, we also visualize the saliency map obtained by the SCCov network to investigate its performance

Manuscript received November 6, 2018; revised March 7, 2019; accepted May 24, 2019. Date of publication July 11, 2019; date of current version May 1, 2020. This work was supported in part by the National Natural Science Fund of China under Grant 61890962, Grant 61520106001, and Grant 61771192, in part by the Science and Technology Plan Project Fund of Hunan Province under Grant CX2018B171, Grant 2017RS3024, and Grant 2018TP1013, in part by the Science and Technology Talents Program of Hunan Association for Science and Technology under Grant 2017TJ-Q09, and in part by the National Key R&D Program of China under Grant 2018YFB1305200. *(Corresponding author: Leyuan Fang.)*

N. He, L. Fang, and S. Li are with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China, and also with the Key Laboratory of Visual Perception and Artificial Intelligence of Hunan Province, Hunan University, Changsha 410082, China (e-mail: henanjun@hnu.edu.cn; fangleyuan@gmail.com; shutao_li@hnu.edu.cn).

J. Plaza and A. Plaza are with the Hyperspectral Computing Laboratory, Department of Technology of Computers and Communications, Escuela Politécnica, University of Extremadura, E-10003 Cáceres, Spain (e-mail: jplaza@unex.es; aplaza@unex.es).

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

Digital Object Identifier 10.1109/TNNLS.2019.2920374

Fig. 1. Example to illustrate the problem of LSV in RSSC. From top to bottom, we can observe the existing scale variability between objects belonging to the same class in the following classes: *airplane* (top row), *storage tank* (second row), *harbor* (third row), and *bridge* (bottom row), respectively.

in RSSC applications. The main innovative contributions of this paper can be summarized as follows.

1) We develop a new end-to-end learning model that embeds two new modules into the CNN model for RSSC purposes. The proposed approach exhibits competitive or superior classification performance when compared with current state-of-the-art methods, using a much less amount of training parameters.
2) We investigate how our newly proposed SCCov network performs in RSSC problems by visualizing the saliency map of the test image, which gives us important insights about the working mechanism of the SCCov network model.

The remainder of this paper is organized as follows. Section II discusses some related works and highlights the innovative contributions of our method. Section III details the architecture of the proposed SCCov network. In Section IV, a comprehensive experimental assessment of the proposed methodology (in comparison with other state-of-the-art methods) is conducted on three widely used data sets. Section V concludes this paper with some remarks and hints at plausible future research lines.

II. Related Works and Novelty of Our Method

A. Related Works

During the past several years, a considerable number of approaches have been proposed for RSSC. Generally, these methods can be categorized into three main classes: hand-crafted feature-based methods, feature learning-based methods, and end-to-end learning systems.

1) Hand-Crafted Feature-Based Methods: Hand-crafted feature-based methods usually consist of the following three steps: 1) feature extraction; 2) feature encoding; and 3) classifier training. In the first step, a classical hand-crafted feature descriptor, such as the scale-invariant feature transform (SIFT) [25] or histogram of gradient (HoG) [26], is used to extract features that represent the images, and then, the obtained features are aggregated by some feature encoding methods, such as bag of visual words (BoVW), improved Fisher vector (IFK), sparse coding, or probability topic models. Finally, the encoded features are used to train a classifier [e.g., the support vector machine (SVM)] for scene recognition. Some related methods can be found in [2] and [27]–[34].

2) Feature Learning-Based Methods: Feature learning-based methods usually adopt a similar procedure as hand-crafted feature-based methods. However, instead of using hand-crafted features (e.g., SIFT), feature learning-based methods utilize some representation-based learning approaches for feature extraction. Specifically, Zhang *et al.* [35] used a sparse autoencoder for unsupervised feature learning on saliency image patches. In [36], a shallow weighted deconvolution network is utilized for feature extraction by minimizing the Euclidean distance between the original and the reconstructed image. Hu *et al.* [37] utilized spectral clustering to discover intrinsic structures among image patches for feature learning. Recently, due to the powerful generalization ability exhibited by CNN models [38], [39], CNN models that have been pre-trained on ImageNet [40] have widely been used as feature extractors for RSSC. Hu *et al.* [41] investigated different CNN models as feature extractors and integrate them with various feature encoding methods for RSSC purposes. Their results show that using the CNN model as a feature extractor, usually results in better performance than that provided by hand-crafted feature-based methods. Cheng *et al.* [42] utilized the BoVW model to aggregate the convolutional activation layer. In [43], the last two fully connected (FC) layers of a CNN model are combined together to represent the image. In [44], a multi-scale IFK coding method is proposed to integrate the feature maps from different layers. He *et al.* [45] adopted a simple yet effective method (i.e., covariance pooling) to combine the different layers of pre-trained CNN models for RSSC. In [46], a feature ensemble framework is proposed to combine hand-crafted features and features extracted by a pre-trained CNN model.

3) End-to-End Learning Systems: In general, the methods in the two previously discussed categories exhibit a satisfactory classification performance. However, these methods are made up of several separated steps, and thus, a large storage space is needed to store the intermediate results (features). This limits their potential application in practice. Under this context, the development of end-to-end systems represents a promising direction for RSSC. Castelluccio *et al.* [47] fine-tuned two classical pre-trained CNN models (i.e., Caffenet and GoogLenet) for RSSC purposes. Cheng *et al.* [48] added a new item into the loss function of the aforementioned pre-trained CNN model to minimize the intra-class distance and maximize the inter-class distance, thus improving the classification performance. However, this method [48] needs to measure the distance between different images, and thus, the image pairs need to be selected manually as the input of the CNN,

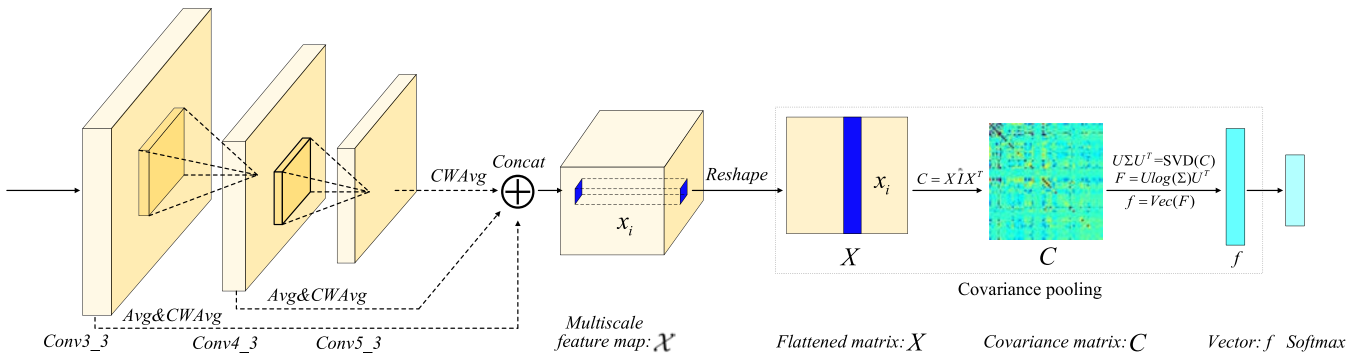


Fig. 2. Architecture of the proposed SCCov network. The off-the-shelf VGG16 is used as the backbone for illustration purposes. The feature maps from different layers are combined together by the skip connection operation, followed by a covariance pooling strategy. *Concat* denotes the concatenate operation, SVD denotes the singular value decomposition, and *Vec* stands for the vectorization operation. *Avg* denotes average pooling and CWAvg stands for channel-wise average pooling (see Section III and Table I for additional notations and details).

which is quite consuming from a computational standpoint. In [49], a multi-scale CNN model is presented to address the LSV problem in RSSC. Anwer *et al.* [50] extracted a classical feature descriptor, i.e., local binary pattern (LBP), as the input of the CNN model. By considering object-level information, the region proposal network [51] is added to the CNN model in [52] to enhance the classification performance in an RSSC context. In [53], a deep structural metric learning model that can explore the structural information among training samples is presented and discussed in the context of RSSC applications.

B. High-Order Pooling

Recently, the exploitation of high-order information in deep neural networks has become a hot topic in the computer vision community, since the traditional CNN models only take the first-order information into consideration. In [54], a bilinear pooling network was first proposed for fine-grained classification, which achieved state-of-the-art classification performance. Li *et. al.* [55] further investigated the utilization of second-order pooling for the classification of large-scale image data sets. Moreover, considering the possible complementarities between the first-order features (i.e., those obtained by average pooling) and the second-order features (i.e., those obtained by high-order pooling), some works proposed to combine these two kinds of features. Specifically, in [56], the first-order information obtained by average pooling and the second-order information obtained by a bilinear model [54] are combined together by a concatenation operation. In addition, in [57], a Gaussian embedding strategy was applied to fuse the first-order and the second-order information.

C. Novelty of the Proposed Method

The proposed SCCov network belongs to the third discussed category, i.e., end-to-end learning systems. Compared to hand-crafted feature-based methods or feature learning-based methods, our method can be trained in an end-to-end fashion, thus enhancing the obtained classification performance. Compared to the methods in [48], [50], and [52], the proposed method does not need to perform image preprocessing (e.g., searching image pairs or feature extraction) and also shows better classification performance than the methods in [48]–[50] and [52], as well as competitive classification performance when compared to the method in [48]. An important characteristic of our method is that it needs a much lower amount of training parameters. Specifically, our SCCov network needs only 10% of the parameters required by its counterparts. This is an important innovative aspect, since the very reduced number of parameters required by our proposed approach is more likely to avoid the problem of overfitting when training a deep CNN model on a relatively small data set.

III. Proposed Learning Network

Fig. 2 shows the architecture of the proposed SCCov network using VGG16 as the backbone. Specifically, three convolution layers: "conv3-3," "conv4-3," and "conv5-3" are concatenated by means of skip connections. If the obtained multi-resolution feature maps are denoted by $\mathcal{X}$, we can observe that the $\mathcal{X}$ volume is reshaped into a matrix along the feature maps' channel dimension. Then, a covariance pooling layer is used to aggregate the obtained multi-resolution feature maps. Finally, this layer is followed by an FC layer and a softmax layer. In the following, we elaborate the newly added modules, i.e., the skip connections and the covariance pooling.

A. Skip Connections for Multi-Layer Aggregation

Let us assume that three sets of feature maps with the same spatial resolution are available, i.e., $\mathcal{X}_1 \in \mathbb{R}^{H\times W\times D_1}$, $\mathcal{X}_2 \in \mathbb{R}^{H\times W\times D_2}$, and $\mathcal{X}_3 \in \mathbb{R}^{H\times W\times D_3}$. In this case, the aggregated multi-resolution feature map $\mathcal{X}$ can be obtained by means of a skip connections strategy as follows:

$$\mathcal{X} = [\mathcal{X}_1; \mathcal{X}_2; \mathcal{X}_3] \in \mathbb{R}^{H\times W\times (D_1+D_2+D_3)} \tag{1}$$

where [; , ; , ;] denotes the concatenation operation along the third dimension. An illustration of a skip connection strategy for three feature maps is shown in Fig. 3. The motivation of using skip connections to aggregate multi-layer feature maps is twofold. First, as pointed in [5] and [24], the CNN model can naturally extract feature maps with pyramidal shape by means of hierarchical layers, which addresses the problem of scale variance in classification and object detection tasks.

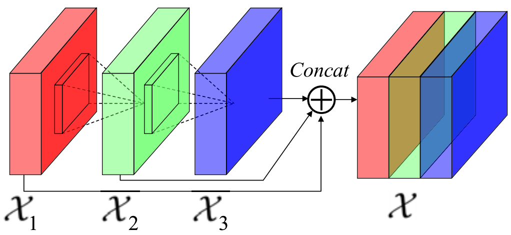


Fig. 3. Graphical example illustrating the skip connections for three feature maps. Concat refers to a concatenation operation.

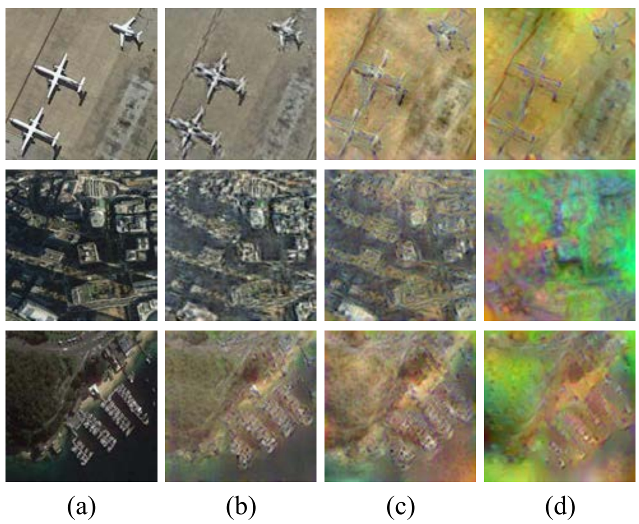


Fig. 4. Graphical example illustrating the feature maps extracted from different layers of Alexnet for three different images. (a) Input image. (b) Feature map from the third convolutional layer. (c) Feature map from the fourth convolutional layer. (d) Feature map from the fifth convolutional layer.

Second, the feature maps from different layers contain complementary information [15], [41], [43], [45]. For illustrative purposes, Fig. 4 shows an example using the feature maps extracted from different layers of Alexnet. As can be seen, such feature maps exhibit various characteristics and therefore provide complementary information that can be exploited by the skip connection operation to improve the classification performance.

Note that, in order to concatenate feature maps with different spatial resolutions, average pooling is adopted. In addition, CWAvg pooling is also adopted to reduce the number of channels of each set of feature maps, before concatenating them together. The mathematical definition of the CWAvg pooling is detailed as follows. Given a 3-D feature map tensor $\mathcal{Y} = [Y_1; Y_2; \ldots; Y_L] \in \mathbb{R}^{H\times W\times L}$, where $Y_i \in \mathbb{R}^{H\times W}$ is a single feature map, and assuming stride k, the CWAvg pooling is conducted as follows:

$$Z_j = \frac{1}{k}\sum_{i=(j-1)\times k+1}^{j\times k} Y_i, \quad j = 1, 2, \ldots, L/k. \tag{2}$$

As a result, the output feature map tensor $\mathcal{Z} = [Z_1, Z_2, \ldots, Z_{L/k}] \in \mathbb{R}^{H\times W\times(L/k)}$ is obtained. In practice, we choose the value of k to make sure that L is divisible by k. In other words, L/k is an integer.

B. Forward Propagation of Covariance Pooling

Given a feature matrix $X \in \mathbb{R}^{D\times N}$ (e.g., the matrix X in Fig. 2) with each row being normalized by the l_2-norm, where $D = D_1 + D_2 + D_3$ is the dimensionality of the features and $N = H \times W$ is the number of features, the forward propagation of covariance pooling is conducted as follows. First, a covariance matrix C is calculated

$$X \mapsto C, \quad C = X\hat{I}X^T \tag{3}$$

where $\hat{I} = (1/N-1)(I-(1/N)\mathbf{1}\mathbf{1}^T)$, I is an $N\times N$ identity matrix, and $\mathbf{1}$ is an N-dimensional column vector with all entries set to 1. Then, the matrix logarithm is used to transform the covariance matrix from a manifold space to Euclidean space in order to obtain the pooled feature F [58]

$$C \mapsto F, F = U\log(\Sigma)U^T \tag{4}$$

where $C = U\Sigma U^T$, and U and Σ are the eigenvector matrix and eigenvalue matrix of C. In Fig. 2, f is the vectorization of F. Note that, F is symmetric matrix, and therefore, only the entries in the upper triangle of F need to be vectorized, i.e., the dimensionality of vector f is $D(D+1)/2$.

C. Backward Propagation of Covariance Pooling

Different from the traditional max or average pooling strategies, which process the spatial coordinates of the intermediate variable (a matrix or a vector) independently, covariance pooling is based on global and structured matrix computations. Here, we adopt the matrix back-propagation methodology formulated in [59] to compute the partial derivative of loss function L with respect to the input matrix of covariance pooling. Given the partial derivative propagated from the upper FC layer, $(\partial L/\partial F)$, we first consider $(\partial L/\partial U)$ and $(\partial L/\partial\Sigma)$. The chain rule expression is shown in the following:

$$\frac{\partial L}{\partial U} : dU + \frac{\partial L}{\partial \Sigma} : d\Sigma = \frac{\partial L}{\partial F} : dF \tag{5}$$

where $d(\cdot)$ denotes the variation of the corresponding variable. Symbol : is the operation, and $A : B = \mathrm{trace}(A^T B)$. From (4), we can obtain the following formulation:

$$dF = dU\log(\Sigma)U^T + Ud(\log(\Sigma))\mathrm{U}^\mathrm{T} + \mathrm{U}\log(\Sigma)\mathrm{dU}^\mathrm{T}. \tag{6}$$

Plugging (6) into (5), $(\partial L/\partial U)$ and $(\partial L/\partial\Sigma)$ are derived as follows:

$$\begin{cases} \dfrac{\partial L}{\partial U} = \left(\dfrac{\partial L}{\partial F} + \left(\dfrac{\partial L}{\partial F}\right)^T\right)U\log(\Sigma) \\ \dfrac{\partial L}{\partial \Sigma} = \Sigma^{-1}U^T\dfrac{\partial L}{\partial F}U. \end{cases} \tag{7}$$

Next, for the given $(\partial L/\partial U)$ and $(\partial L/\partial\Sigma)$, let us compute $(\partial L/\partial C)$ through the eigendecomposition (EIG) of C and $C = U\Sigma U^T$. The chain rule expression is detailed as follows:

$$\frac{\partial L}{\partial C} : dC = \frac{\partial L}{\partial U} : dU + \frac{\partial L}{\partial \Sigma} : d\Sigma. \tag{8}$$

Similar to (6), we can obtain the variant of matrix C

$$dC = dU\Sigma U^T + Ud\Sigma U^T + U\Sigma dU^T. \quad (9)$$

By combining (8) and (9), and using the properties of the matrix inner product, :, and the properties of the EIG, the partial derivatives of the loss function L with respect to C can be derived as follows:

$$\frac{\partial L}{\partial C} = U\left\{\left(K \circ \left(U^T\frac{\partial L}{\partial U}\right)_{\text{sym}}\right) + \left(\frac{\partial L}{\partial \Sigma}\right)_{\text{diag}}\right\}U^T \quad (10)$$

where $\circ$ denotes the Hadamard product, $(\cdot)_{\text{sym}}$ denotes a symmetric operation, $(\cdot)_{\text{diag}}$ is $(\cdot)$ with all off-diagonal elements being 0, and K is computed by manipulating the eigenvalues σ in Σ as shown in the following:

$$K(i,j) = \begin{cases} \dfrac{1}{\sigma_i - \sigma_j}, & \text{if } i \neq j \\ 0, & \text{if } i = j. \end{cases} \quad (11)$$

More details about the calculation of (7) and (10) can be found in [59]. Finally, given $(\partial L/\partial C)$, the partial derivative of the loss function L with respect to feature matrix X is computed as follows:

$$\frac{\partial L}{\partial X} = \hat{I}X^T\left(\frac{\partial L}{\partial C} + \left(\frac{\partial L}{\partial C}\right)^T\right). \quad (12)$$

IV. Experimental Results and Discussion

A. Data Sets

To assess the performance of our newly proposed method, we perform extensive experiments on three popular remote sensing scene image data sets.

1) The AID30 (AID) [4] data set comprises 10 000 images divided into 30 scene classes. Each class contains hundreds of images, ranging from 220 to 420, with a size of 600 × 600 pixels in RGB space. The spatial resolution changes from about 8 to 0.5 m. Fig. 5 shows some examples of the AID data set.
2) The UC Merced Land Use (UC) [2] data set consists of 2100 images and 21 scene categories. Each class consists of 100 images with a size of 256 × 256 pixels in RGB color space. Each image has a pixel resolution of one foot. Fig. 6 shows some examples of the UC data set.
3) The NWPU-RESISC45 (NWPU) [3] comprises 31 500 images that are divided into 45 scene classes. Each class consists of 700 images with a size of 256 × 256 pixels in RGB space. The spatial resolution changes from about 30 to 0.2 m/pixel for most of the scene classes. This is one of the largest data set available according to the number of scene classes and the total number of images. Thus, it contains large-scale image variations, within-class diversity, and inter-class similarity when compared with the other data sets.

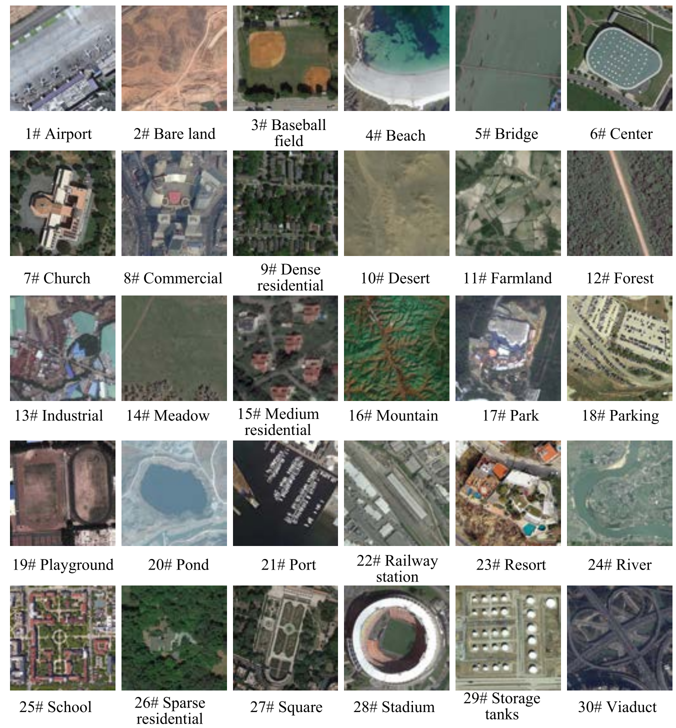


Fig. 5. Some examples of the AID data set used in experiments.

Fig. 6. Some examples of the UC data set used in experiments.

B. Implementation Details

In our experiments, two popular off-the-shelf CNN models, Alexnet [22] and VGG16 [23], are adopted as the backbone to derive the proposed SCCov network. Specifically, three convolutional layers (i.e., "conv3," "conv4," and "conv5") of Alexnet and three convolutional layers (i.e., "conv3-3," "conv4-3," and "conv5-3") of VGG16 are selected as multi-scale feature maps. The detailed architecture of the proposed SCCov network is presented in Table I. In addition, Table II shows a comparison focused on the amount of parameters needed by the original

TABLE I

ARCHITECTURE OF SCCOV NETWORK BASED ON ALEXNET AND VGG16. THE INPUT IMAGE SIZE IS 227×227 FOR ALEXNET AND 224×224 FOR VGG16. THE LAYER NAMES IN BOLD INDICATE THAT SUCH LAYERS ARE COMBINED BY SKIP CONNECTIONS. CWAVG POOLING DENOTES CHANNEL-WISE AVERAGE POOLING. FEA DENOTES FEATURE AND PROB STANDS FOR PROBABILITY

SCCovnet (Alexnet)			SCCovnet (VGG16)		
Layer name	Output size	Operation	Layer name	Output size	Operation
Conv1	$55 \times 55 \times 96$	Convolution (11×11)	Conv1-1	$224 \times 224 \times 64$	Convolution (3×3)
			Conv1-2	$224 \times 224 \times 64$	Convolution (3×3)
Conv2	$27 \times 27 \times 256$	Convolution (5×5)	Conv2-1	$112 \times 112 \times 128$	Convolution (3×3)
			Conv2-2	$112 \times 112 \times 128$	Convolution (3×3)
Conv3	$13 \times 13 \times 384$	Convolution (3×3)	Conv3-1	$56 \times 56 \times 256$	Convolution (3×3)
Fea3	$13 \times 13 \times 64$	CWAvg pooling (k=6)	Conv2-2	$56 \times 56 \times 256$	Convolution (3×3)
			Conv3-3	$56 \times 56 \times 256$	Convolution (3×3)
			Conv3-3_ds	$14 \times 14 \times 256$	Avg pooling (4×4)
			Fea3	$14 \times 14 \times 128$	CWAvg pooling (k=2)
Conv4	$13 \times 13 \times 384$	Convolution (3×3)	Conv4-1	$28 \times 28 \times 512$	Convolution (3×3)
Fea4	$13 \times 13 \times 64$	CWAvg pooling (k=6)	Conv4-2	$28 \times 28 \times 512$	Convolution (3×3)
			Conv4-3	$28 \times 28 \times 512$	Convolution (3×3)
			Conv4-3_ds	$14 \times 14 \times 512$	Avg pooling (2×2)
			Fea4	$14 \times 14 \times 128$	CWAvg pooling (k=4)
Conv5	$13 \times 13 \times 256$	Convolution (3×3)	Conv5-1	$14 \times 14 \times 512$	Convolution (3×3)
Fea5	$13 \times 13 \times 128$	CWAvg pooling (k=2)	Conv5-2	$14 \times 14 \times 512$	Convolution (3×3)
			Conv5-3	$14 \times 14 \times 512$	Convolution (3×3)
			Fea5	$14 \times 14 \times 128$	CWAvg pooling (k=4)
Concat	$13 \times 13 \times 256$	[**Fea3**;**Fea4**;**Fea5**]	Concat	$14 \times 14 \times 384$	[**Fea3**;**Fea4**;**Fea5**]
CP	32896	Covariance pooling	CP	73920	Covariance pooling
FC	Classes	Fully connection	FC	Classes	Fully connection
Prob	Classes	Softmax	Prob	Classes	Softmax

TABLE II

COMPARISON OF THE AMOUNT OF PARAMETERS NEEDED BY THE ORIGINAL CNN AND THE PROPOSED SCCOV NETWORK USING THE UCM21 DATA SET. FC DENOTES THE FULLY CONNECTED LAYER. CONV STANDS FOR THE CONVOLUTIONAL LAYER

Alexnet		SCCovnet (Alexnet)		VGG16		SCCovnet (VGG16)	
Module	Parameters	Module	Parameters	Module	Parameters	Module	Parameters
Conv1	$11 \times 11 \times 3 \times 96$	Conv1	$11 \times 11 \times 3 \times 96$	Conv1-1	$3 \times 3 \times 3 \times 64$	Conv1-1	$3 \times 3 \times 3 \times 64$
				Conv1-2	$3 \times 3 \times 64 \times 64$	Conv1-2	$3 \times 3 \times 64 \times 64$
Conv2	$5 \times 5 \times 96 \times 256$	Conv2	$5 \times 5 \times 96 \times 256$	Conv2-1	$3 \times 3 \times 64 \times 128$	Conv2-1	$3 \times 3 \times 64 \times 128$
				Conv2-2	$3 \times 3 \times 128 \times 128$	Conv2-2	$3 \times 3 \times 128 \times 128$
Conv3	$3 \times 3 \times 256 \times 384$	Conv3	$3 \times 3 \times 256 \times 384$	Conv3-1	$3 \times 3 \times 128 \times 256$	Conv3-1	$3 \times 3 \times 128 \times 256$
				Conv3-2	$3 \times 3 \times 256 \times 256$	Conv3-2	$3 \times 3 \times 256 \times 256$
				Conv3-3	$3 \times 3 \times 256 \times 256$	Conv3-3	$3 \times 3 \times 256 \times 256$
Conv4	$3 \times 3 \times 384 \times 384$	Conv4	$3 \times 3 \times 384 \times 384$	Conv4-1	$3 \times 3 \times 256 \times 512$	Conv4-1	$3 \times 3 \times 256 \times 512$
				Conv4-2	$3 \times 3 \times 512 \times 512$	Conv4-2	$3 \times 3 \times 512 \times 512$
				Conv4-3	$3 \times 3 \times 512 \times 512$	Conv4-3	$3 \times 3 \times 512 \times 512$
Conv5	$3 \times 3 \times 384 \times 256$	Conv4	$3 \times 3 \times 384 \times 256$	Conv5-1	$3 \times 3 \times 512 \times 512$	Conv5-1	$3 \times 3 \times 512 \times 512$
				Conv5-2	$3 \times 3 \times 512 \times 512$	Conv5-2	$3 \times 3 \times 512 \times 512$
				Conv5-3	$3 \times 3 \times 512 \times 512$	Conv5-3	$3 \times 3 \times 512 \times 512$
FC1	9216×4096	FC1	32896×21	FC1	25088×4096	FC1	73920×21
FC2	4096×4096			FC2	4096×4096		
FC3	4096×21			FC3	4096×21		
Total 60M (FC:91%;Conv:9%)		Total 6M		Total 130M (FC:91%;Conv:9%)		Total 13M	

CNN and our SCCov network, using the UCM21 data set (that contains 21 categories). As it can be observed, since the SCCov network contains less FC layers, it requires much less parameters than its counterpart. To train the proposed SCCov network, a two-stage training strategy is adopted. In the first training stage, we only train the last FC layer by freezing all the previous layers. Then, we unfreeze all the previous layers and train them together using the last FC layer. The learning rate is set to 0.001, and the weight decay is set to 0.0005 for all the unfrozen layers, on the two considered training stages. The batch size is set to 64. An Adagrad optimizer [60] is used for optimization. The details of the experimental settings are shown in Table III. In the first training stage, the last FC layer is initialized by means of a Gaussian distribution with zero mean and standard deviation of 0.01. The random horizontal flipping with 50% probability method is adopted for data augmentation, and no other data augmentation approaches are used. The proposed SCCov network is implemented on the MatConvNet [61] (a MATLAB toolbox for CNN).[1] Since random sampling is utilized to generate the training and test sets [3], all experiments are carried out five times. Therefore, we report the average and standard deviation of the overall accuracy (OA) after five runs. We will make our code available online.[2]

C. Comparison With State-of-the-Art Approaches

In this section, we compare our method with other state-of-the-art techniques for RSSC. Specifically, we conduct three different experiments with each of the considered data sets.

1) Experiment 1 AID Data Set: First, we conduct experiments on the AID data set. Following the experimental setup of [4], two kinds of data splits are used here. In the first split, 20% of the available samples are randomly selected for training and the rest of them are used for testing. In the second split, 50% of the available samples are randomly selected for training and the rest of them are used for testing. The following five approaches are considered for comparison.

1) Fine-tuned Alexnet and VGG16. Here, the last FC layer of the CNN model is replaced by a randomly initialized layer with specified output dimension (i.e., the output dimension is equal to the number of categories) and then trained on the test data set.
2) The method in [43], in which the authors utilize VGG net as the feature extractor and concatenate two FC layers in order to obtain the final features. Here, the linear SVM is used for classification.
3) The multi-layer stacked covariance pooling (MSCP) method in [45], where covariance pooling is used to combine deep features extracted by a pre-trained CNN model.
4) A multi-scale CNN [49], where the authors established two categories of CNNs, i.e., a fixed-size CNN and a variable-size CNN, in an attempt to address the problem of LSV in RSSC.
5) The discriminative CNN (DCNN) in [48], in which metric learning is combined with a CNN model to enlarge the distance between different classes and reduce the distance within the same class.

Table IV shows the classification results obtained in this experiment. As can be seen, the proposed SCCov network with VGG16 as the backbone outperforms the rest of the methods in almost all the cases. For example, when training rate (Tr) = 20%, the proposed SCCov network can achieve 93.12% OA, which surpasses the classification accuracy obtained by the baseline method (i.e., the fine-tuned VGG16, with OA = 90.53%) and the MSCP (with OA = 91.52%) by 2.59% and 1.6%, respectively. A similar situation can also be observed when using the Alexnet as the backbone. In addition, Fig. 7 shows the confusion matrices obtained by the baseline method (i.e., the fine-tuned VGG16) and the proposed framework on the same single experiment with Tr = 50%. From Fig. 7,

[1]http://www.vlfeat.org/matconvnet/

[2]https://github.com/henanjun

TABLE III

SETTING OF HYPERPARAMETERS FOR THE OPTIMIZATION OF THE PROPOSED SCCOV NETWORK. LR DENOTES THE LEARNING RATE, CONV DENOTES THE CONVOLUTIONAL LAYER, AND FC STANDS FOR THE FC LAYER

Models	Stage	Iterations	Batch sizes	Lr (Conv)	Lr (FC)	Weight decay	Optimizer
Alexnet	I	100 (epochs)	64	0	0.001	0.0005	Adagrad
	II	80 (epochs)	64	0.001	0.001	0.0005	Adagrad
VGG16	I	100 (epochs)	64	0	0.001	0.0005	Adagrad
	II	80 (epochs)	64	0.001	0.001	0.0005	Adagrad

TABLE IV

COMPARISON OF OAS (%) OBTAINED ON THE AID DATA SET. THE BEST VALUE IS HIGHLIGHTED IN BOLD. *PARAMETERS DENOTE THE WEIGHTS THAT NEED TO BE LEARNED IN A NEURAL NETWORK

Backbone	Method	OA (Tr=20%)	OA (Tr =50%)	*Parameters
VGG-M	[43]	-	91.86±0.28	-
Alexnet	MSCP [45]	88.99±0.38	92.36±0.21	-
VGG16	MSCP [45]	91.52±0.21	94.42±0.17	-
Alexnet	Fine-tuning	85.56±0.32	92.02±0.22	60M
VGG16	Fine-tuning	90.53±0.16	95.03±0.26	130M
Alexnet	Multiscale CNN [49]	-	91.80±0.22	60M
Alexnet	DCNN [48]	85.62±0.10	94.47±0.12	60M
VGG16	DCNN [48]	90.82±0.16	**96.89±0.10**	130M
Alexnet	SCCov (our method)	91.10±0.15	93.30±0.13	**6M**
VGG16	SCCov (our method)	**93.12±0.25**	96.10±0.16	13M

the following two observations can be concluded. First, the following categories, 7# (church), 23# (resort), 25# (school), and 27# (square), are the ones that are more difficult to recognize for both the fine-tuned VGG16 and the proposed SCCov network. The classification accuracy obtained for those classes by both the fine-tuned VGG16 and our SCCov network did not surpass 85%, which is lower than the classification accuracy obtained on the remaining categories. This is mainly due to the fact that these categories usually contain the same objects, such as buildings and trees, which makes these categories be difficult to discriminate. Second, we can also observe that the proposed SCCov network improves the classification accuracy of most categories when compared to the fine-tuned VGG16. For example, the classification accuracy of category 1# (airplane) is improved from 94.4% to 96.1%; the classification accuracy of category 6# (center) is improved from 87.7% to 90.0%, and the classification accuracy of category 25# (school) is improved from 82.0% to 88.7%. These results demonstrate the effectiveness of the proposed framework.

2) Experiment 2 UC Data Set: In this experiment, we use the UC data set for evaluating the performance of the proposed network. In this experiment, 80% of the samples are randomly selected for training and the rest are used for testing, which corresponds to the standard spilt for the UC data set [2]. The following approaches are used for comparison in this experiment:

1) the fine-tuned Alexnet and VGG16;
2) the methods in [43], [45], [48], and [49];
3) two state-of-the-art hand-crafted feature-based methods, i.e., SIFT+BoVW [2] and a feature ensemble method [34];
4) the method in [44], where an IFK and multi-scale resolution analysis are adopted to fuse the convolutional features from different layers.

The classification results are reported in Table V. From Table V, we can observe that the proposed method outperforms

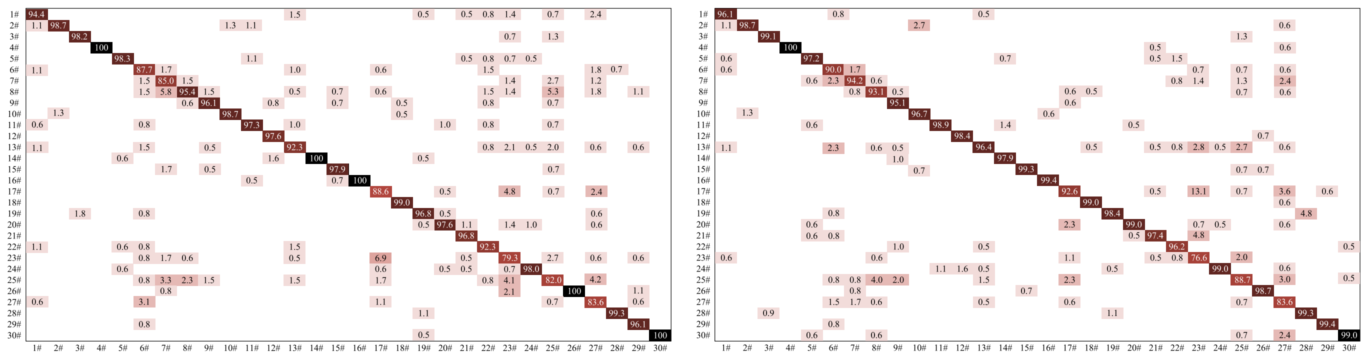


Fig. 7. Confusion matrices obtained by different methods on the AID data set with Tr = 50%. The leftmost matrix is obtained by a fine-tuned VGG16, and the rightmost one is obtained by the proposed SCCovnet with a pre-trained VGG16 as the backbone.

TABLE V

COMPARISON OF OAs (%) OBTAINED ON THE UC DATA SET. THE BEST VALUE IS HIGHLIGHTED IN BOLD. *PARAMETERS DENOTE THE WEIGHTS THAT NEED TO BE LEARNED IN A NEURAL NETWORK

Backbone or feature	Method	OA(Tr=80%)	*Parameters
SIFT	BoVW [2]	76.81	-
SIFT&Wavelet	Topic model [34]	98.33±0.98	-
VGG-M	[43]	97.42±1.79	-
VGG16	IFK [44]	98.57±0.34	-
Alexnet	MSCP [45]	97.29±0.63	-
VGG16	MSCP [45]	98.36±0.58	-
Alexnet	Fine-tuning	96.67±0.26	60M
VGG16	Fine-tuning	98.03±0.26	130M
Alexnet	Multiscale CNN [49]	96.66±0.90	60M
Alexnet	DCNN [48]	96.67±0.10	60M
VGG16	DCNN [48]	98.93±0.10	130M
Alexnet	SCCov (our method)	98.04±0.23	**6M**
VGG16	SCCov (our method)	**99.05±0.25**	13M

TABLE VI

COMPARISON OF OAs (%) OBTAINED ON THE NPWU DATA SET. THE BEST VALUE IS HIGHLIGHTED IN BOLD. *PARAMETERS DENOTE THE WEIGHTS THAT NEED TO BE LEARNED IN A NEURAL NETWORK

Backbone	Method	OA(Tr=10%)	OA(Tr =20%)	*Parameters
Alexnet	BoVW [42]	55.22±0.39	59.22±0.18	-
VGG16	BoVW [42]	82.65±0.31	84.32±0.17	-
Alexnet	MSCP [45]	81.70±0.23	85.58±0.16	-
VGG16	MSCP [45]	85.33±0.21	88.93±0.14	-
Alexnet	Fine-tuning	80.66±0.29	84.74±0.31	60M
VGG16	Fine-tuning	87.76±0.10	91.67±0.12	130M
Alexnet	DCNN [48]	85.56±0.20	87.24±0.12	60M
VGG16	DCNN [48]	89.22±0.50	91.89±0.22	130M
Alexnet	SCCov (our method)	84.33±0.26	87.30±0.23	**6M**
VGG16	SCCov (our method)	**89.30±0.35**	**92.10±0.25**	13M

TABLE VII

COMPARISON OF OAs (%) AND KAPPA COEFFICIENTS OBTAINED ON THE NWPU DATA SET IN OUR ABLATION EXPERIMENTS. THE BEST VALUES ARE HIGHLIGHTED IN BOLD

Backbone	Method	OA	Kappa
Alexnet	Fine-tuning	80.66±0.29	80.36±0.25
	SCCov without skip	83.79±0.39	83.26±0.27
	SCCov with GAP	75.65±0.31	75.21±0.23
	SCCov	**84.33±0.26**	**84.22±0.19**
VGG16	Fine-tuning	87.76±0.10	86.87±0.15
	SCCov without skip	87.33±0.21	87.02±0.18
	SCCov with GAP	80.30±0.29	80.10±0.25
	SCCov	**89.30±0.35**	**89.17±0.30**

all the counterparts with OA of 99.05%. Moreover, Fig. 8 shows the confusion matrices obtained by the baseline method (i.e., the fine-tuned VGG16) and the proposed framework in one single experiment. From the confusion matrices, we can also observe that the following categories are relatively difficult to recognize: 1) 7# (dense residential); 2) 13# (medium residential); and 3) 20# (storage tank). This is because the first two categories have very similar semantic information—they all describe residential areas and the only difference among them is the density of the buildings. Thus, it could easily lead to misclassifications. In this regard, the proposed SCCov network can also improve the classification performance achieved by the fine-tuned VGG16 in most categories. For instance, the OA of category 1# (agriculture) is improved from 95% to 100% and the OA of category 5# (building) is improved from 85% to 100%.

3) Experiment 3 *NWPU Data Set:* Finally, the proposed SCCov network is also compared to several RSSC methods using the NWPU data set. These methods include the following:

1) the fine-tuned Alexnet and VGG16;
2) the methods in [45] and [48];
3) the method in [42], where the BoVW method is used to encode the convolutional features for RSSC.

Two kinds of data splits are used in this experiment for comprehensive comparison. The first one randomly selects 10% of the available samples for training and uses the rest of them for testing. The second one randomly selects 20% of the available samples for training and uses the rest of them for testing. Both data splits are derived following the experimental setup in [3]. The classification results obtained by these methods are shown in Table VI. As can be seen in Table VI, the proposed SCCov network exhibits a better classification performance than the rest compared methods.

D. Ablation Experiments

In this section, we conduct two ablation experiments to, respectively, demonstrate the effectiveness of our two new modules, i.e., skip connections and covariance pooling. The NWPU data set has been selected for illustrative purposes, with 10% of the available samples randomly selected for training and the rest of the samples used for testing. The experimental settings, including training strategy, batch size,

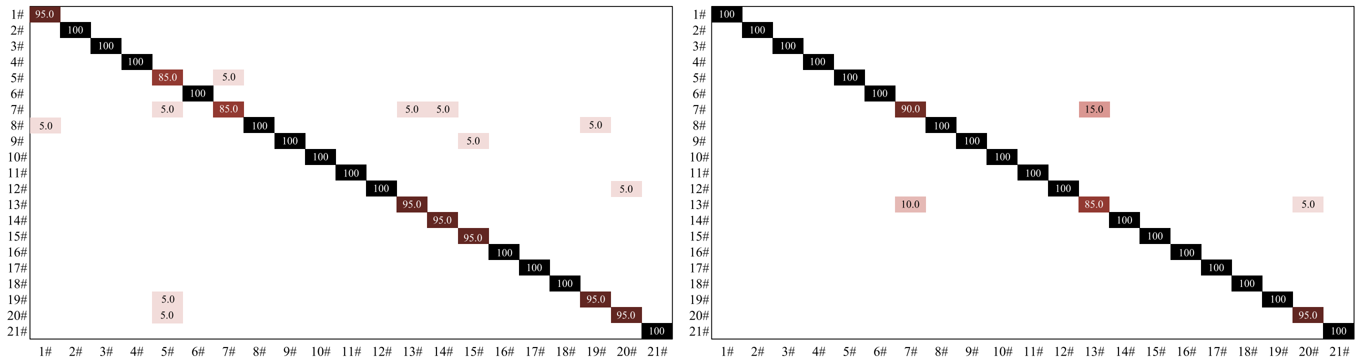


Fig. 8. Confusion matrices obtained by different methods on the UC data set with Tr = 80%. The leftmost matrix is obtained by a fine-tuned VGG16, and the rightmost one is obtained by the proposed SCCovnet with a pre-trained VGG16 as the backbone.

Fig. 9. Visualization results obtained by the fine-tuned Alexnet and SCCov network on the AID data set. The first line contains the original image, the second line contains the saliency image obtained by the fine-tuned Alexnet, and the last line contains the saliency image obtained by the SCCov network. Both the fine-tuned Alexnet and the proposed SCCov network are trained on the same data set split, and the images from the test set are used for visualization purposes. The description under the original image is the true label, while the description under the saliency image is the prediction label. (a)–(c) Classified correctly by both fine-tuned Alexnet and SCCov net. (d)–(f) Classified correctly only by SCCov net.

and learning rate, are kept the same for our SCCov network before and after ablation for a fair comparison.

1) Experiment 1: In the first ablation experiment, we remove the skip connection module of the SCCov network and append the covariance pooling behind the last convolutional layer (SCCov network without skip). Specifically, for the Alexnet, the covariance pooling is appended after the "conv5" convolutional layer. For the VGG16, the last convolution layer "conv5-3" is first transformed by a 1×1 convolutional kernel and then followed by the covariance pooling. The 1×1 convolutional kernel is adopted here to make sure that the SCCov network has the same amount of training parameters before and after the ablation. Note that, the other components along with the training strategy and the hyperparameters of the SCCov network without skip are kept the same as in the original SCCov network. The experimental results based on the OA and Kappa coefficient are shown in Table VII. For more comprehensive comparison, the classification results of the baseline methods (i.e., the fine-tuned Alexnet and fine-tuned VGG16) are also shown. As it can be observed, after removing the skip connections module from the SCCov network, the classification performance of the proposed SCCov network drops significantly. For example, with the VGG16 as the backbone, the classification accuracy achieved by the proposed SCCov is 89.30%, whereas the classification accuracy achieved by SCCov without skip connections is 87.33%. Additionally, we can also observe that the Kappa coefficient for the proposed method is 89.17%, while the Kappa coefficient for the SCCov without skip connections [62], [63] is 87.02%. Both the classification accuracy and the Kappa coefficient drop

Fig. 10. Visualization results obtained by the fine-tuned Alexnet and SCCov network on the UC data set. The first line contains the original image, the second line contains the saliency image obtained by the fine-tuned Alexnet, and the last line contains the saliency image obtained by the SCCov network. Both the fine-tuned Alexnet and the proposed SCCov network are trained on the same data set split, and the images from the test set are used for visualization purposes. The description under the original image is the true label, while the description under the saliency image is the prediction label. (a)–(c) Classified correctly by both fine-tuned Alexnet and SCCov net. (d)–(f) Classified correctly only by SCCov net.

over 1.9%. The main reason is that with the skip connections' module, the multi-resolution feature maps with pyramidal shape can be integrated together, which is helpful to address the problem of LSV in RSSC.

2) Experiment 2: In the second ablation experiment, the covariance pooling in the proposed SCCov network is replaced by a classical pooling strategy, i.e., global average pooling (GAP) [64]. Specifically, the GAP is appended behind the concatenated multi-resolution feature map, which is followed by FC and a softmax layer. The method after ablation is denoted by the SCCov network with GAP. The corresponding classification results based on the OA and the Kappa coefficient are reported in Table VII. For more comprehensive comparison, the classification results of the baseline method (i.e., fine-tuned Alexnet and fine-tuned VGG16) are also shown. From Table VII, we can observe that the proposed SCCov network can outperform the SCCov network with GAP by a considerable margin. For instance, with Alexnet as the backbone, the classification performance obtained by the SCCov network with GAP is less than 76%, while the classification performance obtained by the SCCov is 84.33%. In addition, the Kappa coefficient for the SCCov with GAP is 75.21%, whereas the Kappa coefficient of the proposed method is 84.22%. The improvement in terms of both OA and Kappa coefficient obtained by the covariance pooling method is over 8%. The main reason is that the covariance pooling can fully exploit the high-order information among the multi-resolution feature maps, which is beneficial to learn more representative features.

E. Statistical Test

In order to better assess the statistical significance of the difference between the proposed method and the baseline methods (i.e., the fine-tuned Alexnet and the fine-tuned VGG16), we conduct McNemar's test [65], which is based upon a standardized normal test as described in the following:

$$Z = \frac{l_{12} - l_{21}}{\sqrt{|l_{12} - l_{21}|}} \tag{13}$$

where l_{12} indicates the number of samples classified correctly by method 1 and incorrectly by method 2. If $|Z| > 1.96$, we can conclude that the difference in accuracy between methods 1 and 2 is statistically significant. The sign of Z indicates whether method 1 is more accurate than method 2 ($Z > 0$) or vice versa ($Z < 0$). McNemar's test results corresponding to our study are reported in Table VIII. As can be seen, all the values of Z are much greater than 1.96, and thus, we can conclude that the improvements of our proposed method over the baseline methods are statistically significant.

F. Visualization Experiment

In this experiment, we attempt to figure out which parts of the considered images make more significant contributions to the final scene recognition to further investigate the working mechanism of SCCov network. To achieve this goal, we visualize the saliency of the test image obtained by the SCCov network with Alexnet as the backbone. Specifically, we first find the weight in the last FC with respect to the max score

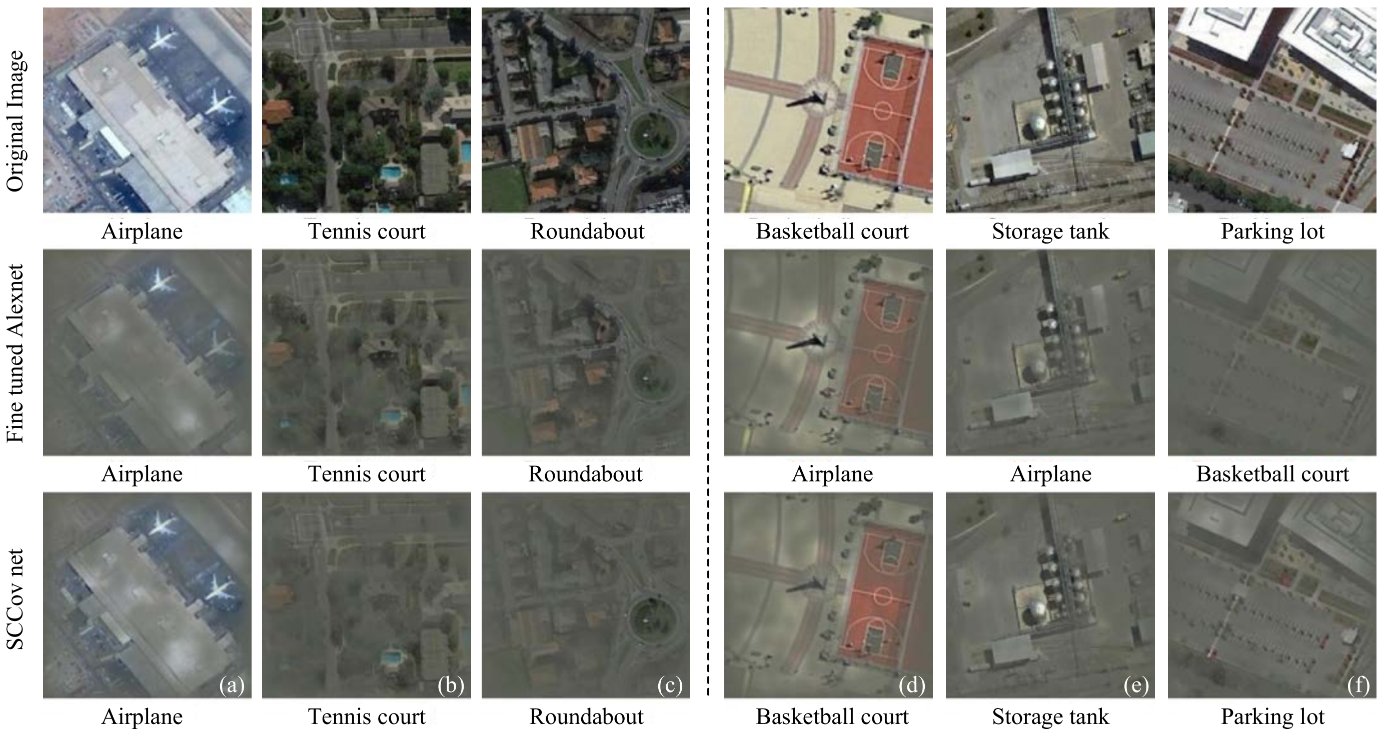


Fig. 11. Visualization results obtained by the fine-tuned Alexnet and SCCov network on the NWPU data set. The first line contains the original image, the second line contains the saliency image obtained by the fine-tuned Alexnet, and the last line contains the saliency image obtained by the SCCov network. Both the fine-tuned Alexnet and the proposed SCCov network are trained on the same data set split, and the images from the test set are used for visualization purposes. The description under the original image is the true label, while the description under the saliency image is the prediction label. (a)–(c) Classified correctly by both fine-tuned Alexnet and SCCov net. (d)–(f) Classified correctly only by SCCov net.

TABLE VIII

STATISTICAL SIGNIFICANCE, MEASURED BY MCNEMAR'S TEST, FOR THE PROPOSED SCCOV NET AND THE BASELINE METHODS

NWPU Data Set (Tr=10%)	NWPU Data Set (Tr=20%)
Z/significant?/better?	Z/significant?/better
SCCov with Alexnet *vs* Fine tuned Alexnet	
36.26/yes/yes	25.80/yes/yes
SCCov with VGG16 *vs* Fine tuned VGG16	
25.24/yes/yes	17.75/yes/yes

in the softmax layer, and then, the obtained weight is used as the derivative of the last FC for backward propagation to get the derivative of the input image with respect to the weight. Finally, the derivative of the input image is regarded as the saliency image, and the dot product of the saliency image and the original image is used for visualization purposes. In addition, to make a comprehensive comparison, the saliency images obtained by the baseline fine-tuned Alexnet are also visualized. The visualization results of some samples from the three test data sets are shown in Figs. 9–11. From the visualization results, the following three observations can be made. First, the working mechanism of the CNN model is similar to a human vision recognition system when recognizing a remote sensing scene image, that is, to recognize a scene, the human vision system pays more attention to the representative objects in the scene. For example, in order to recognize an *airport* scene, both the fine-tuned VGG16 net and the proposed method pay more attention to the *airplane* in the scene, which corresponds to the highlighted zoomed-in view of the saliency images (see the first column in Fig. 9). To recognize a *parking* scene, both two networks focus on the *cars* (see the third column in Fig. 9). Second, we can see that the fine-tuned Alexnet pays less attention to the representative objects in a scene, despite making the correct prediction [see Figs. 9(a)–(c), 10(a)–(c), and 11(a)–(c)]. Last but not least, in Figs. 9(d)–(f), 10(d)–(f), and 11(d)–(f), we can observe that the baseline method latches on the incorrect objects in the remote sensing scene categories, which are corrected by the proposed SCCov network. We emphasize that the aforementioned explanation of the visualization results is intuitive and qualitative. A more quantitative and precise interpretation needs to be developed in the future developments.

V. CONCLUSION

In this paper, we presented a new end-to-end learning model called SCCov network for remote sensing scene classification. By introducing two new components, i.e., skip connections and covariance pooling in the associated CNN, our SCCov network can not only combine the multi-resolution feature maps from different layers in the CNN model together but also exploit the high-order information for achieving a more representative feature learning. Comprehensive experiments on three publicly available remote sensing image scene classification data sets, as well as a detailed comparison with state-of-the-art methods, verify the effectiveness of our newly developed approach.

In the future, we will explore the development of quantitative experiments to analyze the visualization and interpretation of results performed by our SCCov compared to other approaches. Moreover, we will also consider combining the first-order information and the second-order information of feature maps in the CNN model to further improve the classification performance.

Acknowledgment

The authors would like to thank the editors and the anonymous reviewers for their valuable comments and suggestions, which greatly helped them to enhance the technical quality and presentation of this paper.

References

[1] 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.

[2] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. Int. Conf. Adv. Geographic Inf. Syst.*, 2010, pp. 270–279.

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

[4] G.-S. Xia *et al.*, "AID: A benchmark data set for performance evaluation of aerial scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3965–3981, Jul. 2017.

[5] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[6] D. Zhang, J. Han, J. Han, and L. Shao, "Cosaliency detection based on intrasaliency prior transfer and deep intersaliency mining," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 6, pp. 1163–1176, Jun. 2016.

[7] B. Shi, X. Bai, W. Liu, and J. Wang, "Face alignment with deep regression," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 1, pp. 183–194, Jan. 2018.

[8] Y. Yuan, L. Mou, and X. Lu, "Scene recognition by manifold regularized deep learning architecture," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 10, pp. 2222–2233, Oct. 2015.

[9] J. Liu, M. Gong, K. Qin, and P. Zhang, "A deep convolutional coupling network for change detection based on heterogeneous optical and radar images," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 3, pp. 545–559, Mar. 2018.

[10] M. Gong, J. Zhao, J. Liu, Q. Miao, and L. Jiao, "Change detection in synthetic aperture radar images based on deep neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 1, pp. 125–138, Jan. 2016.

[11] J. Liu, M. Gong, Q. Miao, X. Wang, and H. Li, "Structure learning for deep neural networks based on multiobjective optimization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 6, pp. 2450–2463, Jun. 2018.

[12] X. Niu, M. Gong, T. Zhan, and Y. Yang, "A conditional adversarial network for change detection in heterogeneous images," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 1, pp. 45–49, Jan. 2019.

[13] L. Mou, P. Ghamisi, and X. X. Zhu, "Unsupervised spectral–spatial feature learning via deep residual Conv–Deconv network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 1, pp. 391–406, Jan. 2018.

[14] R. Dian, S. Li, A. Guo, and L. Fang, "Deep hyperspectral image sharpening," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 11, pp. 5345–5355, Nov. 2018.

[15] 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.

[16] G. Cheng, Z. Li, J. Han, X. Yao, and L. Guo, "Exploring hierarchical convolutional features for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 11, pp. 6712–6722, Nov. 2018.

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

[18] Z. Gong, P. Zhong, Y. Yu, W. Hu, and S. Li, "A CNN with multiscale convolution and diversified metric for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 6, pp. 3599–3618, Jun. 2019. doi: 10.1109/TGRS.2018.2886022.

[19] N. He *et al.*, "Feature extraction with multiscale covariance maps for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 755–769, Feb. 2019.

[20] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "High-resolution aerial image labeling with convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 7092–7103, Dec. 2017.

[21] S. Yang, M. Wang, Z. Feng, Z. Liu, and R. Li, "Deep sparse tensor filtering network for synthetic aperture radar images classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 8, pp. 3919–3924, Aug. 2018. doi: 10.1109/TNNLS.2017.2688466.

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

[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–13.

[24] T. Lin, P. Doll'r, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 936–944.

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

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

[27] 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.

[28] M. L. Mekhalfi, F. Melgani, Y. Bazi, and N. Alajlan, "Land-use classification with compressive sensing multifeature fusion," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 10, pp. 2155–2159, Oct. 2015.

[29] B. Zhao, Y. Zhong, G.-S. Xia, and L. Zhang, "Dirichlet-derived multiple topic scene classification model for high spatial resolution remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 4, pp. 2108–2123, Apr. 2016.

[30] Q. Zhu, Y. Zhong, B. Zhao, G.-S. Xia, and L. Zhang, "Bag-of-visual-words scene classifier with local and global features for high spatial resolution remote sensing imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 6, pp. 747–751, Jun. 2016.

[31] K. Qi, H. Wu, C. Shen, and J. Gong, "Land-use scene classification in high-resolution remote sensing images using improved correlatons," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 12, pp. 2403–2407, Dec. 2015.

[32] B. Zhao, Y. Zhong, and L. Zhang, "A spectral-structural bag-of-features scene classifier for very high spatial resolution remote sensing imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 116, pp. 73–85, Jun. 2016.

[33] 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.

[34] Q. Zhu, Y. Zhong, S. Wu, L. Zhang, and D. Li, "Scene classification based on the sparse homogeneous–heterogeneous topic feature model," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 5, pp. 2689–2703, May 2018.

[35] F. Zhang, B. Du, and L. Zhang, "Saliency-guided unsupervised feature learning for scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 4, pp. 2175–2184, Apr. 2015.

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

[37] F. Hu, G.-S. Xia, Z. Wang, X. Huang, L. Zhang, and H. Sun, "Unsupervised feature learning via spectral clustering of multidimensional patches for remotely sensed scene classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 5, pp. 2015–2030, May 2015.

[38] 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.*, Jun. 2014, pp. 1717–1724.

[39] 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.*, Jun. 2014, pp. 580–587.

[40] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[41] F. Hu, G.-S. Xia, J. Hu, and L. Zhang, "Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery," *Remote Sens.*, vol. 7, no. 11, pp. 14680–14707, Nov. 2015.

[42] G. Cheng, Z. Li, X. Yao, L. Guo, and Z. Wei, "Remote sensing image scene classification using bag of convolutional features," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 10, pp. 1735–1739, Oct. 2017.

[43] S. Chaib, H. Liu, Y. Gu, and H. Yao, "Deep feature fusion for VHR remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 8, pp. 4775–4784, Aug. 2017.

[44] E. Li, J. Xia, P. Du, C. Lin, and A. Samat, "Integrating multilayer features of convolutional neural networks for remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 10, pp. 5653–5665, Oct. 2017.

[45] N. He, L. Fang, S. Li, A. Plaza, and J. Plaza, "Remote sensing scene classification using multilayer stacked covariance pooling," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 12, pp. 6899–6910, Dec. 2018.

[46] Q. Zhu, Y. Zhong, L. Zhang, and D. Li, "Adaptive deep sparse semantic modeling framework for high spatial resolution image scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 10, pp. 6180–6195, Oct. 2018.

[47] M. Castelluccio, G. Poggi, C. Sansone, and L. Verdoliva, "Land use classification in remote sensing images by convolutional neural networks," Aug. 2015, *arXiv:1508.00092*. [Online]. Available: http://arxiv.org/abs/1508.00092

[48] G. Cheng, C. Yang, X. Yao, L. Guo, and J. Han, "When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNs," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 5, pp. 2811–2821, May 2018.

[49] Y. Liu, Y. Zhong, and Q. Qin, "Scene classification based on multiscale convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 12, pp. 7109–7121, Dec. 2018.

[50] R. M. Anwer, F. S. Khan, J. van de Weijer, M. Molinier, and J. Laaksonen, "Binary patterns encoded convolutional neural networks for texture recognition and remote sensing scene classification," *ISPRS J. Photogramm. Remote Sens.*, vol. 138, pp. 74–85, Apr. 2018.

[51] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[52] D. Zeng, S. Chen, B. Chen, and S. Li, "Improving remote sensing scene classification by integrating global-context and local-object features," *Remote Sens.*, vol. 10, no. 5, pp. 1–19, May 2018.

[53] Z. Gong, P. Zhong, Y. Yu, and W. Hu, "Diversity-promoting deep structural metric learning for remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 1, pp. 371–390, Jan. 2018.

[54] T.-Y. Lin, A. RoyChowdhury, and S. Maji, "Bilinear CNN models for fine-grained visual recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1449–1457.

[55] P. Li, J. Xie, Q. Wang, and W. Zuo, "Is second-order information helpful for large-scale visual recognition?" in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2089–2097.

[56] X. Dai, J. Y.-H. Ng, and L. S. Davis, "FASON: First and second order information fusion network for texture recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 6100–6108.

[57] Q. Wang, P. Li, and L. Zhang, "G2DeNet: Global Gaussian distribution embedding network and its application to visual recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 6507–6516.

[58] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, "Geometric means in a novel vector space structure on symmetric positive-definite matrices," *SIAM J. Matrix Anal. Appl.*, vol. 29, no. 1, pp. 328–347, 2006.

[59] C. Ionescu, O. Vantzos, and C. Sminchisescu, "Matrix backpropagation for deep networks with structured layers," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2965–2973.

[60] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, pp. 2121–2159, Feb. 2011.

[61] A. Vedaldi and K. Lenc, "Matconvnet-convolutional neural networks for MATLAB," in *Proc. ACM Int. Conf. Multimedia*, 2015, pp. 689–692.

[62] X. Kang, P. Duan, S. Li, and J. A. Benediktsson, "Decolorization-based hyperspectral image visualization," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4346–4360, Aug. 2018.

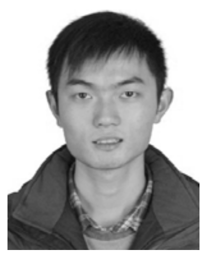
[63] X. Kang, P. Duan, X. Xiang, S. Li, and J. A. Benediktsson, "Detection and correction of mislabeled training samples for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 10, pp. 5673–5686, Oct. 2018.

[64] M. Lin, Q. Chen, and S. Yan, "Network in network," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 1–10.

[65] G. M. Foody, "Thematic map comparison: Evaluating the statistical significance of differences in classification accuracy," *Photogramm. Eng. Remote Sens.*, vol. 70, no. 5, pp. 627–633, May 2004.

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, under the support of the China Scholarship Council. His current research interests include remote sensing image classification and remote sensing object detection.



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

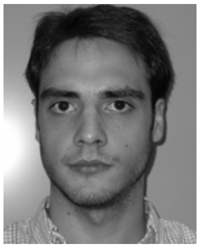
From 2011 to 2012, he was a Visiting Ph.D. Student with the Department of Ophthalmology, Duke University, Durham, NC, USA, under the support of the China Scholarship Council. From 2016 to 2017, he was a Post-Doctoral Research Fellow with the Department of Biomedical Engineering, Duke University. Since 2017, he has been an Associate Professor with the College of Electrical and Information Engineering, Hunan University. His current research interests include sparse representation and multi-resolution analysis in remote sensing and medical image processing.

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

Shutao Li (M'07–SM'15–F'19) received the B.S., M.S., and Ph.D. degrees from Hunan University, Changsha, China, in 1995, 1997, and 2001, respectively.

In 2001, he joined the College of Electrical and Information Engineering, Hunan University. From 2001 to 2001, he was a Research Associate with the Department of Computer Science, The Hong Kong University of Science and Technology, Hong Kong. From 2002 to 2003, he was a Postdoctoral Fellow with the Royal Holloway College, University of London, Egham, U.K. In 2005, he was a Visiting Professor with the Department of Computer Science, The Hong Kong University of Science and Technology. He is currently a Full Professor with the College of Electrical and Information Engineering, Hunan University. He has authored or coauthored over 200 refereed papers. His current research interests include image processing, pattern recognition, and artificial intelligence.

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



Javier Plaza (M'09–SM'15) received the M.Sc. and Ph.D. degrees in computer engineering from the Hyperspectral Computing Laboratory, Department of Technology of Computers and Communications, University of Extremadura, Cáceres, Spain, in 2004 and 2008, respectively.

He is currently a member of the Hyperspectral Computing Laboratory, Department of Technology of Computers and Communications, University of Extremadura. He has authored more than 150 publications, including over 50 JCR journal papers, ten book chapters, and 90 peer-reviewed conference proceeding papers. His current research interests comprise hyperspectral data processing and parallel computing of remote sensing data.

Dr. Plaza received the Best Paper Awards at the IEEE International Conference on Space Technology and the IEEE Symposium on Signal Processing and Information Technology. He was a recipient of the Outstanding Ph.D. Dissertation Award at the University of Extremadura in 2008. He was a recipient of the Best Column Award of the *IEEE Signal Processing Magazine* in 2015 and the Most Highly Cited Paper (2005–2010) of the *Journal of Parallel and Distributed Computing*. He has guest edited three special issues on hyperspectral remote sensing for different journals. He is an Associate Editor of the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS and the IEEE Remote Sensing Code Library.

Antonio Plaza (M'05–SM'07–F'15) received the M.Sc. and Ph.D. degrees in Computer Engineering from the Hyperspectral Computing Laboratory, Department of Technology of Computers and Communications, University of Extremadura Cáceres, Spain, in 1999 and 2002, respectively.

He is currently the Head of the Hyperspectral Computing Laboratory, Department of Technology of Computers and Communications, University of Extremadura. He has authored more than 600 publications, including more than 200 JCR journal papers (more than 150 in IEEE journals), 24 book chapters, and over 300 peer-reviewed conference proceeding papers. He has guest edited ten special issues on hyperspectral remote sensing for different journals. His current research interests include hyperspectral data processing and parallel computing of remote sensing data.

Dr. Plaza is a fellow of the IEEE for his contributions to hyperspectral data processing and parallel computing of earth observation data. He was a member of the Editorial Board of the IEEE Geoscience and Remote Sensing Newsletter from 2011 to 2012 and the IEEE Geoscience and Remote Sensing Magazine in 2013. He was also a member of the Steering Committee of the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING (JSTARS). He received the Best Paper Awards at the IEEE International Conference on Space Technology and the IEEE Symposium on Signal Processing and Information Technology. He was a recipient of the recognition of Best Reviewers of the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS in 2009 and the recognition of Best Reviewers of the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING in 2010. He was a recipient of the Best Column Award of the *IEEE Signal Processing Magazine* in 2015, the 2013 Best Paper Award of the JSTARS journal, and the Most Highly Cited Paper (2005–2010) of the *Journal of Parallel and Distributed Computing*. He served as an Associate Editor for the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING from 2007 to 2012. He is also an Associate Editor of the IEEE ACCESS. He served as the Director of Education Activities for the IEEE Geoscience and Remote Sensing Society (GRSS) from 2011 to 2012 and the President of the Spanish Chapter of the IEEE GRSS from 2012 to 2016. He has reviewed more than 500 manuscripts for over 50 different journals. He served as the Editor-in-Chief of the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING journal from 2013 to 2017.