Branch Feature Fusion Convolution Network for Remote Sensing Scene Classification

Cuiping Shi, *Member, IEEE*, Tao Wang, and Liguo Wang, *Member, IEEE*

***Abstract*—Convolutional neural networks (CNNs) have outstanding advantages in the classification of remote sensing scenes. Deep CNN models with better classification performance typically have high complexity, whereas shallow CNN models with low complexity rarely achieve good classification performance for remote sensing images with complex spatial structures. In this article, we proposed a new lightweight CNN classification method based on branch feature fusion (LCNN-BFF) for remote sensing scene classification. In contrast to a conventional single linear convolution structure, the proposed model had a bilinear feature extraction structure. The BFF method was utilized to fuse the feature information extracted from the two branches, which improved the classification accuracy. In addition, combining depthwise separable convolution and conventional convolution to extract image features greatly reduced the complexity of the model on the premise of ensuring the accuracy of classification. We tested the method on four standard datasets. The experimental results showed that, compared with recent classification methods, the number of weight parameters of the proposed method only accounted for less than 5% of the other methods; however, the classification accuracy was equivalent to or even superior to certain high-performance classification methods.**

***Index Terms*—Combined convolution (CConv) structures, convolutional neural network (CNN), depthwise separable convolution (DSC), feature extraction, feature fusion, remote sensing scene classification.**

I. Introduction

MORE and more researchers have performed research work in the field of remote sensing. Remote sensing scene classification assigns a specific label based on the image content of a random scene [1]–[3]. The relevant research results are widely used in many important fields, such as national defense security, climate change monitoring, environmental monitoring and management, land classification for different purposes, ground target identification and detection, loss assessment in natural disasters, and other important fields [4]–[8].

Remote sensing scene classification is very challenging research work. Due to the difference of distance and the diversity of landforms, scene images typically have a complex spatial structure, which easily produces high within-class differences and between-class similarities. This makes it difficult to accurately classify remote sensing scenes. Deep learning technology provides a new way to solve these problems. This technology can learn representative and differentiated abstract features from data and was considered one of the top ten breakthrough technologies in 2013. In recent years, deep learning has become a research hotspot in the field of computer vision, and has been gradually introduced to the field of geosciences and remote sensing for intelligent algorithm research and big data analysis [9], [10].

Convolutional neural networks (CNNs) are one of the most representative neural networks in the field of deep learning technology, and they show excellent performance in the field of computer vision [11]–[13]. In recent years, CNNs have been widely used in remote sensing image classification. As deep CNN models can extract more representative image features, the classification of remote sensing image scenes with complex spatial structures typically demonstrates better performance. At present, in order to improve the performance of remote sensing image classification, deep CNN models are used in many studies on remote sensing scene classification [46], [47].

Although the deeper networks can effectively extract the representative features of images, there are still two difficulties in remote sensing scene image classification. On the one hand, remote sensing scene images typically have high within-class differences and between-class similarities. Fig. 1 shows some image examples, which are from the NWPU-RESISC45 (NWPU) dataset [14]. We can see that, in this case, it is very difficult to classify remote sensing scene images accurately. On the other hand, deeper networks often have higher computational complexity. With the development of imaging technology, the resolution of remote sensing images is increasing, and the amount of data is large, which makes the data processing more time consuming for the deep CNN models with high complexity. In practical applications, not only the accuracy of scene classification but also the time-consumption of the model should also

Manuscript received May 17, 2020; revised July 20, 2020; accepted August 16, 2020. Date of publication August 20, 2020; date of current version September 11, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 41701479 and Grant 61675051, in part by China Postdoctoral Science Foundation under Grant 2017M621246, in part by the Postdoctoral Science Foundation of Heilongjiang Province of China under Grant LBH-Z17052, in part by the Project plan of Science Foundation of Heilongjiang Province of China under Grant QC2018045, in part by the Fundamental Research Funds in Heilongjiang Provincial Universities of China under Grant 135309342 and Grant 135309456, in part by the Key project of Education Science of the 13th Five-year Plan in Heilongjiang Province in 2020 under Grant GJB1320385, in part by the Research Project of Higher Education Teaching Reform in Heilongjiang Province under Grant SJGY20190718, and in part by the Innovation and Entrepreneurship Training Program for College Students under Grant 201910232059. *(Corresponding author: Liguo Wang.)*

Cuiping Shi is with the Department of Communication Engineering, Qiqihar University, Qiqihar 161000, China, and also with the College of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China (e-mail: scp1980@126.com).

Tao Wang is with the Department of Communication Engineering, Qiqihar University, Qiqihar 161000, China (e-mail: 2681704096@qq.com).

Liguo Wang is with the College of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China (e-mail: wangliguo@hrbeu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2020.3018307

Fig. 1. Examples of remote sensing scene images. (a) Scenes with large within-class differences. (b) Scenes with high between-class similarities.

be considered. Aiming at these two problems, in this article, we proposed a new lightweight CNN classification method based on branch feature fusion (LCNN-BFF). The network structure consists of nine parts (Groups 1–9). Groups 1–8 are the feature extraction structure and Group 9 is the classification layer. Groups 4–7 adopt the structure of bilinear convolution feature extraction, which utilizes two branches to extract image features.

According to the structure of bilinear feature extraction, we proposed a BFF method to fuse the image feature information extracted from the two branches. To solve the problem of model complexity caused by a bilinear convolution structure, the depthwise separable convolution (DSC) method was adopted in this article. In addition, two convolution methods, i.e., DSC and CConv, were combined to extract scene image features, which can effectively avoid insufficient feature extraction and reduce the number of weight parameters. The experimental results showed that the proposed method outperformed not only the classical CNN, but also, in terms of certain evaluation indexes, some of the state-of-the-art classification methods presented recently.

Our main contributions in this article are listed as follows.

1) Aiming at the problems of high within-class differences and between-class similarities in remote sensing scene images, we proposed a bilinear convolution feature extraction structure and BFF method. Through BFF, the different features extracted from two branches were fused and complemented, which greatly improved the classification accuracy.
2) In view of the high complexity of the model caused by the structure of bilinear convolution feature extraction, we proposed a new convolution strategy, which combined the two convolution methods of light convolution DSC and CConv. This method not only reduced the complexity of the model, but also improved the classification accuracy further.

The remainder of this article is organized as follows. In Section II, we discuss the related works of remote sensing scene classification. In Section III, the proposed LCNN-BFF method is described in detail. In Section IV, in order to prove the high effectiveness of the proposed method, we performed some numerical experiments, and compared the results with certain state-of-the-art classification methods. Finally, our conclusions and discussions are provided in Section V.

II. Related Work

A CNN is one of the most representative neural networks in deep learning technology. In the field of remote sensing, a CNN is applied in many applications, especially in target detection [15], [16], semantic annotation [65], high-resolution image classification [17], [18], hyperspectral image classification [19], [20], [66], and remote sensing scene classification [21]–[24]. Various methods based on a CNN have shown excellent performance in the field of remote sensing, mainly as a deep neural network can extract better image representation features, for example, VGG16 [25], AlexNet [26], MobileNet [27], and other network models [41], [42].

In particular, the MobileNet network was proposed as a lightweight network that requires less parameters and computation, but with better performance than certain deep networks with high complexity. In the MobileNet network, a new convolution method, called DSC, was proposed, which is widely used in image classification [67], [68]. In recent years, researchers have made great efforts in remote sensing scene classification. In addition to the above-mentioned deep CNN feature learning-based methods, there are also the handcrafted feature-based methods and the unsupervised feature learning-based methods [14]. The following is a brief introduction to these works.

A. Handcrafted Feature-Based Methods

The early works of scene classification are mainly handcrafted feature-based methods, which often use handcrafted feature descriptors to extract the features of remote sensing scenes. Different feature descriptors are chosen, among which the most widely used ones include, but are not limited to, color histograms [28], texture descriptors [29], GIST [30], scale-invariant feature transform (SIFT) [31], and histogram of oriented gradients (HOG) [32]. However, the feature extraction ability of the handcrafted methods is poor. In using this method to classify remote sensing scene images with rich details, not only is the work heavy but the classification performance is also insufficient, which makes it difficult to meet the practical applications of remote sensing scene classification.

B. Unsupervised Feature Learning Based Methods

To overcome the limitations of manual operation, many researchers proposed automatic feature extraction methods, i.e.,

unsupervised feature learning. Unsupervised feature learning can extract features from images automatically. In this way, more features of images can be obtained. Common unsupervised feature learning methods include K-means clustering, sparse coding [33], and autoencoders [34]. In [35], Zhu *et al.* proposed a local–global feature bag-of-visual-words scene classifier for high spatial resolution remote sensing imagery. In this method, the shape-based invariant texture index was designed as the global texture feature, the mean and standard deviation values were employed as the local spectral feature, and the dense SIFT feature was utilized as the structural feature.

In [36], Cheriyadat adopted dense low-level feature descriptors to characterize the local spatial patterns. These unlabeled feature measurements were exploited in a novel way to learn a set of base functions. The low-level feature descriptors were encoded in terms of the base functions to generate new sparse representation for the feature descriptors. In [37], Cheng *et al.* proposed a new and effective autoencoder-based method to learn a shared midlevel visual dictionary. The discriminative midlevel visual elements, rather than individual pixels or low-level image features, were used to represent images. Overall, the classification performance of the unsupervised feature learning-based method was typically better than that of the handcrafted-based method. However, it is still difficult to further improve the performance of remote sensing scene classification as the learned features of this method are often low-level image features, and thus the description ability of the features is limited.

C. Deep CNN Feature Learning Based Methods

The deep CNN feature learning-based method is to use a CNN model with a deep feature extraction structure to automatically learn more representative and discriminative features from the data. In recent years, due to the excellent performance of CNNs in the field of computer vision, many researchers proposed a variety of image classification methods based on CNNs [38]–[48]. In [14], Cheng *et al.* conducted experiments on the NWPU-RESISC45 dataset based on VGG16, AlexNet, and GoogleNet [49], and found that the classification performance based on CNN deep feature learning was far better than the first two methods.

In [50], Lu *et al.* proposed a remote sensing scene classification method based on an end-to-end feature aggregation CNN (FACNN). The semantic label was considered to learn the scene feature representation. In an FACNN, they proposed a supervised convolutional feature encoding module and a progressive aggregation strategy to leverage the semantic label information to aggregate the intermediate features, which improved the classification accuracy. Li *et al.* proposed a hybrid architecture, called aggregated deep Fisher feature (ADFF). In a pretrained CNN model, the optimal encoding layer was utilized, which naturally fused the local and global image information in a novel way, and thus the ability of semantic acquisition was further enhanced [51].

Cheng *et al.* proposed a simple but effective method based on three classic CNN depth models to learn discriminative CNN (DCNN). The proposed DCNN models were trained by optimizing a new discriminative objective function. The problems of high within-class differences and between-class similarities in remote sensing scene images were avoided effectively, and the classification accuracy was further improved [52]. He *et al.* proposed a multilayer stacked covariance pooling (MSCP). In this method, they used the pretrained network model to extract the multilayer convolution feature maps and to stack these feature maps, which was able to improve the classification accuracy [53].

In [54], a skip connected covariance network (SCCov) was proposed. This method embedded the modules of skip connections and covariance pooling into the CNN model, and superimposed the feature mapping information of different resolutions, which could effectively avoid the influence of different resolution images in the remote sensing datasets, and improved the classification accuracy. In [66], a simple yet effective method was proposed to extract hierarchical deep spatial features for hyperspectral image classification by exploring the power of off-the-shelf CNN models. In [69], a part-based CNN (P-CNN) was proposed for fine-grained visual categorization, which aimed to classify subordinate categories. In [70], a multisource compensation network was proposed to solve the problems of distribution differences and category incompleteness.

III. Methodology

A. Overall Structure of the Proposed Method

The proposed LCNN-BFF network was composed of nine parts (Groups 1–9), and this structure is shown in Fig. 2. According to the structure of the first three modules in the VGG16 network and the strategy of reducing model complexity introduced in this article, Groups 1–3 were defined. In Groups 1–3, the maximum pooling layer was used to downsample the remote sensing images, to reduce the spatial dimensions of the images, retain the main features of the images, and avoid the problem of overfitting. Groups 4–8 were mainly defined to extract representative features. Groups 4–7 used a bilinear convolution structure to extract more abundant feature information.

According to the bilinear convolution structure, the BFF method was proposed. BFF fuses the feature information extracted from the two branches to obtain more effective feature information. Part C of this section gives the difference between the feature maps extracted from the two branches and the feature maps fused using the BFF method. In addition, the number of convolution channels was increased in Groups 5 and 8 to widen the network so that it could learn more features [55], [56]. For the specific channel number settings of each group, please refer to Section IV-B. Group 9 was defined for classification, to convert the extracted feature information into the probability of each scene class.

In the feature extraction structure (Groups 1–8), lightweight convolution DSC and CConv were combined to extract image features, which greatly reduced the complexity of the model. Batch normalization (BN) [57] was used to normalize the output of each volume accumulation layer, and then the rectified linear unit function was used to activate the neurons. After BN processing, the learning speed of the model could be accelerated and converged quickly. To a certain extent, this can avoid the problem of the gradient disappearing with the deepening of the network,

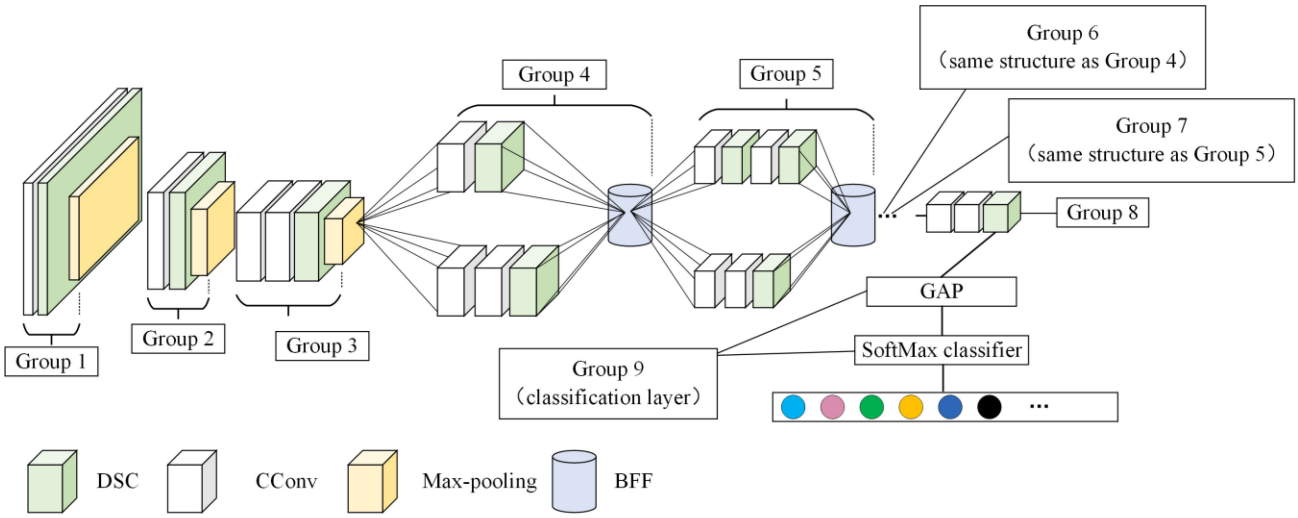


Fig. 2. Overall structure of the proposed LCNN-BFF method. DSC and conventional convolution (CConv).

and improve the generalization ability of the model. In addition, due to the small number of images in the divided training set, this may cause the problem of overfitting in the process of network training. Therefore, an L2 regularization penalty was added to the weight of the convolution layer, and the penalty coefficient was 0.0005.

In Group 9, the global average pooling (GAP) [58] was used instead of the flatten layer, which can reduce the model size and overfitting. Specifically, suppose the output result of the last convolution layer is $\mathrm{D} = [d_1; d_2; \ldots; d_i \ldots; d_N] \in \mathbb{R}^{N\times H\times W\times C}$, where [;;;] represents the cascade operation along the batch dimension, and $\mathbb{R}$ represents the real number set. In addition, *N*, *H*, *W,* and *C* represent the batch size, height, width, and channel number of the input data, respectively. If the output result of GAP layer is $\mathrm{G} = [g_1; g_2; \ldots; g_i \ldots; g_N] \in \mathbb{R}^{N\times C}$, the processing of gap layer to $\forall d_i \in \mathbb{R}^{H\times W\times C}$ can be represented as

$$g_i = \frac{\sum_{h=1}^{H}\sum_{w=1}^{W} d_i}{H \times W}. \tag{1}$$

From formula (1), GAP makes the feature mapping of the convolution output of the last layer more intuitively to each category. GAP summarizes the information of input space and performs a more robust operation on the input space information. As the weight parameters are not needed in the GAP layer, the overfitting phenomenon in the process of the training model can be reduced. In this article, the SoftMax classifier was used for classification. If the result of any output $g_i \in G$ processed by a fully connected (FC) layer with the number of cells as the number of classification *Z* is $\mathrm{V} \leftarrow [\,v_1\ v_2\ \ldots\ v_j\ \ldots\ v_Z\,] \equiv \mathrm{FC}(g_i)$, and the output result of SoftMax is $\mathrm{S} = [\,s_1\ s_2\ \ldots\ s_j\ \ldots\ s_Z\,]$, then the output result S of SoftMax classifier can be represented as

$$s_j = \frac{e^{\mathrm{V}[j-1]}}{\sum_{k=0}^{Z-1} e^{\mathrm{V}[k]}}. \tag{2}$$

Here, $\mathrm{V}[j-1]$ represents the *j*th element in V (index number starts from 0). The classified cross-entropy loss is adopted as a loss function. If $Q = [\,q_1\ q_2\ \ldots\ q_j\ \ldots\ q_Z\,]$ is used to represent the coding result of the input sample label, the loss function is

$$L = -\sum_{j=1}^{Z} q_j \log(s_j). \tag{3}$$

Here, *Z* represents the number of categories, s_j represents the output result of SoftMax, and the input sample label adopts the one-hot coding rule.

B. Strategy to Reduce Model Complexity

As the weight parameters and computation of the proposed method are mainly concentrated in the feature extraction structure, reducing the number of weight parameters and computation in the feature extraction process is an effective method to reduce the complexity of the model. In Groups 1–8 of the LCNN-BFF network, two convolution methods, CConv and light convolution DSC, were used to extract depth representative image features. Compared with CConv, the advantages and disadvantages of DSC are described as follows. Suppose an RGB image M, $\mathrm{M} \in \mathbb{R}^{W\times W\times 3}$, and $W \times W$ represents the size of the image. Without considering bias, there are *K* convolution kernels with $f \times f$ size. The convolution step size *S* is 1, and the filling size *Pad* is 0. The convolution process of CConv is that each convolution kernel is convoluted with each channel of the image, and one convolution kernel is convoluted with the feature maps of three channels to obtain three tensors. The result of adding these three tensors is a 2-D feature map. Suppose A_K is the *k*th convolution kernel, the *k*th 2-D feature map of convolution output can be represented as

$$B_{\mathrm{K}} = A_K \otimes \mathrm{M} \tag{4}$$

where "⊗" represents the convolution operator. If the size of the output feature map calculated by $T = \frac{W-f+2\mathrm{Pad}}{S} + 1$ is $T \times T$, the output M_{co} of image M after CConv can be represented as

$$M_{co} = [B_1; B_2; B_3, \ldots; B_K] \in \mathbb{R}^{T\times T\times K} \tag{5}$$

where [;;;] represents a cascading operation along the channel dimension.

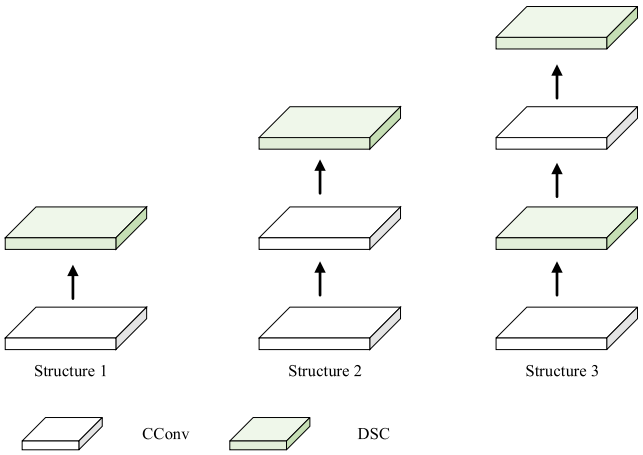


Fig. 3. Three combined structures of convolution.

Different from CConv, the process of DSC includes the deep convolution process and the point-by-point convolution process. In the process of deep convolution, an image is divided into components based on the number of input channels, and then each component is convoluted with different convolution kernels. Each component of the image is $M_i \in \mathbb{R}^{W\times W}, i = 1, 2, 3$. The three components are convoluted with three different convolution kernels ($U_i \in \mathbb{R}^{f\times f}, i = 1, 2, 3$) with $f \times f$ size. The results can be represented as

$$O_i = M_i \otimes U_i, i = 1, 2, 3 \tag{6}$$

$$O = [O_1; O_2; O_3] \in \mathbb{R}^{T\times T\times 3}. \tag{7}$$

The point-by-point convolution process is used to convolute the output of the deep convolution process O with K convolution kernels of 1×1 size. The results of each feature map are shown in (4) and (5). The dimension of the DSC result $M_{do} \in \mathrm{R}^{T\times T\times K}$ is the same as that of the CConv result. The number of weight parameters of two convolutions is different. The number of weight parameters of CConv can be represented as

$$P_c = C_i \times f \times f \times C_o. \tag{8}$$

The number of weight parameters of DSC can be represented as

$$P_d = C_i \times f \times f + 3C_o \tag{9}$$

where C_i is the number of input channels, $f \times f$ is the convolution kernel size of the deep convolution process, and C_o is the number of output channels. If $C_i = 3, f = 3,$ and $C_o = 256$, the numbers of weight parameters of CConv and DSC are 6912 and 795, respectively. The number of weight parameters of DSC is only 11.5% that of CConv.

From the above, DSC can be seen to effectively reduce the complexity of the model and, however, is not suitable to extract image features alone. Due to the large reduction of the number of weight parameters, it may lead to insufficient or incorrect learning of the network model. Therefore, this article proposes three ways of combining DSC and CConv to extract image features, as shown in Fig. 3. The three convolution structures are based on the network structure of VGG. These convolution structures first extract image features using CConv, and then extract deeper feature information using DSC.

According to the LCNN-BFF network structure shown in Fig. 2, structure 1 is mainly used for Groups 1, 2, 4, and 6; structure 2 is mainly used for Groups 3–8; and structure 3 is mainly used for Groups 5 and 7. These combined structures can not only extract more representative features, but also effectively avoid the learning problems caused by the reduction of the number of weight parameters. In addition, in order to further improve the classification performance, a BFF method was proposed. This can fuse and complement different feature information extracted from two branches, which plays a great role in dealing with the problem of high within-class difference and between-class similarity.

C. Fusion of Branch Features

In Fig. 2, the BFF method is utilized in Groups 4–7 to fuse the information of the final convolution results of two branches after BN processing. Specifically, it can be seen from [57] that the convolution output of the last layer of a branch is assumed to be $\mathrm{X}_c \in \mathbb{R}^{N\times H\times W\times C}$. Take data $\mathrm{X} = [x_1 \dots x_m] \in \mathbb{R}^{N\times H\times W}$ in a channel, and the result after BN layer processing can be represented as

$$m = N \times H \times W \tag{10}$$

$$\mu_{\mathrm{X}} = \frac{1}{m}\sum_{i=1}^{m} x_i \tag{11}$$

$$\sigma_{\mathrm{X}}^2 = \frac{1}{m}\sum_{i=1}^{m} (x_i - \mu_{\mathrm{X}})^2 \tag{12}$$

$$\hat{x}_i = \frac{x_i - \mu_{\mathrm{X}}}{\sqrt{\sigma_{\mathrm{X}}^2 + \varepsilon}} \tag{13}$$

$$y_i = \gamma\hat{x} + \beta \tag{14}$$

where m is the number of values of X, μ_{X} represents the mean value of this Group of data, σ_{X}^2 represents the variance of this Group of data, $\hat{x}_i$ represents the result of standardizing each number in data X, y_i represents the result of this group of data processed by BN, and γ and β are a pair of learnable parameters. After C cycles, the data of all channels are standardized and represented by $Y_c \in \mathbb{R}^{N\times H\times W\times C}$. Taking Group 4 as an example, suppose $N = 1$. If $Y_{4,1} \in \mathbb{R}^{H\times W\times C}$ and $Y_{4,2} \in \mathbb{R}^{H\times W\times C}$ are used to represent the 3-D feature maps of the output of the first and second branches in Group 4, then the 2-D feature map of any channel in $Y_{4,1}$ and $Y_{4,2}$ can be represented as

$$\xi_{b,i} = Y_{4,b}[:, :, i] \tag{15}$$

$$\forall i \in \{0, 1, 2, \dots, C-1\} \tag{16}$$

where $Y_{4,b}[:, :, i]$ is the $(i + 1)$th 2-D feature map along the channel dimension in 3-D feature maps (the index number starts from 0). BFF superimposes the corresponding units from the first to the Cth 2-D feature map of $Y_{4,1}$ and $Y_{4,2}$ to realize feature fusion, which can be represented as

$$\sum_{i=0}^{C-1}\sum_{m=0}^{H-1}\sum_{n=0}^{W-1} \xi_{1,i}[m, n] + \xi_{2,i}[m, n]. \tag{17}$$

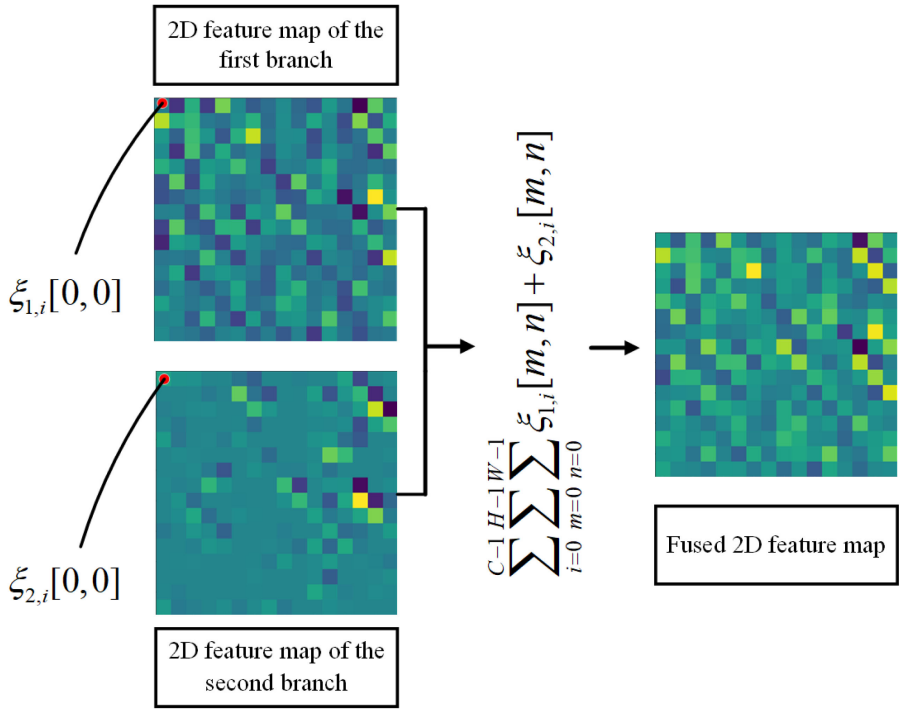


Fig. 4. Schematic diagram of the BFF method.

The schematic diagram of the BFF method is shown in Fig. 4. In Fig. 4, the BFF method was used to fuse and complement the feature information of each unit in the feature mapping process of two branches. BFF can effectively avoid the problem that some features are not fully extracted or even not extracted. Certain feature maps extracted by two branches in Group 4 and that after BFF fusion are shown in Fig. 5. The dark areas of the feature maps in Fig. 5 are the extracted features from these scenes. It can be seen from Fig.5 that for the same scene, the first branch and the second branch extract different features, and the effect of feature extraction is related to the scene. For example, the first branch is more effective than the second branch in extracting beach and ship scenes, and the second branch has more advantages in extracting airplane and overpass scenes.

In this article, the features of two branches are fused with the BFF method, which can extract more abundant feature information than that of the two branches. Particularly, for the bridge, golf course, island, and rectangular farm scenes, the features extracted from the first and second branches are part of the main features of these scenes. BFF method fuses the features extracted from the two branches, and obtains the main features in these scenes. To sum up, the two branches can extract different features of the image, but the extracted feature information is only a part of the main features. BFF can fuse and complement the feature information extracted from the two branches and obtain more complete features, which can effectively improve the classification performance of remote sensing scene images. In addition, BFF does not require parameters, which makes it more appealing.

IV. Experiment and Result Analysis

In this section, we comprehensively evaluated the proposed LCNN-BFF method using different methods. Experiments were performed on four datasets with high challenge. The performance of the proposed method was compared with that of state-of-the-art methods. The experimental results verified the effectiveness of the proposed method.

A. Datasets

The UC Merged Land Use (UC) dataset [59] contains 2100 remote sensing scene images, which are divided into 21 scene classes. Each class contains 100 aerial images with 256 × 256 pixels and the spatial resolution of the images is 1 ft. Scene examples of this dataset are shown in Fig. 6. In the experiment, 80% of the images of each scene class were randomly selected as a training set, and the rest were divided into the test set (80/20 UC).

The RSSCN7 (RSSCN) dataset [60] contains 7 scene classes, with a total of 2800 remote sensing scene images. Each class contains 400 images with 400 × 400 pixels. These images come from different seasons and weather changes and were sampled at four different scales. Examples of the scene images in this dataset are shown in Fig. 7. In the experiment, the image size was adjusted to 256 × 256. We randomly selected 50% of the images in each scene class as the training set, and the rest were divided into the test set (50/50 RSSCN).

The Aerial Image dataset (AID) [61] is composed of 30 scene classes and 10 000 remote sensing scene images. Each scene class contains 220–420 scene images with 600 × 600 pixels, and

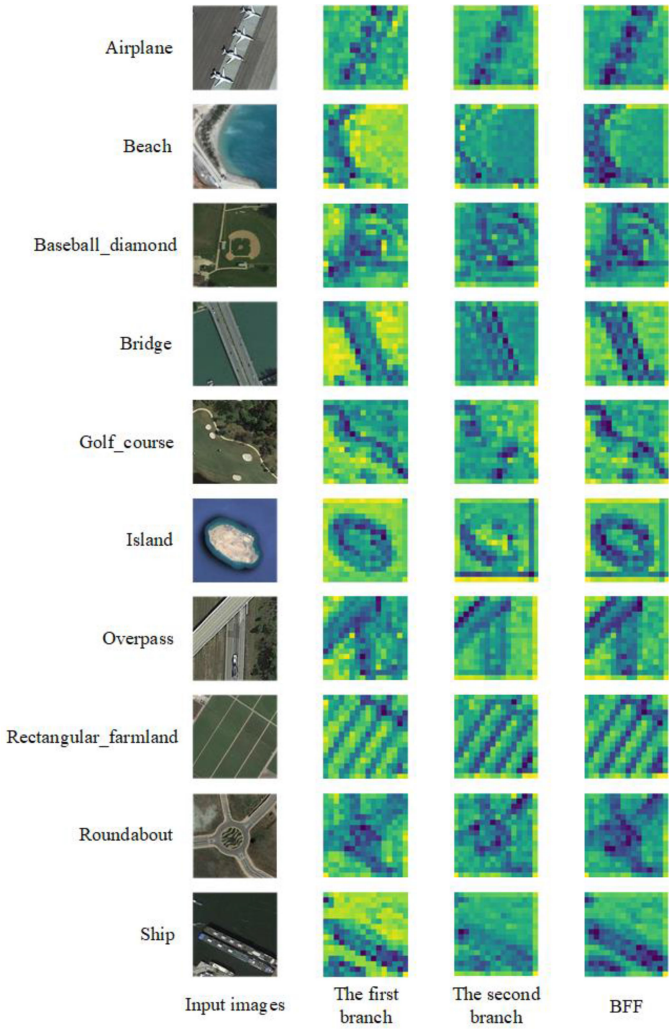


Fig. 5. Feature maps of some example images.

Fig. 6. Scene examples from the UC Merged Land Use (UC) dataset.

the spatial resolution changes from about 8 to 0.5 m. Examples of scene images of this dataset are shown in Fig. 8. In the experiment, the image size was adjusted to 256 × 256. 20% and 50% of the images of each scene class were randomly selected as training sets, and the rest were divided into test sets (20/80 AID, 50/50 AID).

The NWPU-RESISC45 (NWPU) dataset [14] consists of 45 scene classes, with a total of 31 500 remote sensing scene images. Each scene class contains 700 scene images with 256 × 256 pixels. The spatial resolution of most scene images varies from 30 to 0.2 m. This dataset is one of the largest in both the number of scene categories and the total number of scene images. The

Fig. 7. Scene examples from the RSSCN7 (RSSCN) dataset.

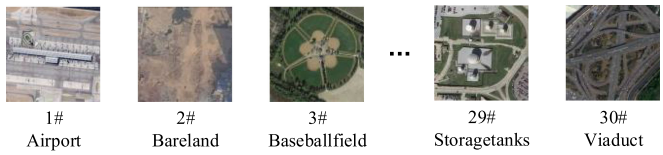


Fig. 8. Scene examples from the Aerial Image dataset (AID).

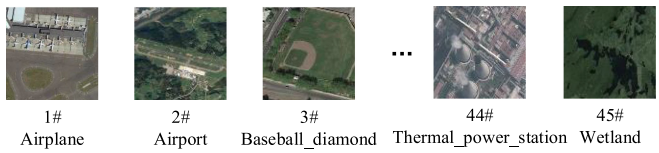


Fig. 9. Scene examples from the NWPU-RESISC45 (NWPU) dataset.

rich changes of images lead to high within-class differences and between-class similarities, which makes this dataset more challenging. Scene image examples from this dataset are shown in Fig. 9. In the experiment, 10% and 20% images of each scene class were randomly selected as training sets, and the rest were divided into test sets (10/90 NWPU, 20/80 NWPU).

B. Setting of Experiments

The setting of the LCNN-BFF network was as follows. In Group 1, the number of convolution channels was set to 32, and the convolution kernel size of each convolution was set to 3 × 3. In Group 2, the number of convolution channels was set to 64, and the convolution kernel size of each convolution was set to 3 × 3. In Group 3, the number of convolution channels was set to 128, the convolution kernel size of the first convolution was set to 1 × 1, and those of the remaining convolutions were set to 3 × 3. The pool sizes of the max-pooling layers in Groups 1–3 were set to 2 × 2, and pooling stride was 2. The number of convolution channels in Group 4 was the same as that in Group 3. In the two branches of Groups 4–7, the convolution kernel size of the first convolution was set to 1 × 1, and those of the other convolutions were set to 3 × 3. The last convolution stride of the two branches was set to 2, and the remaining convolution strides were set to 1. The number of convolution channels of Groups 5–7 was set to 256, and Group 8 was set to 512. In Group 8, the convolution kernel size of the first convolution was set to 1 × 1, the convolution kernel sizes of the remaining convolutions were set to 3 × 3, and the convolution stride was set to 1.

Data enhancement was employed for the datasets as follows.

1) For the input image, the rotation range was 0–60°.
2) The length and width of the input image were offset randomly, and the offset coefficient was 0.2.

TABLE I
COMPARISON OF LCNN-BFF AND MOBILENET WITH THE OA AND KAPPA

	LCNN-BFF		MobileNet	
	OA (%)	Kappa (%)	OA (%)	Kappa (%)
80/20 UC	**99.29**	**99.25**	97.62	97.50
50/50 RSSCN	**94.64**	**93.75**	91.50	90.08
20/80 AID	**91.66**	**91.37**	87.21	86.76
50/50 AID	**94.62**	**94.41**	92.12	91.84
10/90 NWPU	**86.53**	**86.22**	82.66	82.27
20/80 NWPU	**91.73**	**91.54**	87.85	87.57

The bold entity represents the method with the best performance on a validation metric.

3) We randomly flipped the input image horizontally and vertically.

After the data enhancements, all the samples were normalized by batch. In addition, to avoid memory overflow during training, the sizes of the input images were adjusted to 256 × 256.

The initial learning rate of training the LCNN-BFF network was set to 0.01. An automatic learning rate reduction mechanism was added. In the training process, the batch size was set to 16 and the proposed LCNN-BFF was optimized with a momentum optimization algorithm, and the momentum coefficient was set to 0.9. All the experimental results are the average values after ten experiments. The computer configuration was as follows: RAM: 8 GB; Processor: Intel (R) Pentium (R) CPU G4600 @ 3.60 GHz; GPU: NVIDIA GeForce GTX 4G 1050 Ti.

C. Performance of the LCNN-BFF Method

In order to verify the performance of the proposed LCNN-BFF method, evaluation indexes, including the overall accuracy (OA), average precision (AP), kappa coefficient (Kappa), F1 score (F1), confusion matrix, average training time (ATT), and weight parameters (parameters), were utilized in the following experiments. OA represents the percentage of correctly classified images in the total test set. AP represents the average of the accuracy rate of each scenario class on the test set. ATT represents the average time that a model processes an image during training.

MobileNet [27] is the basic network of the proposed LCNN-BFF method. In this work, in order to verify the effectiveness of bilinear convolution structure and the BFF method, some experiments were conducted using MobileNet and the LCNN-BFF method on the UC, RSSCN, AID, and NWPU datasets. The OA, AP, Kappa, F1, and confusion matrix were chosen as the evaluating indices.

In this article, Keras was used to reproduce the MobileNet network and the parameters of the last layer were fine-tuned. The classification performance of the LCNN-BFF and MobileNet networks evaluated by OA and kappa indexes on six datasets are listed in Table I. In Table I, the OA and Kappa values of the proposed method were significantly better than those of MobileNet. The performance of LCNN-BFF on the UC dataset was very good, with the OA and kappa values at 99.29% and 99.25%, respectively. In addition, for the NWPU and AID datasets with less data training, the performance advantage of LCNN-BFF was more prominent, which shows that the proposed method is more robust. Next, the performance of the proposed method was evaluated by AP, F1, and a confusion matrix.

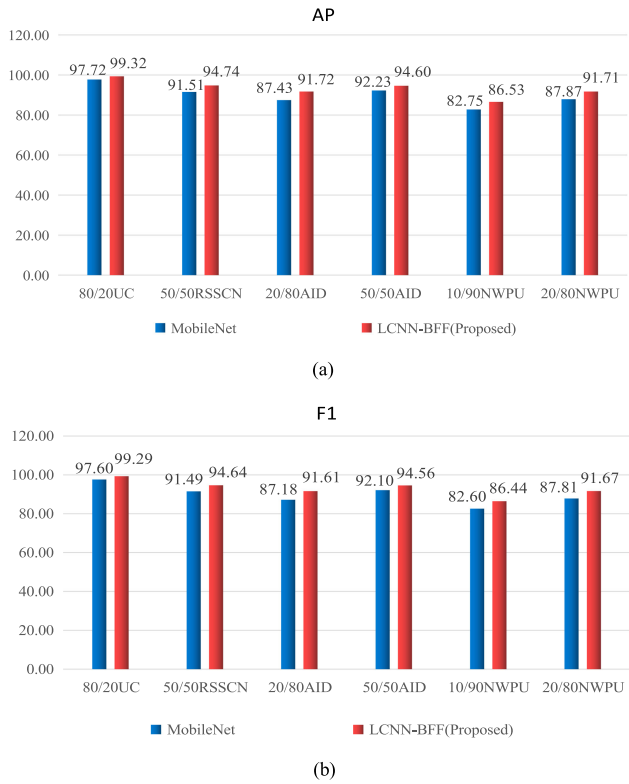


Fig. 10. Performance comparison of LCNN-BFF and MobileNet. (a) Comparison of LCNN-BFF and MobileNet on AP. (b) Comparison of LCNN-BFF and MobileNet on F1.

The AP and F1 results of the proposed LCNN-BFF method and MobileNet are shown in Fig. 10. In Fig. 10(a), the AP values of the LCNN-BFF method are all higher than those of MobileNet for different datasets. The classification performance of the LCNN-BFF method on the 50/50RSSCN, 20/80AID, 10/90NWPU, and 20/80NWPU datasets was excellent, with AP values at 3.59%, 4.29%, 3.78%, and 3.84% higher than those of MobileNet. The classification performance of the LCNN-BFF method in Fig. 10(b) was also good, especially on the test sets of 20/80AID, 10/90NWPU, and 20/80NWPU. The F1 scores of the LCNN-BFF were 4.43%, 3.84%, and 3.86% higher than those of MobileNet. As shown in Fig.10, the classification performance of the proposed LCNN-BFF method was always better than that of the MobileNet network, and the proposed method was more robust.

The confusion matrices of the proposed LCNN-BFF method and MobileNet tested on the 80/20UC and 50/50RSSCN datasets are shown in Figs. 11 and 12. The values on the diagonal of the confusion matrix represent the precision of each class, and other values in the same row represent the percentage of misclassification. In Fig. 11, we can see that the classification error rate of the LCNN-BFF method was significantly lower than that of MobileNet. In Fig. 12, the classification accuracies of each class of the proposed LCNN-BFF method were higher than

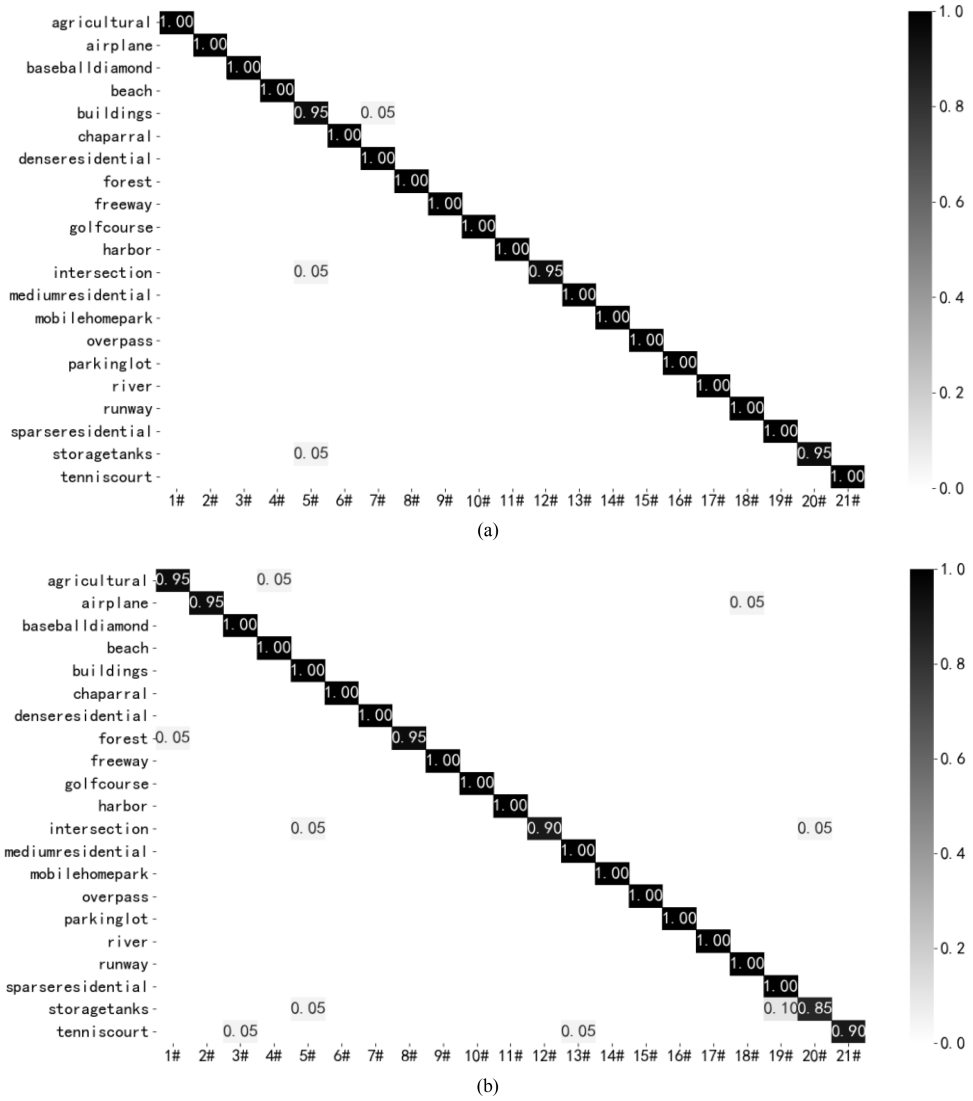


Fig. 11. Confusion matrices of LCNN-BFF and MobileNet on the 80/20 UC test set. (a) Confusion matrix of LCNN-BFF. (b) Confusion matrix of MobileNet.

or equal to those of MobileNet. This proved that the proposed BFF method was very effective in dealing with the problems of high within-class differences and between-class similarities in remote sensing scene images.

The above experiments proved the effectiveness of the proposed method using the OA, kappa, AP, F1, and confusion matrix. This proved that the proposed bilinear convolution structure and BFF method can significantly improve the classification performance of remote sensing scene images. In addition, the bilinear convolution structure and BFF method also improved the robustness of the classification network.

D. Performance Comparison With the State-of-the-Art Methods

In this section, the proposed LCNN-BFF method was compared with state-of-the-art methods for remote sensing scene classification in terms of the model complexity and classification accuracy. Experiments were performed on the UC, RSSCN, AID, and NWPU datasets. The OA, parameters, Kappa, and ATT were used to evaluate these methods. The methods for comparison can be divided into two categories. One category is unsupervised feature learning-based methods. In [21], a variable-weighted multifeature fusion (VWMF) classification method based on kernel collaborative representation-based classification (KCRC) and the support vector machine (SVM) was proposed to solve the problems of high within-class differences and between-class similarities of remote sensing scene images.

In [44], a multiresolution block feature (MRBF) classification method based on completed double cross pattern (CDCP) and Fisher vectors was proposed to solve the problem of complicated and various scene types and complex backgrounds in scene images. In [71], Yan *et al.* introduced semisupervised representation learning (SSRL) into a generative adversarial network (GAN), to have the discriminator learn more discriminative features from labeled data and unlabeled data. The mixed data augmentation method was introduced into their classification model to augment the data and stabilize the training process. As

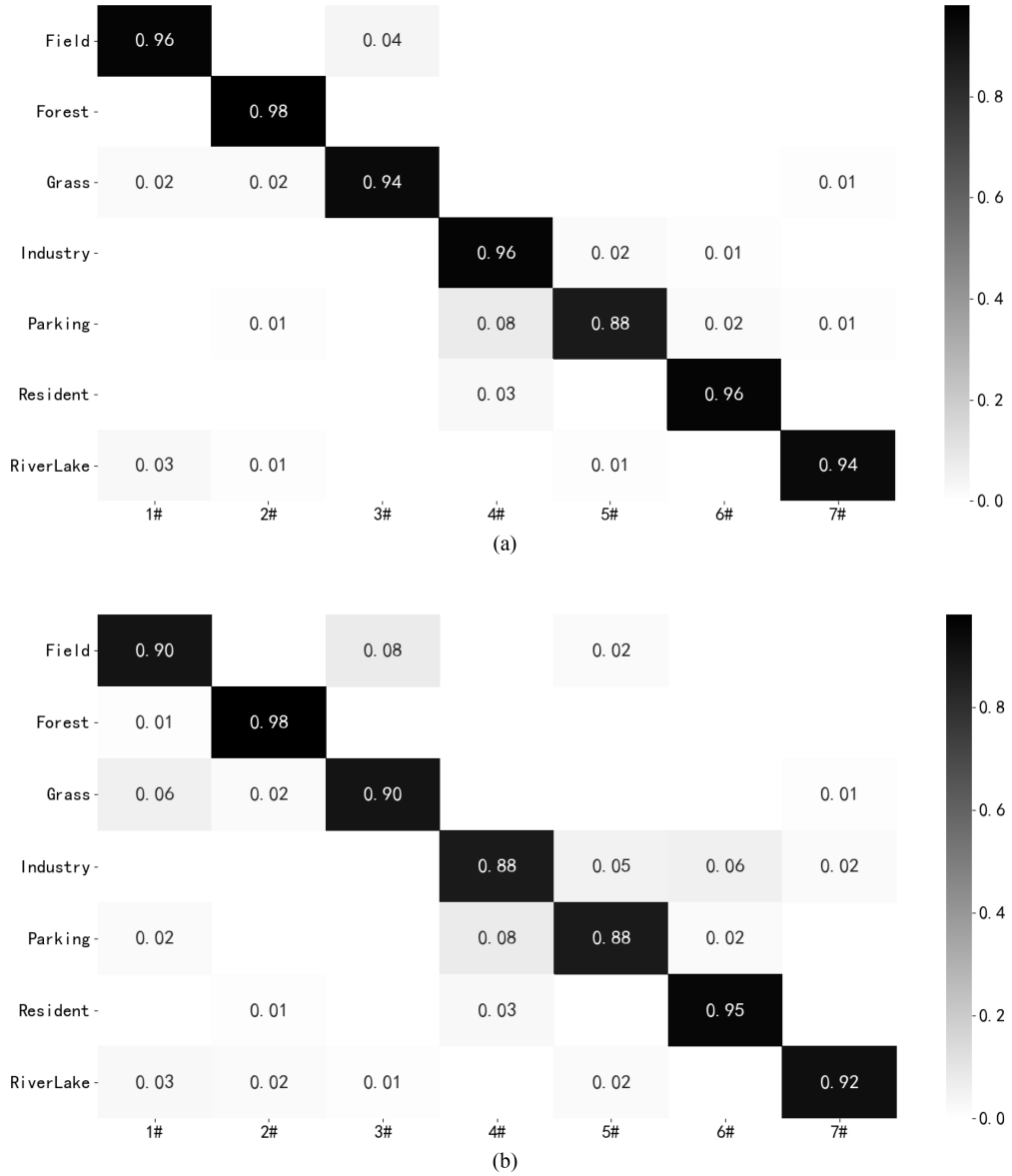


Fig. 12. Confusion matrices of LCNN-BFF and MobileNet on the 50/50 RSSCN test set. (a) Confusion matrix of LCNN-BFF. (b) Confusion matrix of MobileNet.

the two methods have difficulty in extracting more representative features in remote sensing scene images, the classification accuracy was not high.

The other category is deep CNN feature learning-based methods, mainly including those presented in [39], [42], [48], [50], [54], [61], and [63]. Among them, in [47], a Siamese CNN method based on the CNN recognition and verification model was proposed, aiming at the existing methods that have difficulty in extracting rich remote sensing scene label information. On this basis, a classification method of rotation-invariant feature learning and joint decision-making method (R.D.) based on the Siamese CNN was proposed in [43], to improve the classification accuracy.

The above two methods do not fully consider the important spatial information of remote sensing scene images, as well as the problems of high within-class differences and between-class similarities, which limits the further improvement of the classification performance. To solve these problems, a new classification structure, called CNN-CapsNet, was proposed in [42], and this method combines the advantages of CNN and capsule network (CapsNet). In [62], a gated bidirectional network (GBNet) was proposed for remote sensing scene classification. In [50], an end-to-end FACNN was proposed for remote sensing scene classification.

These three methods greatly improved the classification accuracy; however, the model complexity remained relatively high. In view of the complexity of the model, in [51], some optimal coding layers in the pretraining networks were explored, and an ADFF-based method was proposed. In [53], a classification method of MSCP was proposed. In [54], a classification method based on a new CNN was proposed, which combines the skip connect and the covariance pooling (SCCov).

In [72], Cao *et al.* proposed a new method, called self-attention-based deep feature fusion (SAFF), to aggregate deep

TABLE II
COMPARISON OF THE OA AND WEIGHT PARAMETERS ON THE 80/20 UC TEST SET

Basic method/network	Methods	OA (%)	Parameters
KCRC and SVM	VWMF [21]	97.79	—
VGG16	BAFF [39]	95.48	130M
AlexNet	MCNN [41]	96.66±0.90	60M
Inception-v3	Inception-v3-CapsNet [42]	99.05±0.24	22M
Siamese ResNet50	R.D [43]	94.76	—
CDCP	MRBF [44]	94.19±1.5	—
VGG16	VGG16-DF [45]	98.97	130M
VGG16	SF-CNN with VGGNet [46]	99.05±0.27	130M
ResNet50	Siamese [47]	94.29	—
ResNet50	WSPM-CRC [48]	97.95	23M
VGG16	FACNN [50]	98.81±0.24	130M
ResNet50	ADFF [51]	98.81±0.51	23M
VGG16	DCNN [52]	98.93±0.10	130M
VGG16	MSCP [53]	98.36±0.58	—
AlexNet	SCCov [54]	98.04±0.23	**6M**
VGG16	SCCov [54]	99.05±0.25	13M
VGG16	GBNet + global feature [62]	98.57±0.48	138M
GAN	SSRL [71]	94.05±1.2	210M
VGG16	VGG_VD16 + SAFF [72]	97.02±0.78	15M
ResNet50	PANet50 [73]	99.21±0.18	28M
MobileNet	LCNN-BFF(Proposed)	**99.29±0.24**	**6M**

The bold entity represents the method with the best performance on a validation metric.

layer features and emphasize the weights of the complex objects of remote sensing scene images for remote sensing scene classification. In [73], a general positional context aggregation (PCA) module in deep CNNs was proposed. The PCA module has the form of a self-attention mechanism, in which two proposed blocks, the spatial context aggregation and the relative position encoding, are used to capture the spatial-dipartite contextual aggregation information and the relative position encoding information.

Compared with these methods, in this article, we utilized the bilinear convolution structure to extract the rich feature information of the scene image, and explore the BFF method to fuse and complement the differing feature information extracted from the two branches, which greatly improved the classification accuracy. In terms of model complexity, three feature extraction structures were proposed by combining DSC and CConv convolution, which greatly reduced the complexity of the model. Next, the performance comparison of different classification methods was explored with experiments.

First, some comparative experiments were performed on the UC dataset. The results of the OA and the size of the parameters of the proposed method and state-of-the-art methods are listed in Table II. In Table II, the average OA of the proposed method was higher than those of the other methods. Among them, the average OA of the proposed method was 0.24% higher than those by the methods in [42], [46], and [54]. From the perspective of the parameters, the parameters of LCNN-BFF only accounted for 4.61%, 27.27%, and 46.15% of [46], [42], and [54], respectively. The reason is that, for the proposed LCNN-BFF network, the feature extraction structure combines two convolution methods, i.e., light convolution DSC and CConv, which greatly reduced the size of the parameters and improved the classification accuracy.

In addition, compared with the current state-of-the-art methods proposed in [62] and [71]–[73], the proposed method demonstrated better performance, both in the classification accuracy and weight parameters. The results of the Kappa values of the proposed method and the state-of-the-art methods on the UC dataset are listed in Table III. In Table III, the average Kappa of the proposed method was 99.25%. The performances of the proposed method on Kappa were 31.25%, 7.25%, and 5.25% higher than those of the three methods in [47], respectively, and 30.25%, 6.25%, 4.75% higher than those of the three methods in [43], respectively. With the experimental results of Tables II and III, we proved that LCNN-BFF had a good classification performance on the UC dataset from the perspective of the Kappa index.

TABLE III
COMPARISON OF THE KAPPA ON THE 80/20 UC TEST SET

Basic method/network	Methods	Kappa (%)
Siamese AlexNet	R.D [43]	69.00
Siamese VGG16	R.D [43]	93.00
Siamese ResNet50	R.D [43]	94.50
AlexNet	Siamese [47]	68.00
VGG16	Siamese [47]	92.00
ResNet50	Siamese [47]	94.00
MobileNet	LCNN-BFF(Proposed)	**99.25**

The bold entity represents the method with the best performance on a validation metric.

TABLE IV
COMPARISON OF THE OA AND ATT ON THE 80/20 UC TEST SET

Basic method/network	Methods	ATT (s)
VGG16	Siamese [47]	0.053
ResNet50	Siamese [47]	0.039
VGG16	GBNet + global feature [62]	0.052
VGG16	GBNet [62]	0.048
MobileNet	LCNN-BFF(Proposed)	**0.029**

The bold entity represents the method with the best performance on a validation metric.

TABLE V
COMPARISON OF THE OA AND PARAMETERS ON THE 50/50 RSSCN TEST SET

Basic method/network	Methods	OA (%)	Parameters
KCRC and SVM	VWMF [21]	89.1	—
ResNet50	SPM-CRC [48]	93.86	23M
ResNet50	WSPM-CRC [48]	93.9	23M
ResNet50	ADFF [51]	**95.21±0.50**	23M
VGG16	VGG16 + SVM [61]	87.18	130M
CaffeNet and VGG16	Two-stage deep feature fusion [63]	92.37±0.72	—
MobileNet	LCNN-BFF(Proposed)	94.64±0.21	**6M**

The bold entity represents the method with the best performance on a validation metric.

In order to further verify the effectiveness of the proposed method, we compared the ATT with several state-of-the-art methods on the UC dataset. The comparison results between the proposed method and the state-of-the-art methods in terms of the ATT are listed in Table IV. It can be seen from Table IV that the proposed method took less time than the compared methods. The proposed LCNN-BFF network processed an image in 0.029, 0.023, and 0.019 s faster than the two methods in [62]. The ATT of the proposed method was tested under the condition where the computer configuration was inferior to the two methods being compared. Therefore, the ATT result of the proposed method is not optimal. If better experimental equipment was used, ATT would be smaller.

Second, experiments are conducted on the RSSCN dataset. The results of the OA and parameters of the proposed method and the state-of-the-art methods are listed in Table V. In Table V,

TABLE VI
COMPARISON OF THE OA AND PARAMETERS ON THE AID TEST SETS

Basic method/network	Methods	OA (20/80) (%)	OA (50/50) (%)	Parameters
VGG16	BAFF [39]	—	93.56	130M
VGG16	VGG16-CapsNet [42]	91.63±0.19	94.74±0.17	130M
VGG16	FACNN [50]	—	95.45±0.11	130M
AlexNet	DCNN [52]	85.62±0.10	94.47±0.12	60M
VGG16	DCNN [52]	90.82±0.16	**96.89±0.10**	130M
AlexNet	MSCP [53]	88.99±0.38	92.36±0.21	—
VGG16	MSCP [53]	91.52±0.21	94.42±0.17	—
AlexNet	SCCov [54]	91.10±0.15	93.30±0.13	**6M**
VGG16	Fine-tuning [61]	86.59±0.29	89.64±0.36	130M
VGG16	GBNet [62]	90.16±0.24	93.72±0.34	18M
VGG16	GBNet + global feature [62]	**92.20±0.23**	95.48±0.12	138M
CaffeNet and VGG16	Two-stage deep feature fusion [63]	—	91.8	—
VGG16	VGG_VD16 + SAFF [72]	90.25±0.29	93.83±0.28	15M
MobileNet	LCNN-BFF(Proposed)	91.66±0.48	94.62±0.16	**6M**

The bold entity represents the method with the best performance on a validation metric.

TABLE VII
COMPARISON OF THE OA AND PARAMETERS ON THE NWPU TEST SETS

Basic method/network	Methods	OA (10/90) (%)	OA (20/80) (%)	Parameters
VGG16	VGG16-CapsNet [42]	85.08±0.13	89.18±0.14	130M
Siamese VGG16	R.D [43]	—	91.03	—
AlexNet	DCNN [52]	85.56±0.20	87.24±0.12	130M
VGG16	DCNN [52]	**89.22±0.50**	**91.89±0.22**	130M
AlexNet	MSCP [53]	81.70±0.23	85.58±0.16	—
VGG16	MSCP [53]	85.33±0.17	88.93±0.14	—
AlexNet	SCCov [54]	84.33±0.26	87.30±0.23	**6M**
VGG16	VGG_VD16 + SAFF [72]	84.38±0.19	87.86±0.14	15M
MobileNet	LCNN-BFF(Proposed)	86.53±0.15	91.73±0.17	**6M**

The bold entity represents the method with the best performance on a validation metric.

compared with the methods in [21], [48], [58], and [63], the proposed method had better classification performance. In terms of the OA, the proposed method was 7.46%, 5.54%, 2.27%, 0.78%, and 0.74% higher than the methods in [61], [21], and [63], and the two methods in [48], respectively. In terms of the weight parameters, the total parameters of the proposed method only accounted for 4.48% of the method in [61] and 26.09% of the two methods in [48]. In addition, the average OA of the proposed method was slightly lower than that of the ADFF method [51]. However, the amounts of parameters of the proposed method only accounted for 26.09% of the ADFF method.

Finally, we performed experiments on the AID and the NWPU dataset. The comparison results of the proposed method and state-of-the-art methods are listed in Tables VI and VII, respectively. It can be seen from Table VI that whether the training and testing ratio was 2:8 or 5:5, the average OAs of the proposed method were higher than those of the other methods. The OA of the proposed method was slightly lower than that of the optimal method proposed in [50], [52], and [62], but in terms of the total parameters of the weight parameters, the proposed method only required 4.62% of the total parameters of the optimal method in [50] and [52], and 4.35% of the total parameters of the optimal method in [62]. The method in [54]

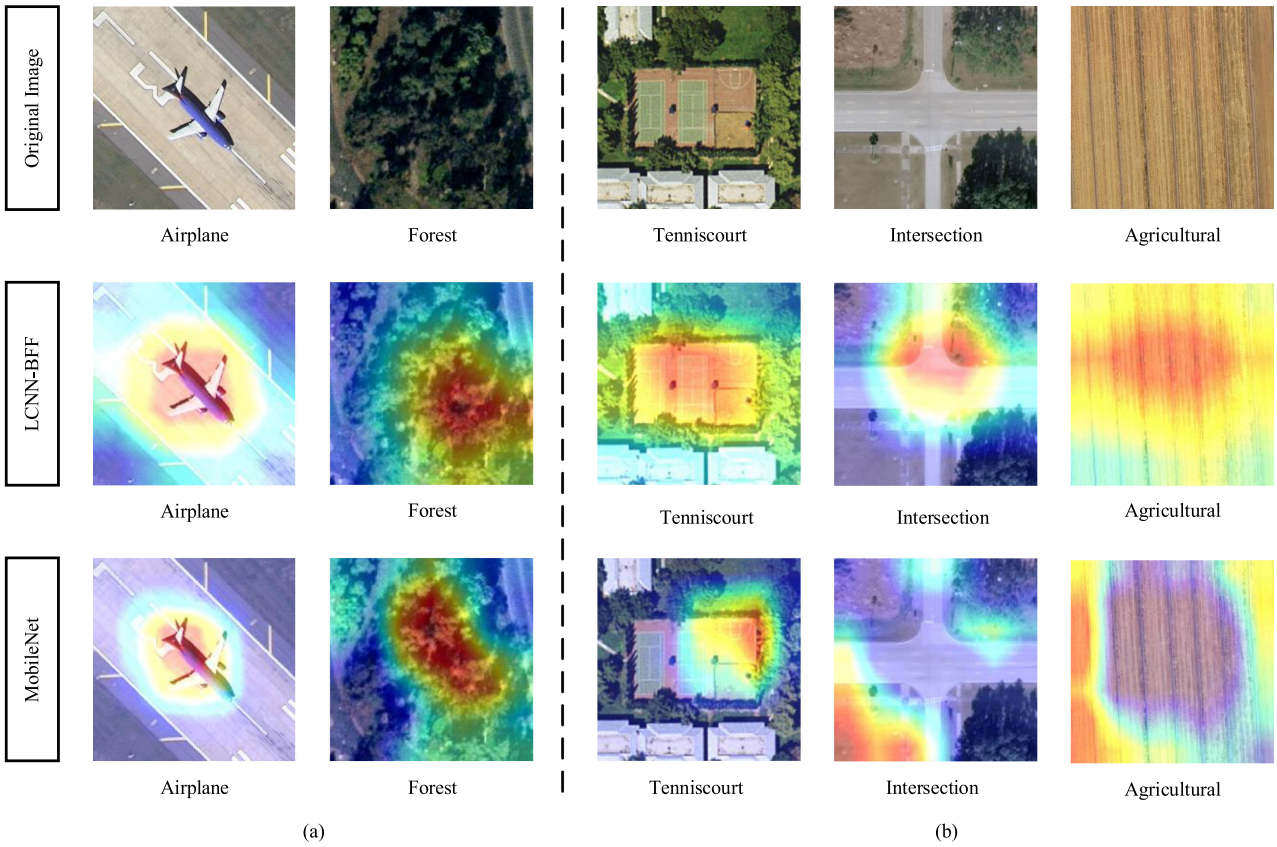


Fig. 13. Visualization results of LCNN-BFF and MobileNet on the UC dataset. (a) Scene images classified correctly by both methods. (b) Scene images classified correctly only by LCNN-BFF.

had almost the same amounts of parameters as the proposed method; however, the OA of was low.

In Table VII, we can see that the OA of the proposed method was only slightly lower than that of the optimal method in [52]. For the 1:9 and 2:8 training test ratios, the average OA of the proposed method was 2.69% and 0.16% lower than that of the optimal method in [52]. However, only 4.61% of the amounts of parameters of the optimal method in [52] was required for the proposed method. Most of the comparison methods with high classification performance in this article adopted the pretraining method. In [74], Chen *et al.* indicated that the pretraining CNN-based models could achieve significantly better classification performance compared to the CNN models trained from scratch.

The proposed method was directly trained on the remote sensing scene dataset. In most cases, this method could still obtain higher classification accuracy with lower complexity. This further proved the effectiveness of the proposed method. The above-mentioned experimental data demonstrated that the proposed method was a simple and effective method for remote sensing scene classification, in which the complexity and classification accuracy of the model were considered simultaneously, and were very suitable for practical application and development.

E. Visual Results of Different Methods

In order to further analyze the proposed method, in this section, we discuss which part of the feature information of the scene image is utilized to make the correct classification decision. For visualizing the outputs of the last convolution of LCNN-BFF and MobileNet, we selected representative images from UC and NWPU datasets, and input them into the LCNN-BFF and MobileNet networks, respectively. The gradient-weighted class activation mapping (Grad-CAM) method [64] was adopted for visualization. The visualization results of the proposed method and MobileNet on the UC and NWPU datasets are shown in Figs. 13 and 14, respectively.

It can be seen in Figs. 13 and 14 that the six scene images, including "Tennis court," "intersection," "agricultural," "storage tank," "stadium," and "roundabout," were misclassified as other categories by MobileNet. Clearly, the feature regions of interest (FROIs) extracted by MobileNet were not accurate. The reasons are as follows.

1) Most of the FROIs were the edges of the main features.
2) The extracted FROIs did not represent the main features of this category.
3) The extracted FROIs were not abundant and lacked additional discrimination information.

We use example images to demonstrate this. For "agricultural," the extracted FROIs by MobileNet were the edges of the main features of the image, and these regions were scattered. For "intersection," the MobileNet could not extract the main features of this kind of image at all. For "Roundabout," the feature regions extracted by MobileNet were incomplete and lacked other feature information for classification.

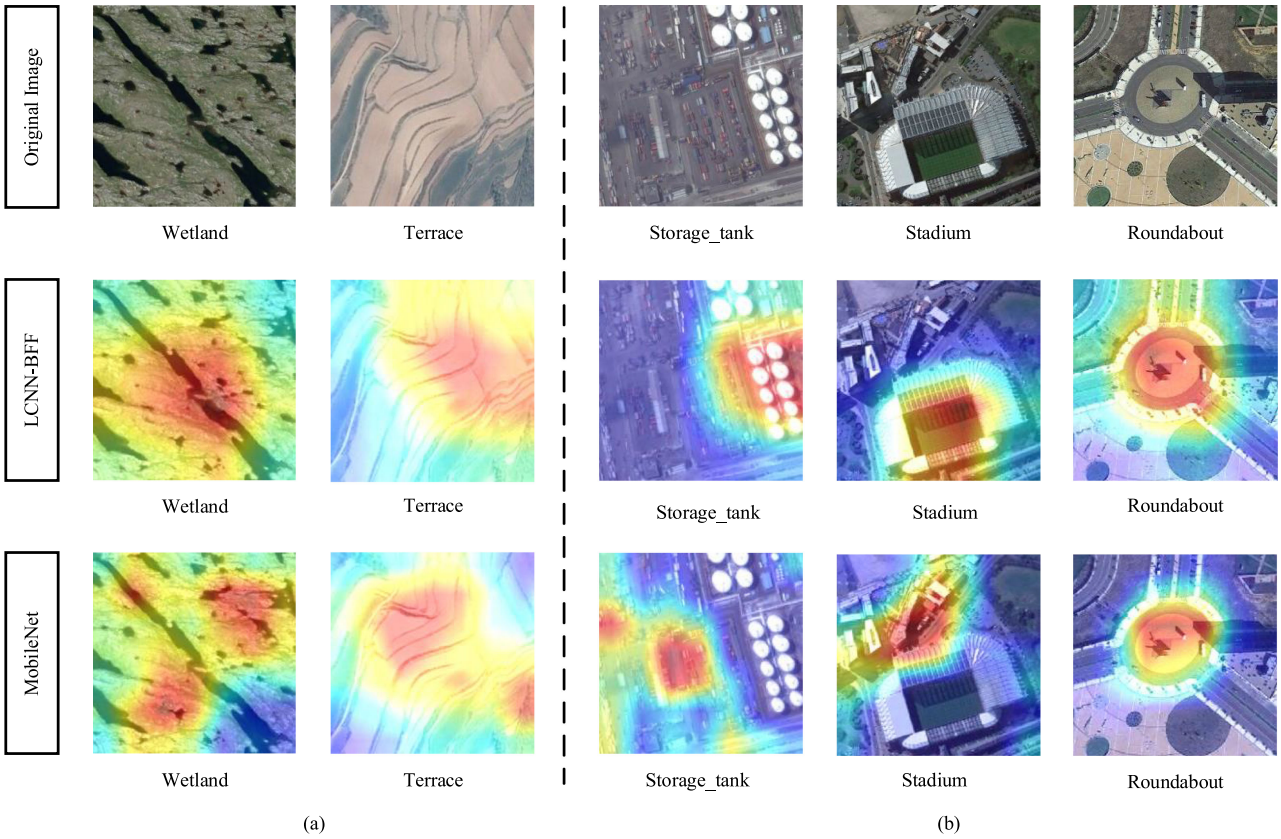


Fig. 14. Visualization results of LCNN-BFF and MobileNet on the NWPU dataset. (a) Scene images classified correctly by both methods. (b) Scene images classified correctly only by LCNN-BFF.

For the proposed method, the extracted feature information was more abundant and concentrated. The feature of interest of the proposed method also included abundant edge information that can assist in correct classification. This is because LCNN-BFF used a bilinear convolution structure to extract image features, which can extract differing feature information through two different branches. The feature information of the two branches was fused and complemented, using the BFF method, and then more abundant and representative feature information was extracted.

The visualization experiments on the two datasets demonstrated that the proposed method could extract the features of the remote sensing scene images more accurately, to help to solve the problems of within-class differences and between-class similarities in the classification of remote sensing scene images.

V. Conclusion

In the research of remote sensing scene classification, we proposed a novel network with a bilinear convolution structure based on a CNN. In addition, the BFF method was proposed to fuse and complement the feature information extracted from the two branches to obtain more abundant and representative feature information. In view of the problem that the bilinear convolution structure improves the complexity of the model, we proposed three kinds of convolution structures, which combined DSC and CConv, to greatly reduce the amounts of parameters and computational complexity of the model. The proposed method was compared with MobileNet and other state-of-the-art methods on four remote sensing scene images using evaluation indices.

The experimental results demonstrated that the proposed method provided a good classification performance on the remote sensing scene images. There remain problems that need to be improved. The extracted feature information can be stacked, fused, and complemented by the BFF; however, the BFF cannot selectively fuse effective feature information. This may produce redundant data and increase the computational complexity. Future work will further reduce the complexity of the model and speed up the convergence of the model. The BFF method can also be improved for the selective fusing of useful feature information, and thus can accommodate the task of remote sensing scene classification more intelligently.

Acknowledgment

The authors would like to thank the handling editor and the anonymous reviewers for their careful reading and helpful remarks, which are all very valuable for improving the quality of this article. The python code for this article can be downloaded from https://github.com/scp19801980/Remote-sensing-scene-classification.

REFERENCES

[1] 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.

[2] 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.

[3] 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.

[4] G. Cheng and J. Han, "A survey on object detection in optical remote sensing images," *ISPRS J. Photogramm. Remote Sens.*, vol. 117, pp. 11–28, 2016.

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

[6] H. Ghassemian, "A review of remote sensing image fusion methods," *Inf. Fusion*, vol. 32, no. 32, pp. 75–89, 2016.

[7] T. Dan *et al.*, "Multifeature energy optimization framework and parameter adjustment-based nonrigid point set registration," *J. Appl. Remote Sens.*, vol. 12, no. 3, 2018, Art. no. 035006.

[8] L. Li *et al.*, "Image registration using two-layer cascade reciprocal pipeline and context-aware dissimilarity measure," *Neurocomputing*, vol. 371, pp. 1–14, 2020.

[9] 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.

[10] X. X. Zhu *et al.*, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 8–36, Dec. 2017.

[11] Y. Feng *et al.*, "Learning deep event models for crowd anomaly detection," *Neurocomputing*, vol. 219, no. 219, pp. 548–556, 2017.

[12] X. Lu, B. Wang, X. Zheng, and X. Li, "Exploring models and data for remote sensing image caption generation," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2183–2195, Apr. 2018.

[13] W. Zhang, X. Lu, and X. Li, "A coarse-to-fine semi-supervised change detection for multispectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3587–3599, Jun. 2018.

[14] 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.

[15] K. Li, G. Cheng, S. Bu, and X. You, "Rotation-insensitive and context-augmented object detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2337–2348, Apr. 2018.

[16] Y. Long, Y. Gong, Z. Xiao, and Q. Liu, "Accurate object localization in remote sensing images based on convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2486–2498, May 2017.

[17] X. Lv *et al.*, "Very high resolution remote sensing image classification with SEEDS-CNN and scale effect analysis for superpixel CNN classification," *Int. J. Remote Sens.*, vol. 40, no. 2, pp. 506–531, 2019.

[18] 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.

[19] L. Fang, N. He, S. Li, A. J. Plaza, and J. Plaza, "A new spatial–spectral feature extraction method for hyperspectral images using local covariance matrix representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3534–3546, Jun. 2018.

[20] 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.

[21] F. Zhao *et al.*, "A novel two-stage scene classification model based on feature variable significance in high-resolution remote sensing," *Geocarto Int.*, to be published, doi: 10.1080/10106049.2019.1583772.

[22] K. Nogueira *et al.*, "Towards better exploiting convolutional neural networks for remote sensing scene classification," *Pattern Recognit.*, vol. 61, pp. 539–556, 2017.

[23] B. Zhang, Y. Zhang, and S. Wang, "A lightweight and discriminative model for remote sensing scene classification with multidilation pooling module," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 8, pp. 2636–2653, Aug. 2019.

[24] G. Wang, B. Fan, S. Xiang, and C. Pan, "Aggregating rich hierarchical features for scene classification in remote sensing imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 9, pp. 4104–4115, Sep. 2017.

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

[26] A. Krizhevsky *et al.*, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[27] A. G. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," Apr. 2017, *arXiv:1704.04861*. [Online]. Available: https://arxiv.org/abs/1704.04861.

[28] M. James Swain and D. H. Ballard, "Color indexing," *Int. J. Comput. Vis.*, vol. 7, no. 1, pp. 11–32, 1991.

[29] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[30] A. Oliva and T. Antonio, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, 2001.

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

[32] 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.

[33] B. A. Olshausen and D. J. Field, "Sparse coding with an overcomplete basis set: A strategy employed by V1?" *Vision Res*, vol. 37, no. 23, pp. 3311–3325, 1997.

[34] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[35] Q. Zhu, Y. Zhong, B. Zhao, G. 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.

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

[37] G. Cheng *et al.*, "Auto-encoder-based shared mid-level visual dictionary learning for scene classification using very high resolution remote sensing images," *IET Comput. Vis.*, vol. 9, no. 5, pp. 639–647, 2015.

[38] 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.

[39] X. Lu *et al.*, "Bidirectional adaptive feature fusion for remote sensing scene classification," *Neurocomputing*, vol. 328, pp. 135–146, 2019.

[40] H. Zhao *et al.*, "Convolutional neural network based heterogeneous transfer learning for remote-sensing scene classification," *Int. J. Remote Sens.*, vol. 40, no. 22, pp. 8506–8527, 2019.

[41] Y. Liu *et al.*, "Scene classification based on multiscale convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 12, pp. 7109–7121, 2018.

[42] W. Zhang *et al.*, "Remote sensing image scene classification using CNN-CapsNet," *Remote Sens.*, vol. 11, no. 5, 2019, Art. no. 494.

[43] Y. Zhou *et al.*, "Remote sensing scene classification based on rotation-invariant feature learning and joint decision making," *EURASIP J. Image Video Process.*, vol. 2019, no. 1, pp. 1–11, 2019.

[44] C. Wang *et al.*, "Multiple resolution block feature for remote-sensing scene classification," *Int. J. Remote Sens.*, vol. 40, no. 18, pp. 6884–6904, 2019.

[45] Y. Boualleg, M. Farah, and I. R. Farah, "Remote sensing scene classification using convolutional features and deep forest classifier," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 12, pp. 1944–1948, Dec. 2019.

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

[47] X. Liu, Y. Zhou, J. Zhao, R. Yao, B. Liu, and Y. Zheng, "Siamese convolutional neural networks for remote sensing scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 8, pp. 1200–1204, Aug. 2019.

[48] B. Liu *et al.*, "Weighted spatial pyramid matching collaborative representation for remote-sensing-image scene classification," *Remote Sens.*, vol. 11, no. 5, 2019, Art. no. 518.

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

[50] X. Lu, H. Sun, and X. Zheng, "A feature aggregation convolutional neural network for remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7894–7906, Oct. 2019.

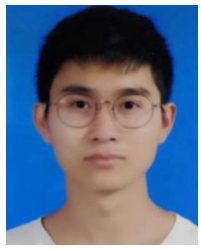
[51] B. Li *et al.*, "Aggregated deep fisher feature for VHR remote sensing scene classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 9, pp. 3508–3523, Sep. 2019.

[52] 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.
[53] 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.
[54] N. He, L. Fang, S. Li, J. Plaza, and A. Plaza, "Skip-connected covariance network for remote sensing scene classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 5, pp. 1461–1474, May 2020.
[55] Z. Lu *et al.*, "The expressive power of neural networks: A view from the width," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6232–6240.
[56] W. Shang *et al.*, "Understanding and improving convolutional neural networks via concatenated rectified linear units," in *Proc. 33rd Int. Conf. Int. Conf. Machine Learn.*, 2016, vol. 48, pp. 2217–2225.
[57] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Machine Learn.*, 2015, pp. 448–456.
[58] M. Lin *et al.*, "Network in network," in *Proc. Int. Conf. Learn. Representations*, 2014, pp. 1–10.
[59] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. Int. Conf. Adv. Geogr. Inf. Syst.*, 2010, pp. 270–279.
[60] Q. Zou, L. Ni, T. Zhang, and Q. Wang, "Deep learning based feature selection for remote sensing scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 11, pp. 2321–2325, Nov. 2015.
[61] G. 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.
[62] H. Sun, S. Li, X. Zheng, and X. Lu, "Remote sensing scene classification by gated bidirectional network," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 1, pp. 82–96, Jan. 2020.
[63] Y. Liu, Y. Liu, and L. Ding, "Scene classification based on two-stage deep feature fusion," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 2, pp. 183–186, Feb. 2018.
[64] R. R. Selvaraju *et al.*, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," *Int. J. Comput. Vis.*, vol. 128, no. 2, pp. 336–359, 2020.
[65] X. Yao, J. Han, G. Cheng, X. Qian, and L. Guo, "Semantic annotation of high-resolution satellite images via weakly supervised learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 6, pp. 3660–3671, Jun. 2016.
[66] 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.
[67] X. Bangquan and W. Xiao Xiong, "Real-time embedded traffic sign recognition using efficient convolutional neural network," *IEEE Access*, vol. 7, pp. 53330–53346, 2019.
[68] H. Pan, Z. Pang, Y. Wang, Y. Wang, and L. Chen, "A new image recognition and classification method combining transfer learning algorithm and MobileNet model for welding defects," *IEEE Access*, vol. 8, pp. 119951–119960, 2020.
[69] J. Han, X. Yao, G. Cheng, X. Feng, and D. Xu, "P-CNN: Part-based convolutional neural networks for fine-grained visual categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published, doi: 10.1109/TPAMI.2019.2933510.
[70] X. Lu, T. Gong, and X. Zheng, "Multisource compensation network for remote sensing cross-domain scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2504–2515, Apr. 2020.
[71] P. Yan, F. He, Y. Yang, and F. Hu, "Semi-supervised representation learning for remote sensing image classification based on generative adversarial networks," *IEEE Access*, vol. 8, pp. 54135–54144, Mar. 2020.
[72] R. Cao, L. Fang, T. Lu, and N. He, "Self-attention-based deep feature fusion for remote sensing scene classification," *IEEE Geosci. Remote Sens. Lett.*, to be published, doi: 10.1109/LGRS.2020.2968550.
[73] D. Zhang, N. Li, and Q. Ye, "Positional context aggregation network for remote sensing scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 6, pp. 943–947, Jun. 2020.
[74] Z. Chen, Y. Wang, W. Han, R. Feng, and J. Chen, "An improved pretraining strategy-based scene classification with deep learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 5, pp. 844–848, May 2020.

Cuiping Shi (Member, IEEE) received the M.S. degree in signal and information processing from Yangzhou University, Yangzhou, China, in 2007, and the Ph.D. degree in information and communication engineering from the Harbin Institute of Technology (HIT), Harbin, China, in 2016.

From 2017 to 2019, she held a postdoctoral research position with the College of Information and Communications Engineering, Harbin Engineering University, Harbin, China. She is currently an Associate Professor with the Department of Communication Engineering, Qiqihar University, Qiqihar, China. Her main research interests include image processing, pattern recognition, and machine learning. She has authored two academic books in optical remote sensing image processing and authored or coauthored more than 40 papers in journals and conference proceedings.

Dr. Shi's doctoral dissertation won the nomination award of Excellent Doctoral Dissertation of HIT, in 2016.



Tao Wang is currently working toward the bachelor's degree with Qiqihar University, Qiqihar, China.

His research interests include remote sensing image processing and machine learning.

Mr. Wang was the recipient of two provincial students awards for his research project.

Liguo Wang (Member, IEEE) received the M.S. and Ph.D. degrees in signal and information processing from the Harbin Institute of Technology, Harbin, China, in 2002 and 2005, respectively.

From 2006 to 2008, he held a postdoctoral research position with the College of Information and Communications Engineering, Harbin Engineering University, Harbin, China, where he is currently a Professor. He has authored two books about hyperspectral image processing and authored or coauthored more than 130 papers in journals and conference proceedings. His main research interests include remote sensing image processing and machine learning.