Scene Classification of Remote Sensing Images Based on Saliency Dual Attention Residual Network

DONGEN GUO\(^1,2\), YING XIA\(^1\), AND XIAOBO LUO\(^1\)

\(^1\)Chongqing Engineering Research Center for Spatial Big Data Intelligent Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China
\(^2\)School of Software, Nanyang Institute of Technology, Nanyang 473000, China

Corresponding author: Ying Xia (xiaying@cqupt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 41871226 and Grant 41571401, and in part by the Special Foundation of Postdoctoral Scientific Research Project of Chongqing under Grant Xm2016081.

ABSTRACT Scene classification of high-resolution Remote Sensing Images (RSI) is one of basic challenges in RSI interpretation. Existing scene classification methods based on deep learning have achieved impressive performances. However, since RSI commonly contain various types of ground objects and complex backgrounds, most of methods cannot focus on saliency features of scene, which limits the classification performances. To address this issue, we propose a novel Saliency Dual Attention Residual Network (SDAResNet) to extract both cross-channel and spatial saliency information for scene classification of RSI. More specifically, the proposed SDAResNet consists of spatial attention and channel attention, in which spatial attention is embedded in low-level feature to emphasize saliency location information and suppress background information, and channel attention is integrated to high-level features to extract saliency meaningful information. Additionally, several image classification tricks are used to further improve classification accuracy. Finally, Extensive experiments on two challenging benchmark RSI datasets are presented to demonstrate that our methods outperform most of state-of-the-art approaches significantly.

INDEX TERMS Remote sensing images, scene classification, spatial attention, channel attention, residual attention network.

I. INTRODUCTION

With the rapid development of remote sensing technology and satellite sensors, a great number of high-resolution RSI have become readily available [1]–[3]. It is highly desirable to automatically interpret high-resolution RSI. Scene classification of RSI, i.e. automatically extracting valuable information from each scene image and categorizing them into different classes based on their semantic information, has become a research hotspot in RSI interpretation [1], [4], [5]. Scene classification of RSI has a wide range of applications, including urban planning, natural disaster detection, land-cover/land-use classification, environment monitoring and so on [6], [7].

High-resolution RSI is quite different from natural images due to their unique imaging perspective and capture mode, which results in images with various types of ground objects and complex backgrounds. Effective feature representation plays a crucial role in scene classification of RSI. Over the past decades, considerable efforts have been made to solve this problem and numerous approaches have been proposed. Existing scene classification methods are usually divided into two categories according to the used features: (a) handcrafted-based feature methods; and (b) learned-based methods, especially deep learning-based methods [5]. In recent years, with the fast development of Convolutional Neural Network (CNN), a variety of CNN-based methods have been dominating the field of scene classification mainly due to its capacity to learn hierarchical representation to describe the image scenes [5], [8], [9]. Most CNN-based methods usually tend to generate a global representation of image with the same contribution of each part and leverage the output of Fully Connected (FC) layer as the global representation to classify scene images. However, RSI commonly contain various types of ground objects and complex backgrounds, but not all objects are useful for scene classification.
classification. Therefore, it is very crucial to recognize the critical objects and regions in scenes, focus on them and abandon the useless ones when representing scene images [5]; in other words, more attention should be paid to saliency object information [10]. Therefore, improving the feature representation by capturing saliency scene information is mainly discussed in this paper.

To solve the aforementioned problems, the Visual Attention Mechanism (VAM) is proposed, which not only presents where to focus but also improves the representation of interests [11]. Recently, there have been several attempts to integrate attention mechanism into CNNs to improve the performance in many tasks, such as image classification [12]–[14], saliency object detection [15], semantic segmentation [16] and so on. Wang et al. [14] propose Residual Attention Network (RAN) which generates attention-aware features by stacking attention module embedded into deep residual network to improve classification performance. Hu et al. [13] introduce “Squeeze-and-Excitation” block by using global average-pooled features for modelling interdependencies between channels explicitly to improve performance. Furthermore, Woo et al. [12] propose an attention module to sequentially infer an attention map along channel and spatial dimensions for adaptive feature refinement. Zhao and Wu [15] propose a saliency object detection method by using channel-wise attention and spatial attention to obtain saliency object location. The spatial attention on low-level feature maps is used to obtain saliency location information, while channel-wise attention on high-level feature maps is used to capture saliency regions. Fu et al. [16] propose a Dual Attention Network (DANet) to adaptively integrate position attention module and a channel attention module for semantic segmentation.

All the above mentioned works are proposed for natural image processing, there have been also some proposals on scene classification of RSI involving VAM [4], [5] currently. Li et al. [4] propose a method called Deep Discriminative Representation Learning with Attention Map (DDRL-AM), which resorts to Gradient-weighted Class Activation Mapping (Grad-CAM) method to produce attention map to extract discriminative features for scene classification of RSI. Guo et al. [5] propose a global-local attention network (GLANet) to capture both global and local information for aerial scene classification. The above mentioned works mainly apply the attention mechanism to the high-level features to enhance feature representation, but they have no consideration to filter the background of the low-level features, so the representation of saliency areas in high-level features is insufficient. Therefore, mainly inspired by some proposals in [12], [15], we propose a saliency dual attention residual network named SDAResNet to extract saliency object features for scene classification. More specifically, we introduce spatial attention and channel attention into ResNet101 to construct residual attention network, in which spatial attention is integrated into low-level features of shallow layer in ResNet101 to highlight saliency region and suppress background information, while channel attention is embedded into high-level features of deep layer in ResNet101 to enhance the scene saliency for extracting saliency semantic information.

In addition, in the past few years, some tricks for improving image classification accuracy have been proposed [17]–[24]. Especially, He et al. [24] summarize some tricks for natural image classification with CNN in detail. Extensive experimental results show that several tricks and their combinations can improve significantly the classification performance of natural images. However, due to the huge differences between RSI and natural images in terms of resolution, imaging perspective, object size and objects quantity, the availability of these tricks on scene classification of RSI needs to be investigated. In this paper, we have carefully selected several simple tricks for practical verification, such as Xavier initialization [17], mixup training [18], learning rate cosine decay [20], learning rate warm up [21], no bias decay [22], etc. Empirical evaluations are adopted to verify which tricks are effective for scene classification of RSI.

We summarize the main contributions of our work as follows:

1. We propose a Saliency Dual Attention Residual Network (SDAResNet) to extract saliency scene information and improve the feature representation of RSI, it can simultaneously utilize cross-channel information and spatial information to capture discriminative saliency scene information to boost the performance of scene classification.

2. A spatial attention block embedded into low-level feature is proposed to suppress background and highlight saliency location information, while a channel attention block integrated in high-level features is used to extract saliency semantic information. The two attention blocks can focus on the most critical part of a scene image to improve saliency feature representation of RSI.

3. Several image classification tricks and effective combinations are introduced into the proposed SDAResNet to improve training process for better scene classification performance.

4. Extensive experimental results show that our proposed SDAResNet outperform most of recent state-of-the-art methods on two benchmark RSI datasets, while several image classification tricks and effective combinations can further improve classification accuracy.

The remainder of this paper is organized as follows. Section II gives a brief review of related works. Section III describes the proposed method in detail. Section IV introduces experimental designs and results. Experimental results are discussed in Section V. Finally, Section VI draws the conclusion and states future work.

II. RELATED WORK

In this section, we review the related work including visual attention mechanism, residual attention network and image classification tricks.
A. VISUAL ATTENTION MECHANISM

It is well known that visual attention mechanism (VAM) stems from the studies of human vision, it plays an important role in human perception [11]. In the field of cognitive science, one does not attempt to process a whole scene at once, but selectively focuses on saliency parts because of bottlenecks in information processing [25], which is called VAM.

Actually, the attention mechanism is involved in learning the weight distribution of different parts, corresponding to different degrees of concentration [12]. The benefits of this property have been proven in many tasks, such as machine translation [25], object recognition [26], [27], semantic segmentation [16], image captioning [28] and image classification [13], [29], [30].

Recently, there have been several attempts to integrate attention processing into CNNs to improve the performance in large-scale image classification tasks [12]–[14], [29]. Wang et al. [14] propose Residual Attention Network which generates attention-aware features from different modules by stacking attention module, which can improve classification performance. Hu et al. [13] introduce “Squeeze-and-Excitation” block, which can bring significant improvements in performance for existing state-of-the-art CNN by using global average-pooled features to explicitly modelling interdependencies between channels. Furthermore, in [12], [29], researchers propose a simple and effective attention module, which can sequentially infer an attention map along channel and spatial two separate dimensions, then attention maps are multiplied to the input feature map for adaptive feature refinement.

More recently, some research works have attempted to focus the feature extraction process on saliency image regions. Zhao and Wu [15] propose a novel saliency object detection method named Pyramid Feature Attention network. A Context-aware Pyramid Feature Extraction (CPFE) module is designed for high-level feature maps, furthermore, channel-wise attention on CPFE feature maps is adopted to capture saliency regions and spatial attention on low-level feature maps is used to obtain saliency location information. Fu et al. [16] propose a Dual Attention Network (DANet) to adaptively integrate local features with their global dependencies for scene segmentation. A position attention module and a channel attention module are introduced to capture global semantic interdependencies.

All the previous works are proposed for natural image processing, and such works have shown excellent performances in image classification, detection and so on. Due to the strong feature representation ability of VAM, it is also suitable for the field of scene classification of RSI. Currently, there have been also some proposals on scene classification of RSI. Some attempts have been made to combine attention mechanism into deep ResNet to improve the classification performance [12], [14], [29]. Wang et al. [14] design residual attention network (RAN) by stacking multiple attention modules to generate attention perception features. Attention module which is constructed by using pre-activated residual units, ResNeXt and Inception as the basic units of RAN is divided into two branches: mask branch and trunk branch. In [12] and [29], researchers propose a simple and effective attention module, which decompose the calculation process of three-dimensional attention map into two separate pathways along the channel and spatial axis to sequentially infer attention map, and then integrate the attention map into residual block to perform adaptive feature refinement. Nevertheless, channel attention and spatial attention are sequentially integrated into each residual block, which not only increases the number of parameters and computation, but also takes no account of the characteristics of different convolution layers and different attentions. Unlike the above mentioned methods, instead of sequentially incorporating channel attention and spatial attention into each BasicBlock (BottleneckBlock) of the ResNet, we integrate the spatial attention into conv2\_x layer of ResNet to capture saliency location information, and channel attention into conv3\_x, conv4\_x and conv5\_x layer of ResNet to obtain saliency semantic
C. IMAGE CLASSIFICATION TRICKS FOR IMPROVING TRAINING PROCESS

With the introduction of AlexNet [34], deep neural network, especially CNN, has become the dominant method of image classification. In just a few years, various new network architectures have emerged, including VGG [35], Inception [36], ResNet [33], DenseNet [37] and NASNet [38]. These new architectures steadily improve classification accuracy of large-scale natural images. However, these improvements do not merely come from the improved model architectures, the improved training process also play an important role [24]. Many tricks have been applied to tasks such as natural image classification and detection. He et al. [24] summarize in detail the tricks for natural image classification with CNN. Empirical evaluation show that several tricks and their combinations lead to significant accuracy improvement. However, due to the huge differences between RSI and natural images in terms of resolution, imaging perspective, object size and objects quantity, tricks applicable to natural image classification may not be well applied to scene classification of RSI. In this paper, we mainly investigate a few simple tricks for improving training process to verify their effectiveness on scene classification of RSI. The following is a brief review of several tricks.

It is well known that data augmentations are the effective strategy to increase the model generalization. Zhang et al. [19] propose a data-agnostic and straightforward data augmentation approach: mixup training, which trains on virtual examples constructed as the linear interpolation of two random examples from the training set and their labels. Mixup can regularize the neural network to increase the robustness and stabilize the training process. When mixed training is used, more training epochs are generally required to obtain better training results. Zhong et al. [39] propose a new data augmentation method named “Random Erasing” (RE), which randomly selects and erases a rectangle region in input image to obtain the training images with occlusion to reduce overfitting and improve the robustness of model.

In Stochastic Gradient Descent (SGD) algorithm, the learning rate determines the speed at which the parameter moves to the optimal value, even the performance of the algorithm. If learning rate is too large, it is likely to miss the optimal value; if learning rate is too small, the efficiency of optimization may be too low, and convergence time is extremely long. Therefore, the learning rate scheduling is proposed. Generally, at the beginning of the training, a higher learning rate is used to achieve fast convergence; as training progresses, the learning rate is gradually reduced, which is helpful to find the optimal solution. In literatures [20], [33], the researchers propose Multi-step (MultiStepLr) and cosine (CosLr) learning rate decay strategy respectively, the former adjusts the learning rate by setting step intervals, and the latter adjusts learning rate on the basis of cosine curve. In addition, the parameters of model are typically initialized randomly at the start of training, directly using a large initial learning rate will affect the training effect. Priya et al. [21] propose a gradual warmup (Warmup) trick that gradually ramps up learning rate from 0 to the initial learning rate linearly to avoid a sudden increase of learning rate and achieve gradual convergence at the beginning of training. After Warmup, it goes back to initial learning rate schedule. In this work, we use the first m epochs (m = 5) to warm up, and the initial learning rate \( \eta \) (\( \eta = 0.1 \)) then at batch i (\( 1 \leq i \leq m \)), the learning rate is \( i^*\eta/m \).

The weight initialization method has a crucial influence on the convergence speed and performance of the model. Glorot and Bengio [17] propose Xavier initialization (XavierInit) to solve the problem of random initialization. The main idea is to get the input and output to follow the same distribution as possible, so that the output value of the activation function at the next layer may not approach 0. Jia et al. [22] propose No Bias decay strategy (NoBias), it is recommended to only apply regularization to the weights in convolution and FC layers, while other parameters including the biases, \( \gamma \) and \( \beta \) in BN layers are left unregularized. NoBias strategy can alleviate overfitting and boost classification performance. Szegedy et al. [23] propose a Label Smoothing (LS) algorithm to regularize the classifier layer by estimating the marginalized effect of label-dropout during training. This algorithm obtains the cross entropy by calculating weighted average and average distribution of hard target in the dataset, which effectively improves the accuracy. It is actually a constraint method to minimize the overfitting degree of the model by adding noise to the output label \( y \).

III. THE PROPOSED METHOD

For extracting saliency features to boost scene classification performance, a novel SDAResNet is proposed for scene classification of RSI. The framework of SDAResNet is...
The framework of our proposed SDAResNet and original ResNet101. Fig. 2 (a) shows the original ResNet101 framework. It consists of conv1, conv2 x (3 BottleneckBlocks), conv3 x (4 BottleneckBlocks), conv4 x (25 BottleneckBlocks), conv5 x (3 BottleneckBlocks) and a FC layer and a sigmoid layer. Fig. 2 (b) represents the framework of proposed SDAResNet. Similar to the original ResNet101, the red dotted box corresponds to conv2 x, and the three blue dotted boxes correspond to conv3 x, conv4 x and conv5 x. In red dotted box, \( F_s \) represents a low-level feature map extracted from conv2 x layer, firstly, spatial refined feature \( F_s \) is generated by SA, then attention residual learning is performed to extract feature maps with SA; in blue dotted box, \( F_r \) represents high-level feature map extracted from conv3 x, conv4 x or conv5 x layer, firstly, channel refined feature \( F_c \) is obtained by CA, and then attention residual learning is performed to extract the feature map with CA. Finally, the feature vector is extracted from FC layer. Where \( \odot \) denotes element-wise addition and \( \oplus \) denotes dot multiplication.

- **A. SPATIAL ATTENTION**

In standard CNN structures, each convolutional feature map commonly consists of three dimensions: the channel, width and height. Remote sensing images usually contain a wealth of ground details and complex backgrounds. In general, the saliency map from low-level features contains a lot of background details which easily influences saliency detection [15]. In order to suppress noise disturbance to extract saliency features for improving performance of scene classification, we need focus more on the foreground saliency regions and suppress background. While spatial attention aims to obtain the inter-dependencies of spatial dimension and then learn the most significant region of an RSI. Woo et al. [12] apply maximum pooling and average pooling to generate spatial attention map integrating with all layers. Zhao and Wu [15] proposed the embedment of spatial attention with the low-level features because high-level features contain abstract semantics which has no need to filter spatial information. Inspired by the proposals of [12] and [15], we design spatial attention block as shown in the dotted box of Fig. 3 (a). The process will be described in detail below.

As shown in Fig. 2 and Fig. 3 (a), a spatial attention block integrated with low-level feature of ResNet101 mainly contains two convolution layers and a softmax layer. In order to increase receptive field and get global information without increasing parameters, similar to [15], we apply two convolution layers instead of maximum and average pooling to generate spatial attention weight matrix. Firstly, the low-level feature map \( F_1 \) is fed into two consecutive convolution layers \( f_1 \) and \( f_2 \), and two feature maps are extracted by using different combinations of two convolution layers, where kernel size of \( f_1 \) is \( 1 \times k \) and the other kernel size is \( k \times 1 \). Secondly, the two feature maps are concatenated to generate an efficient feature descriptor. Finally, softmax operation is performed on the concatenated feature descriptor to capture spatial concerns and generate spatial attention weight matrix \( F_{SA} \in \mathbb{R}^{W \times H} \). In brief, the spatial attention weight matrix is derived as...
follows:

\[ C_1 = f_2(f_1(F_i, W_{i1}^1), W_{i1}^2), \]
\[ C_2 = f_1(f_2(F_i, W_{i2}^1), W_{i2}^2), \]
\[ F_{SA} = F(F_i, \mathcal{W}) = \sigma(C_1 + C_2). \]

where \( \mathcal{W} \) refers to the weight parameters in convolution operation, \( \sigma \) refers to softmax operation, \( F_i \in \mathbb{R}^{W \times H \times C} \) represents low-level feature from conv2_x layer of ResNet101, \( f_1 \) and \( f_2 \) refer to \( 1^k \) and \( k^1 \) convolution layer respectively and we set \( k = 9 \) in experiment. \( C_1 \) represents the results of two convolution operations of the top branch in Fig.3(a), i.e., firstly performing \( 1^k \) convolution operation, subsequently performing \( k^1 \) convolution operation on low-level feature map \( F_i \). Similar to \( C_1 \), \( C_2 \) represents the results of two convolution operations of the bottom branch in Fig.3(a). \( F_{SA} \) refers to spatial attention weight matrix which is computed through performing softmax operation on the results of the \( C_1, C_2 \) concatenation operation.

For a detailed understanding the spatial attention block, according to Eq.1, Eq.2 and Eq.3, the following formulas can be derived:

\[
F_{SA_{i,j}} = \frac{\exp(f_2(f_1(F_i, W_{i1}^1), W_{i1}^2)+f_1(f_2(F_i, W_{i2}^1), W_{i2}^2))}{\sum_{i=1}^{N} \exp(f_2(f_1(F_i, W_{i1}^1), W_{i1}^2)+f_1(f_2(F_i, W_{i2}^1), W_{i2}^2))}.
\]

where \( F_{SA} \) is a matrix, which represents the inter-spatial relationship of each two low-level feature maps, \( N \) is the numbers of spatial features. Specifically, \( F_{SA_{i,j}} \) means the \( i^{th} \) position’s impact on \( j^{th} \) position. If the set of spatial points is defined as \( \mathbb{R} = \{(x, y) \mid x = 1, \ldots, W; y = 1, \ldots, H \} \) then \( i = (x, y), j = (x, y) \) is the \( i^{th} \) and \( j^{th} \) position coordinate, respectively. Here, the probability of the corresponding feature can be obtained through the softmax operation. A larger probability value indicates that the feature is more important, and vice versa. So the final result of \( F_{SA} \) is actually a weight matrix. The final output \( F_S \) is obtained by dot multiplication operation between \( F_i \) and \( F_{SA} \). The formula \( F_S \) is derived as follows:

\[
F_S = F_i \odot F_{SA},
\]

where \( \odot \) denotes dot multiplication.

**B. CHANNEL ATTENTION**

The high-level features in CNNs usually contain meaningful semantic information, while different channels of features correspond to different semantics [13]. In standard CNN framework, since each channel of feature map is computed by convolution operation, the redundancy information among different channel is inevitable. To reduce the inter-channel redundancy of feature map and focus on key part, Woo et al. [12] generate channel attention map by exploiting the inter-channel relationship to focus on ‘what’ is meaningful in image, but they didn’t take into account the difference between low-level and high-level features. In order to enlarge the saliency from high-level features, Zhao and Wu [15] add channel-wise attention module to high-level features for saliency detection. Moreover, in the literatures [12], [13], the researchers conduct comparative experiments on the use of average and maximum pooling to construct channel attention, the results show that average pooling is slightly better than maximum pooling, but Woo et al. [12] verify that using average and maximum pooling together to construct channel attention can obtain better performance. According to the characteristics of maximum pooling and average pooling, for the high-level feature maps of deep layer, average pooling can retain the overall data features to obtain global high-level semantic information for promoting classification performance, while maximum pooling can capture the salient part of the feature (i.e., the saliency feature). So such two pooling methods are used together to better improve classification performance. Inspired by above mentioned proposals, we generate a channel attention weight matrix only for high-level features using maximum and average pooling as well as a shared network, which not only increases the saliency of ground objects in RSI but also reduces the computation cost. The channel attention aims to learn the inter-channel relationships of remote sensing image feature maps. The detailed structure of channel attention block is shown in the dotted box of Fig. 3 (b). The process will be described in detail below. As shown in Fig.2 and Fig. 3 (b), a channel attention integrated with high-level features of ResNet101 mainly contains two pooling layers and a shared Multi-Layer Perceptron (MLP) with one hidden layer. Firstly, two different spatial context descriptors of a high-level feature map \( F_k \), i.e., average-pooled features \( F_{avg} \) and max-pooled features \( F_{max} \), are generated by using average pooling and maximum pooling along channel
axis respectively, and spatial information of \( F_h \) is aggregated by using element-wise addition operation. Secondly, \( F_{avg}^c \) and \( F_{max}^c \) are separately forwarded to a shared MLP network to generate two output feature vectors. To reduce parameter overhead, we only use one hidden layer in the shared MLP network and set the hidden activation size to \( \mathbb{R}^{C/r \times 1 \times 1} \), where \( r \) is the reduction ratio and is set to 9 in the experiment. Then, the output feature vectors are merged by using element-wise addition. Finally, softmax function is used to take the normalization measures to generate the channel attention weight matrix \( F_{CA} \in \mathbb{R}^{C \times 1 \times 1} \). In short, CA is derived as follows:

\[
F_{CA}(F_h) = \sigma(MLP(F_{avg}^c) + MLP(F_{max}^c)) = \sigma(W_1(\delta(V_0(F_{avg}^c))) + W_1(\delta(V_0(F_{max}^c)))), \tag{6}
\]

where \( F_{CA} \) denotes the channel attention weight matrix, \( \sigma \) denotes the softmax function which is performed on each row, \( V_0 \in \mathbb{R}^{C/r \times C} \) and \( W_1 \in \mathbb{R}^{C \times C/r} \) denote weights of the MLP, which are shared for both two inputs and ReLU activation function (\( \delta \)) followed by \( V_0 \).

For a detailed understanding of the channel attention,

\[
F_{CA,i,j} = \frac{\exp(W_1(\delta(V_0(F_{avg}^c)))) + W_1(\delta(V_0(F_{max}^c))))}{\sum_{i=1}^{N} \exp(W_1(\delta(V_0(F_{avg}^c)))) + W_1(\delta(V_0(F_{max}^c))))}, \tag{7}
\]

where \( F_{CA,i,j} \) means the relationship between the \( i \)-th channel and the \( j \)-th channel, \( N \) is the numbers of spatial features. After softmax operation, the final output with channel attention \( F_C \) is obtained by dot multiplication between \( F_h \) and \( F_{CA} \). The \( F_C \) is derived as follows:

\[
F_C = F_h \odot F_{CA}, \tag{8}
\]

where \( \odot \) denotes dot multiplication.

C. ATTENTION RESIDUAL LEARNING

Because of the different characteristics of SA and CA, SA is integrated with Conv2 \( x \) layer of ResNet101, while CA is integrated with Conv3 \( x \), Conv4 \( x \) and Conv5 \( x \) layer of ResNet101. Similar to ideas in residual learning, attention mechanism is introduced to residual learning. As Batch Normalization (BN) is a widely used technology to stabilize the training process, BN is added to attention residual learning for fast convergence [40]. In addition, according to the pre-activation idea proposed in [41], BN is placed in front of ReLU, which can further stabilize training and improve classification performance. For an intuitive understanding, the attention residual block of our proposed SDAResNet is plotted as shown in Figure 4, two attentions are integrated into residual learning. The structure of original residual block is shown in Fig. 4(a), our proposed attention residual block is shown in Fig. 4(b), in which attention feature is added to residual block to perform attention residual learning, at the same time, BN and ReLU are added sequentially before the attention residual learning. As can be seen from Fig. 4, after the computation of attention feature, the corresponding refined feature can be extracted by matrices multiplication and element-wise summation. The process will be described in detail below.

For a given intermediate feature map \( X_l \), BN and ReLU are first used pre-activation, then attention feature matrix is generated by using CA or SA and weighted feature is get by matrices multiplication, and then the refined attention feature is generated using element-wise summation. The following is the derivation of the corresponding formula. After using pre-activation, residual learning formula is as follows:

\[
X_{l+1} = X_l + \mathcal{F}(\hat{f}(X_l), W_l), \tag{9}
\]

where \( \hat{f} \) represents pre-activation function, \( \mathcal{F}() \) represents residual function, \( W_l \) represents weight matrix. According to (9) and (5), (8), the output of spatial attention residual block is calculated as follows:

\[
X_{l+1} = X_l + \mathcal{F}(\hat{f}(F_S \oplus F_l), W_l) = X_l + \mathcal{F}(\hat{f}(F_{CA} \odot F_h \oplus F_l), W_l). \tag{10}
\]

The output of channel attention residual block is calculated as follows:

\[
X_{l+1} = X_l + \mathcal{F}(\hat{f}(F_{CA} \odot F_h), W_l) = X_l + \mathcal{F}(\hat{f}(F_{CA} \odot F_h), W_l), \tag{11}
\]

IV. EXPERIMENTS

In this section, extensive experiments are conducted to evaluate the effectiveness of the proposed SDAResNet. In addition, ablation studies are performed to verify the effectiveness of some tricks and their combinations, which are effectively used in natural image classification and detection. To thoroughly evaluate the effectiveness of our final model, two challenging remote sensing scene datasets (NWPU-RESISC45 [1] and PatternNet [3]) are first introduced. Then, the details of experimental setting and evaluation metrics are explained. Finally, the experiments results and analysis are presented to evaluate the effectiveness of proposed SDAResNet and some tricks.
A. DATASET DESCRIPTION
In order to validate our work, we conduct experiments on two public benchmark remote sensing scene datasets (NWPU-RESISC45 [1] and PatternNet [3]).

The NWPU-RESISC45 dataset [1], which is created by Northwestern Polytechnical University from Google Earth, includes 31,500 images, covering 45 scene classes with 700 images in each class. Each image is with a size of 256 * 256 pixels in the red-green-blue (RGB) color space, while the spatial resolution of those images varies from about 30 to 0.2 m per pixel. To the best of our knowledge, NWPU-RESISC45 is currently the most challenging large-scale remote sensing scene dataset.

The PatternNet [3] is another large-scale high-resolution RSI dataset, in which the images are collected by Wuhan University from Google Earth or the Google Map API for US cities.2 There are 38 classes and each class contains 800 images of size 256 * 256 pixels. Similar to the NWPU-RESISC45 datasets, PatternNet contains images with varying resolution, the highest spatial resolution is around 0.062m and the lowest spatial resolution is around 4.693m.

The NWPU-RESISC45 dataset is randomly split into 10% (and 20%) for training and the rest 90% (and 80%) for testing. For the PatternNet dataset, the training proportions are set 20% and 50%, the rest for testing. These proportions have been selected in accordance with the previous studies in the literature [5], in order to facilitate our comparisons with state-of-the-art approaches.

B. EXPERIMENTAL SETTING AND EVALUATION METRICS
1) EXPERIMENTAL SETTING
Our proposed method is implemented by using pytorch framework. The parameters in the training stage are set as follows. The mini-batch size in SGD optimizer is fixed as 40 to cater to the GPU memory, the weight decay is set 0.00001 and momentum is set 0.9. The initial learning rate is fixed as 0.1 to accelerate convergence. The total training epochs are set 220. The parameters of some tricks mentioned in section 2.3 are as follows: the warmup is 5, step interval of MultiStepLr is [30, 80, 130, 180], the parameter $\alpha$ of Mixup is 0.4. All experiments are evaluated on an Ubuntu 16.04 sever with Intel(R) Xeon(R) Gold6048CPU @2.36 GHz with 8 cores and four TITAN V GPUs.

2) EVALUATION METRICS
To evaluate the results of the proposed SDAResNet for scene classification and the effectiveness of some tricks, the overall accuracy and the confusion matrix are adopted as evaluation metrics in the following experiments.

Overall Accuracy (OA): It is defined as the number of correctly classified samples divide by the total number of samples without taking into account the category to which they belong. The formula is as follows:

$$OA = \frac{T}{(T + F)},$$

where $T$ represents the number of correctly classified samples, $F$ represents the number of misclassification samples.

Confusion Matrix (CM): To show the performance of the algorithm in a visual way, the CM is an informative table used to analyze the errors and confusions between different classes, and it gets by counting each class of correct and incorrect classification of the test images and accumulating the results in the table. In this matrix, each row represents actual categories, and each column represents predicted value. Therefore, it can be very easy to show that whether these multiple categories have been confused or not.

In addition, to obtain reliable results, all of experimental results are obtained as the average of ten repeated experiments using randomly selected training samples.

C. EXPERIMENTAL RESULTS AND ANALYSIS
1) ABLATION STUDY
In this section, the proposed SDAResNet is used as the baseline to evaluate the effectiveness of some tricks under Training Ratio (TR) of 10% and 20% on NWPU-RESISC45 dataset. In order to analyze the influence of different tricks, experiments with different tricks are conducted and the results are reported in Table 1.

In Table 1, Baseline denotes SDAResNet without any tricks, Mixup means mixup training trick [19], CosLr means cosine decay of learning rate [20], MultiStepLr means multi-step decay of learning rate [33], XavierInit denotes weight initialization [17], Warmup denotes warmup training [21], NoBias denotes no bias weight decay [22], LS denotes label smoothing [23], RE denotes a data augmentation trick of random erasing [39].

By comparing the performances of Baseline and each trick under the training ratio of 10% and 20%, we find that CosLr can improve the accuracy of 2.231% and 2.274%, which may be because the cosine learning rate decay can make the gradient much better approach the optimal value and

---

Table 1. OA (%) of different tricks and effective combinations under training ratio of 10% and 20% on NWPU-RESISC45 dataset. The bold and underlined indicate the best and the second performance.

| Trick          | Overall Accuracy |
|----------------|------------------|
| Baseline       | OA (TR=10%) OA (TR=20%) |
| Mixup          | 89.48 ± 0.09     | 91.15 ± 0.13 |
| CosLr          | 91.63 ± 0.11     | 93.42 ± 0.09 |
| MultiStepLr    | 88.37 ± 0.07     | 90.10 ± 0.11 |
| XavierInit     | 88.37 ± 0.11     | 90.10 ± 0.10 |
| Warmup         | 90.00 ± 0.17     | 91.76 ± 0.08 |
| NoBias         | 90.02 ± 0.13     | 91.79 ± 0.11 |
| LS             | 90.27 ± 0.21     | 92.03 ± 0.07 |
| RE             | 89.73 ± 0.19     | 91.48 ± 0.15 |
| [1]            | 92.88 ± 0.13     | 94.36 ± 0.15 |
| [1]+[2]+[6]+[7]+[8]+[9] | 92.98 ± 0.10     | 94.47 ± 0.11 |
| [1]+[2]+[3]+[6]+[7]+[8]+[9] | 93.15 ± 0.09     | 94.86 ± 0.15 |
thus improve the classification performance; LS can improve the accuracy of 0.867% and 0.884%, it is probably because that the label smoothing can effectively prevent the model from overfitting and improve the generalization ability of the model to obtain better classification performance; Regarding NoBias, it may leave some parameters such as bias no regularization during training process to effectively prevent the model from overfitting and obtain better classification performance, which may justify the accuracy improvement of 0.624% and 0.636%; Warmup can improve the accuracy of 0.603% and 0.614%, which might be because the warmup trick can gradually ramps up the learning rate at the training beginning achieving stable convergence and better classification performance; RE can improve the accuracy of 0.328% and 0.334%, which is probably because the RE trick by increasing the training image with occlusion can prevent the model from overfitting and obtain better classification performance; Mixup can only improve the accuracy of 0.063% and 0.065%, the possible reason is that mixup trick generally requires more training epochs to achieve better classification performance. Furthermore, the accuracies of MultiStepLr and XavierInit are reduced by about 1%. XavierInit trick degrades the accuracy because it is used with sigmoid activations (not symmetric around 0), which will yield poor learning dynamics and initial saturation of the top hidden layer. MultiStepLr trick decreases, which may be caused by the unreasonable step interval and fewer training epochs.

To indicate the effectiveness of trick combinations, we verify the combinations of several effective tricks under the training ratio of 10% and 20% on NWPU-RESISC45 dataset. We can observe that the combination of Baseline, CosLr, Warmup, LS and RE improve the accuracy by 3.48% and 3.21% respectively, the combination of Baseline, Mixup, CosLr, Warmup, LS and RE improve the accuracy by 3.58% and 3.32%, while the combination of Baseline, Mixup, CosLr, Warmup, NoBias, LS and RE improve the accuracy by 3.75% and 3.71%, the best accuracy reaches 93.15% and 94.86% respectively. From the above experimental results, it can be seen that the superimposition of several effective tricks can greatly improve the accuracy, which may be because such effective combinations can make full use of the advantages of each trick to achieve better classification performance. Based on this we can see the effectiveness of some tricks and effective combinations for scene classification of RSI.

2) COMPARATIVE RESULTS ON NWPU-RESISC45 DATASET

The Comparative results on NWPU-RESISC45 dataset are showed in Table 2, which includes our proposed SDAResNet and several state-of-the-art representative approaches. All these methods all use CNN-based features. Among them, Alex, GoogleLeNet and VGGNet are CNN-based single feature methods; LASC-CNN (multiscale) [42], TEX-TSNet [43], SAL-TS-Net [43], SCCov [45] and DCNNs [44] are CNN-based multiple feature methods; ResNet+AM [4], ResNet+AM+CL [4] and GLANet [5] are attention-based feature methods. As can be seen in Table 2, CNN-based multiple feature methods outperform CNN-based single feature methods in very big margins under the training ratios of 10% and 20%, which demonstrates that the huge superiority of CNN-based multiple features for scene classification of RSI. In addition, attention-based methods are superior the other CNN-based methods, ResNet+AM+CL [4] and GLANet [5] obtain better accuracy compared with CNN-based multiple feature methods, as attention mechanism is utilized. Last but most important, three combinations of several effective tricks outperform all other methods of comparison, which achieves the overall accuracy 92.88%, 92.98%, 93.15% and 94.36%, 94.43%, 94.86% under the training ratio of 10% and 20% respectively.

Figure 5-6 show the confusion matrix generated by the best combination of proposed SDAResNet and several effective tricks under training ratios of 10% and 20% on NWPU-RESISC45 dataset respectively. As shown in Fig. 6, 40 categories among all 45 achieve classification accuracies greater than 90%. It is well known that there exist similar spatial distribution and identical objects among “medium residential”, “dense residential” and “sparse residential”. Fig. 6 shows that our method can accurately classify these large inter-class similarity scenes (accuracies are 92% and 95% for “dense residential” and “sparse residential”). Furthermore, for “railway”, “freeway”, “runway” and “railway station”, although there are large intra-class diversities, the accuracy can still be achieved 93%, 94%, 93% and 89% respectively. All these indicate that our approach is reasonable. Nevertheless, both “church” and “palace” are still difficult to recognize (accuracy is 82% and 80% respectively). As can be seen from the comparison between figure 5 and figure 6, with the decrease of training samples, the accuracy of almost all classes decreases correspondingly. On the whole, our proposed SDAResNet and effective tricks can achieve better accuracy than the current state-of-the-art methods.

| Method                  | Overall Accuracy |
|-------------------------|-----------------|
|                          | OA(TR=10%)      | OA(TR=20%)     |
| Alex [1]                | 96.69±0.21      | 97.85±0.13     |
| GoogleLeNet [1]         | 97.19±0.38      | 98.48±0.26     |
| VGGNet [1]              | 97.47±0.18      | 97.79±0.15     |
| LASC-CNN(multiscale) [42]| 81.37           | 84.30          |
| TLX-TS-Net [43]         | 84.77±0.24      | 86.36±0.19     |
| SAL-TS-Net [43]         | 85.02±0.25      | 87.01±0.19     |
| DCNN [44]               | 89.22±0.50      | 91.89±0.22     |
| SCCov [45]              | 89.30±0.35      | 92.10±0.25     |
| ResNet+AM [4]           | 91.84±0.05      | 91.03±0.13     |
| ResNet+AM+CL [4]        | 92.17±0.08      | 92.46±0.09     |
| GLANet [5]              | 91.03±0.18      | 93.45±0.17     |
| Baseline+CosLr+Warmup+LS+RE(ours) | 92.88±0.13 | 94.36±0.12    |
| Baseline+Mixup+CosLr+Warmup+LS+RE(ours) | 92.98±0.10 | 94.47±0.11    |
| Baseline+Mixup+CosLr+Warmup+NoBias+LS+RE(ours) | 93.15±0.09 | 94.86±0.15    |
FIGURE 5. Confusion matrix generated by the best combination of proposed SDAResNet and effective tricks under training ratio of 10% on NWPU-RESISC45 dataset.

FIGURE 6. Confusion matrix generated by the best combination of proposed SDAResNet and effective tricks under training ratio of 20% on NWPU-RESISC45 dataset.

3) COMPARATIVE RESULTS ON PATTERNNET DATASET

Extensive experiments on the PatternNet dataset are conducted by the combination of proposed SDAResNet and several tricks. As can be seen in Table 3, our results outperform other comparison approaches [5], [46], [47] with overall accuracy of 99.30% and 99.58% under training ratios of 20% and 50%, respectively. As shown in Table 3, LANet [5] and GLANet(SVM) [5] (98.64% and 98.91% under training ratio of 20%) are superior to all CNN-based single feature methods, which indicates that using attention mechanism is a good strategy to further improve classification accuracy, but still inferior to our SDAResNet. Furthermore, our proposed
FIGURE 7. Confusion matrix generated by the best combination of proposed SDAResNet and effective tricks under training ratio of 20% on PatternNet dataset.

TABLE 3. OA (%) of the combinations of proposed SDAResNet and several effective tricks and other Comparative algorithms on the NWPU-RESISC45 dataset. The bold and underlined indicate the best and the second performance.

| Method                          | Overall Accuracy |
|--------------------------------|------------------|
|                                | OA(TR=20%)       | OA(TR=50%)       |
| Baseline                        | 94.19            | -                |
| GoogleLeNet64*64*6 [46]         | 94.81            | -                |
| ResNet-101 singlecrop [46]      | 95.37            | -                |
| PW-ResNet-101 singlecrop [46]   | 81.37            | 84.30            |
| GoogleLeNet(SVM) [47]           | 97.72±0.02       | -                |
| VGGNet(SVM) [47]                | 97.52±0.02       | -                |
| Inception-V3(SVM) [47]          | 97±0.02          | -                |
| ResNet101(SVM) [47]             | 98.6±0.02        | -                |
| LANE [5]                        | 98.64±0.23       | 98.82±0.24       |
| GLANet(SVM) [5]                 | 94.91±0.19       | 99.40±0.21       |
| SDAResNet(without any tricks)   | 99.08±0.09       | 99.26±0.10       |
| (ours best result)              | 99.30±0.08       | 99.58±0.10       |

SDAResNet without any tricks is better than other methods under training ratio of 20%, and the result under training ratio of 50% is comparable to GLANet (SVM).

The confusion matrix generated by the best combination of proposed SDAResNet and effective tricks under training ratio of 20% and 50% on PatternNet dataset are shown in Figure 7-8, respectively. As shown in Figure 8, 37 of 38 categories can achieve the accuracy of over 99%, and 32 scene classes can be fully recognized. Among these scenes, the “sparse_residential”, which is the most easily confused scene class, achieves the accuracy of 95%. All of these indicate that our method can obtain discriminative feature representation and boost classification performance very well on PatternNet dataset.

V. DISCUSSIONS

Compared with other methods, our method can achieve competitive result on the account of the fact that two important concepts are introduced: (1) In SDAResNet, spatial and channel attention is introduced to low-level feature and high-level features respectively to extract saliency scene information; (2) Various effective tricks combinations are used to assist SDAResNet to further improve the accuracy.

Two variants of proposed SDAResNet first are investigated: (1) SDAResNet-SA is the variant without spatial attention; (2) SDAResNet-CA is the variant without channel attention. Besides, the backbone of proposed SDAResNet named ResNet101 is investigated, as well as the two closest approaches (BAM+ResNet101 and CBAM+ResNet101). The comparisons are illustrated in Table 4. As can be seen from comparisons, our proposed SDAResNet exceeds all the other methods. Especially, it surpasses its counterparts: BAM+ResNet101 and CBAM+ResNet101, although all three methods involve spatial and channel attention. It shows that using different attention to different layers is superior to combining two attentions into all layers. It also indicates that our proposed SDAResNet is more effective.

In order to further illustrate the effectiveness of our proposed SDAResNet, numbers of network parameters, training time and overall accuracy are compared with the standard ResNet101 [33], BAM [29] and CBAM [12], and the results are shown in Table 5. Compared with CBAM, under the same training epochs (220 epochs) our method has fewer parameters, shorter training time, but higher accuracy. Compared with BAM and ResNet101, although our method has more
VI. CONCLUSION

In this paper, a novel SDAResNet is proposed for scene classification of RSI. The proposed SDAResNet uses ResNet101 as backbone and contains SA and CA, in which SA is integrated in low-level feature (conv2x layer of ResNet101) to emphasize saliency location information and suppress background information, while CA is embedded in high-level features (conv3x, conv4x and conv5x layer of ResNet101) to extract saliency meaningful information. Overall, our proposed SDAResNet can make full use of the attention mechanism to represent the saliency features of remote sensing scene, leading to substantial improvement in the classification accuracy. In order to further improve the performances of scene classification, the effectiveness of several tricks exceed all the comparison methods and yield the best results. As mentioned above, some effective tricks and combinations can significantly improve the accuracy.

Of course, there are few tricks that can only slightly improve or even degrade classification performance. As can be seen from Table 1, Mixup can slightly improve classification accuracy, which may be because it generally requires more training epochs to achieve better classification performance. What’s more, MultiStepLr and XavierInit reduce classification accuracy about 1%. XavierInit trick degrades the accuracy because it is used with sigmoid activations (not symmetric around 0), which will yield poor learning dynamics and initial saturation of the top hidden layer. MultiStepLr trick decreases the accuracy that may be due to the unreasonable step interval and fewer training epochs.

V. EXPERIMENTAL RESULTS

For some tricks of natural image classification proposed by predecessors, as can be seen from Table 1-2, CosLr has an obvious effect on performance improvement; it can boost the accuracy to 91.63% and 93.42% respectively under the training proportion of 10% and 20%, and the accuracy of 93.42% under the training proportion of 20% has surpassed most of the current state-of-the-art methods. LS, NoBias and Warmup tricks can also greatly boost the classification performances. The three combinations of SDAResNet and several tricks parameters, longer training time, but higher accuracy. All of these show that our proposed SDAResNet is effective.

For some tricks of natural image classification proposed by predecessors, as can be seen from Table 1-2, CosLr has an obvious effect on performance improvement; it can boost the accuracy to 91.63% and 93.42% respectively under the training proportion of 10% and 20%, and the accuracy of 93.42% under the training proportion of 20% has surpassed most of the current state-of-the-art methods. LS, NoBias and Warmup tricks can also greatly boost the classification performances. The three combinations of SDAResNet and several tricks
are investigated through ablation study. A comprehensive set of experiments on two challenging large-scale scene datasets (NWPU-RESISC45, and PatternNet) are conducted to demonstrate the effectiveness of our method, the results reveal that the combinations of proposed SDAResNet and several effective tricks outperform the current state-of-the-art approaches. The final result shows that not only attention mechanism can improve the accuracy of scene classification, but also the reasonable use of effective tricks can greatly enhance performances. In our future work, we plan to study the combination of attention mechanism and transfer learning to improve the feature representation of remote sensing scenes to obtain higher classification performance. On the other hand, we plan to summarize various tricks of image classification and validate them in remote sensing scene classification.

ACKNOWLEDGMENT
The authors would like to thank Dr. B. Imene and teacher X. Guo for helping check the grammar and expression problems in the manuscript. The authors are thankful for the anonymous reviewers for their helpful suggestions.

REFERENCES
[1] 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.
[2] 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.
[3] W. Zhou, S. Newsam, C. Li, and Z. Shao, “Patternnet: A benchmark dataset for performance evaluation of remote sensing image retrieval,” ISPRS J. Photogram. Remote Sens., vol. 145, pp. 197–209, Jan. 2018.
[4] J. Li, D. Lin, Y. Wang, G. Xu, and C. Ding, “Deep discriminative representation learning with attention map for scene classification,” 2019, arXiv:1902.07967. [Online]. Available: https://arxiv.org/pdf/1902.07967
[5] Y. Guo, J. Ji, X. Lu, H. Huo, T. Fang, and D. Li, “Global-local attention network for aerial scene classification,” IEEE Access, vol. 7, pp. 67200–67212, May 2019.
[6] Z. Lv, X. Li, B. Zhang, W. Wang, Y. Zhu, J. Hu, and S. Feng, “Managing big city information based on WebVRGIS,” IEEE Access, vol. 4, pp. 407–415, May 2016.
[7] Y. Yu and F. Liu, “Aerial scene classification via multilevel features fusion for remote sensing scene classification,” IEEE Access, vol. 11, no. 3, p. 281, Feb. 2019.
[8] E. Li, J. Xiao, 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.
[9] Q. Zhu, L. Zhang, Y. Liu, Y. Zhang, and D. Li, “A deep-local-global feature fusion framework for high spatial resolution imagery,” Remote Sens., vol. 10, no. 4, p. 568, Apr. 2018.
[10] M. Corbetta and G. L. Shulman, “Control of goal-directed and stimulus-driven attention in the brain,” Nature Rev. Neurosci., vol. 3, pp. 201–215, Mar. 2002.
[11] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, “CBAM: Convolutional block attention module,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Sep. 2018, pp. 3–19.
[12] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2018, pp. 7132–7141.
[13] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, “Residual attention network for image classification,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 3156–3164.
[39] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, “Random erasing data augmentation,” 2017, arXiv:1708.04896. [Online]. Available: https://arxiv.org/abs/1708.04896

[40] H. Ling, J. Wu, L. Wu, J. Huang, J. Chen, and P. Li, “Self residual attention network for deep face recognition,” IEEE Access, vol. 7, pp. 55159–55168, Apr. 2019.

[41] K. He, X. Zhang, and S. Ren, “Identity mappings in deep residual networks,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2016, pp. 630–645.

[42] B. Yuan, S. Li, and N. Li, “Multiscale deep features learning for land-use scene recognition,” J. Appl. Remote Sens., vol. 12, no. 1, pp. 1–25, Feb. 2018.

[43] Y. Yu and F. Liu, “Dense connectivity based two-stream deep feature fusion framework for aerial scene classification,” Remote Sens., vol. 10, no. 7, pp. 1158, Jul. 2018.

[44] 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.

[45] 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.

[46] Q. Tian, S. Wan, P. Jin, J. Xu, C. Zou, and X. Li, “A novel feature fusion with self-adaptive weight method based on deep learning for image classification,” in Proc. Pacific-Rim Conf. Multimedia, 2018, pp. 426–436.

[47] M. A. Shafaey, M. A. M. Salem, H. M. Ebeid, M. N. Al-Berry, and M. F. Tolba, “Comparison of CNNs for remote sensing scene classification,” in Proc. Int. Conf. Comput. Eng. Syst. (ICCES), Dec. 2018, pp. 27–32.

DONGEN GUO received the B.S. degree in computer application and technology from Henan Normal University, Henan, China, in 2001, and the M.S. degree in computer application and technology from the Wuhan University of Technology, Wuhan, China, in 2007. He is currently pursuing the Ph.D. degree with the School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, China. His current research interests include deep learning and remote sensing image processing.

YING XIA received the B.S. degree from the Chongqing University of Posts and Telecommunications, Chongqing, China, in 1993, the M.S. degree from Inha University, South Korea, in 2001, and the Ph.D. degree in computer application technology from Southwest Jiaotong University, China, in 2012. She is currently a Professor and the Vice Dean of the Computer Science and Technology School, Chongqing University of Posts and Telecommunications. Her research interests include spatiotemporal data analysis and cross media computing.

XIAOBO LUO received the B.S. and M.S. degrees from the China University of Geosciences, Wuhan, China, in 1999 and 2004, respectively, and the Ph.D. degree in cartography and GIS from the Institute of Remote Sensing Applications, Chinese Academy of Sciences, Beijing, China, in 2010. He is currently a Professor with the Chongqing University of Posts and Telecommunications, Chongqing, China. His research interests include urban thermal infrared remote sensing and remote sensing image classification.

***