remote sensing

MDPI

Article

A Lightweight Convolutional Neural Network Based on Group-Wise Hybrid Attention for Remote Sensing Scene Classification

Cuiping Shi [1,*], Xinlei Zhang [1], Jingwei Sun [1] and Liguo Wang [2]

[1] College of Communication and Electronic Engineering, Qiqihar University, Qiqihar 161000, China; 2020935682@qqhru.edu.cn (X.Z.); 2020910230@qqhru.edu.cn (J.S.)

[2] College of Information and Communication Engineering, Dalian Nationalities University, Dalian 116000, China; wangliguo@hrbeu.edu.cn

* Correspondence: shicuiping@qqhru.edu.com

Abstract: With the development of computer vision, attention mechanisms have been widely studied. Although the introduction of an attention module into a network model can help to improve classification performance on remote sensing scene images, the direct introduction of an attention module can increase the number of model parameters and amount of calculation, resulting in slower model operations. To solve this problem, we carried out the following work. First, a channel attention module and spatial attention module were constructed. The input features were enhanced through channel attention and spatial attention separately, and the features recalibrated by the attention modules were fused to obtain the features with hybrid attention. Then, to reduce the increase in parameters caused by the attention module, a group-wise hybrid attention module was constructed. The group-wise hybrid attention module divided the input features into four groups along the channel dimension, then used the hybrid attention mechanism to enhance the features in the channel and spatial dimensions for each group, then fused the features of the four groups along the channel dimension. Through the use of the group-wise hybrid attention module, the number of parameters and computational burden of the network were greatly reduced, and the running time of the network was shortened. Finally, a lightweight convolutional neural network was constructed based on the group-wise hybrid attention (LCNN-GWHA) for remote sensing scene image classification. Experiments on four open and challenging remote sensing scene datasets demonstrated that the proposed method has great advantages, in terms of classification accuracy, even with a very low number of parameters.

Keywords: remote sensing scene image classification; convolutional neural network (CNN); lightweight; hybrid attention; channel attention; spatial attention

check for updates

Citation: Shi, C.; Zhang, X.; Sun, J.; Wang, L. A Lightweight Convolutional Neural Network Based on Group-Wise Hybrid Attention for Remote Sensing Scene Classification. *Remote Sens.* **2022**, *14*, 161. https://doi.org/10.3390/rs14010161

Academic Editors: Xiangtao Zheng, Fulin Luo and Qi Wang

Received: 25 November 2021
Accepted: 29 December 2021
Published: 30 December 2021

1. Introduction

In recent years, convolutional neural networks (CNNs) have achieved excellent performance in many fields [1–7]. In particular, in the field of image classification [8–11], convolutional neural networks have become the most commonly used method. The core construction element of a convolutional neural network is the convolutional layer. For each convolutional layer, a group of filters is learned along the input channel to represent the local spatial mode, and feature information is extracted by fusing the spatial and channel information of the local receptive field. Improving the quality of spatial coding of the whole feature level of a convolutional neural network to enhance the representation ability of the network is an effective way to improve the performance of the network. It has been shown, with VGGNet [12], that increasing the depth of the network can significantly improve the performance of the network. ResNet [13] addressed the problem of performance degradation caused by network deepening: it expanded the network depth to 150 or even 1000 layers, based on VGGNet, and achieved good performance. InceptionNet [14] divided the input features into four channels, in which different convolution filters were used to adapt to different scales of features. Finally, the extracted features were fused along the channel dimension to improve network performance by increasing the width of the network. Subsequently, a series of lightweight convolutional neural networks were proposed. These networks reduce the complexity of the model while also having good feature extraction ability. Xception [15] and MoblenetV1 [16] introduced depth-wise separable convolution, instead of traditional convolution, for lightweight networks. Depth-wise separable convolution divides traditional convolution into depth-wise convolution and pointwise convolution to reduce the number of parameters of the model. MobilenetV2 [17] proved the validity of depthwise separable convolution. Grouping convolution also provides a way to improve network representation by increasing the width of the network while reducing the computational cost of the network. Assuming that g is used to represent the number of groups, both the number of parameters and the calculation cost of grouping convolution are $1/g$ that of traditional convolution. Grouped convolution was first used in AlexNet due to hardware constraints and served to reduce the associated computational costs. By using grouped convolution in ResNeXts [18] and increasing the depth and width of the model, the classification accuracy was greatly improved. ShuffleNet [19,20] proposed channel shuffling, which can alleviate the loss of information due to a lack of information exchange between channels caused by grouping after grouping convolution.

The improvement of network performance by use of an attention mechanism has been demonstrated in many tasks. SENet [21] improved the network performance by explicitly modeling the dependences between channels. SENet consists of two operations: squeeze and excitation. The squeeze operation extrudes the features spatially through a global average pooling operation to obtain a value with a global receptive field. The resulting values from the excitation are obtained through two consecutive fully connected layers, and the channel attention map is derived from the correlation between the channels, which is used to recalibrate the features. SKNet [22] added two operations, split and fuse, on the basis of SENet. Split operations employ convolution kernels with different receptive field sizes to capture multiscale semantic information. Fusion operations fuse multiscale semantic information, enhance feature diversity, and aggregate feature maps from different size convolution kernels, according to their weights, by use of an SE module. It is also an effective method to improve the performance of the network, by explicitly modeling the dependence between channels and spatial information. CBAM [23] extracts channel attention and spatial attention through a combination of global average pooling and maximum pooling. The input features are enhanced in space and channel by using spatial attention and channel attention, respectively. Finally, the enhanced features are fused to improve the performance of the model. Wang et al. [24] designed a circular attention structure to reduce advanced semantic and spatial features to reduce the number of learning parameters. Tong et al. [25] introduced an attention mechanism into DenseNet to adaptively enhance the weights of important feature channels. Yu et al. [26] improved channel attention and proposed a hierarchical attention mechanism by combining the improved channel attention with a ResNet network. Alhichri et al. [27] proposed a deep attention convolution neural network to learn feature maps from large scene regions. We note that although the introduction of an attention module to a network can help to improve the network performance, adding an attention mechanism directly to the network increases the amount of network parameters and required calculations, thus reducing the running speed of the model. To solve this problem, we first constructed a new channel attention and spatial attention module to recalibrate the features. The channel attention sets the channel compression ratio of the SE module to $1/4$ and replaces the fully connected layer with 1×1 convolution. We compressed the input features using a $5 \times 5 \times 1$ convolution kernel and achieved spatial attention by using the Sigmoid activation function for the compressed features. Next, we propose a group-wise hybrid attention method, which groups input features and introduces hybrid attention to each group. Each group is re-calibrated using the spatial attention and channel attention, respectively, and the recalibrated features are fused.

The main contributions of this study are as follows:

(1) Based on the SE module, we propose a channel attention module which is more suitable for remote sensing scene image classification. In the proposed method, the channel compression ratio is set to $1/4$, and a 1×1 convolution kernel is adopted instead of a fully connected layer. The 1×1 convolution does not destroy the spatial structure of the features, and the size of the input features can be arbitrary.

(2) We propose a spatial attention module with a simpler implementation process. Channels are compressed using a $5 \times 5 \times 1$ convolution kernel directly, and spatial attention features are obtained using the Sigmoid activation function. The convolution kernel of 5×5 is helpful in providing a large receptive field, which can extract more spatial features.

(3) A hybrid attention model is constructed by combining channel attention and spatial attention in parallel, which has higher activation and can learn more meaningful features.

(4) To alleviate the problem that the introduction of attention leads to an increased number of parameters, we further propose a group-wise hybrid attention module. This module first divides input features into four groups in the channel dimension, then introduces hybrid attention to each group. Each group is recalibrated separately with spatial attention and channel attention and, finally, the rescaled features are fused in the channel dimension. Moreover, a lightweight convolutional neural network is constructed based on group-wise hybrid attention (LCNN-GWHA), which is shown to be an effective method for remote sensing scene image classification.

The remainder of this paper is structured as follows. In Section 2, the channel attention, spatial attention, hybrid attention, group-wise hybrid attention, and the proposed LCNN-GWHA method are described in detail. In Section 3, experiments and analyses are carried out, including a comparison with some state-of-the-art methods, in order to demonstrate the superior performance of the proposed method. In Section 4, the feature extraction ability of the proposed LCNN-GWHA method is evaluated by visualization. The conclusion of this paper is given in Section 5.

2. Methods

2.1. Traditional Convolution Process

Assuming that the input feature is $X \in \mathbb{R}^{H \times W \times C'}$, the output feature $Y \in \mathbb{R}^{H \times W \times C}$ is obtained by the convolution operation $F(\cdot)$, as shown in Figure 1. The set of convolution kernels is represented by $U = [u_1, u_2, \ldots, u_C]$, where u_C represents the cth convolution kernel. Then, the features of the cth channel of the output, $Y = [y_1, y_2, \ldots, y_C]$, can be represented as:

$$y_C = u_C * X = \sum_{i=1}^{C'} u_C^i * x^i \tag{1}$$

where $*$ represents the convolution operation, $u_C = [u_C^1, u_C^2, \ldots, u_C^{C'}]$, $X = [x^1, x^2, \ldots, x^{C'}]$, u_C^i represents the ith channel of the cth convolution kernel, x^i represents the ith channel of the input feature, and $u_C^i * x^i$ represents the spatial features learned from the ith channel. $\sum_{i=1}^{C'} u_C^i * x^i$ means summing the spatial features learned by all channels through convolution, such that the final output feature y_C includes both channel features and spatial features. Channel attention and spatial attention can be obtained by modeling the correlations between different channels and different spatial correlations, respectively.

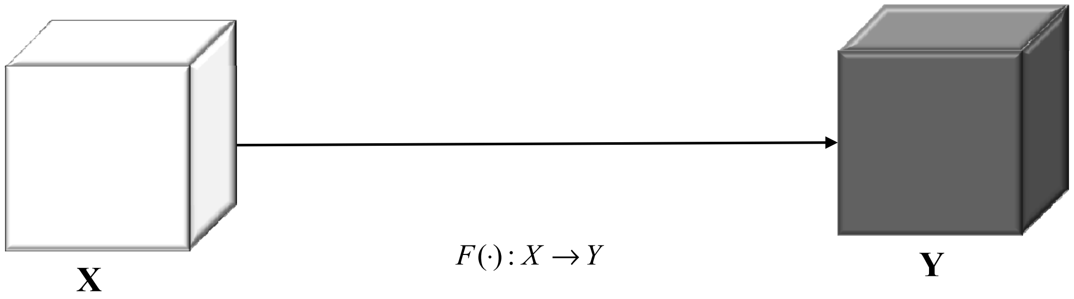

Figure 1. Features obtained by traditional convolution.

2.2. Channel Attention

Channel attention first obtains the feature $M \in \mathbb{R}^{1\times1\times C}$ by spatially compressing the input features, then convolutes the compressed feature $M \in \mathbb{R}^{1\times1\times C}$ to model the correlation among the different channels. Channel attention assigns different weight coefficients to each channel to enhance important features and suppress unimportant features. The process of channel attention is shown in Figure 2.

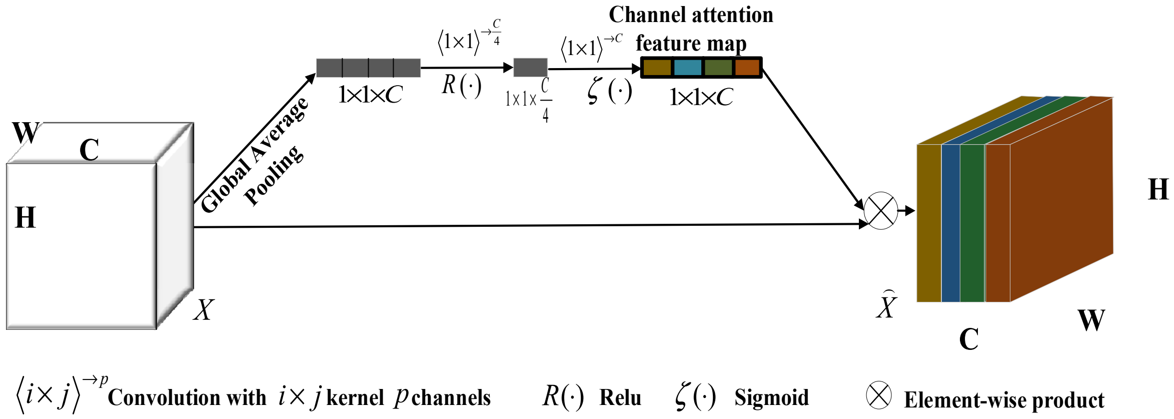
Channel attention
feature map
$\langle 1\times1\rangle^{\to\frac{C}{4}}$
$R(\cdot)$
$\langle 1\times1\rangle^{\to C}$
$\zeta(\cdot)$
$1\times1\times C$
$1\times1\times\frac{C}{4}$
$1\times1\times C$
Global Average Pooling
W
C
H
X
$\widehat{X}$
H
C
W
$\langle i\times j\rangle^{\to p}$ Convolution with $i\times j$ kernel p channels
$R(\cdot)$ Relu
$\zeta(\cdot)$ Sigmoid
$\otimes$ Element-wise product

Figure 2. Channel Attention Module.

Suppose the input feature is $X = [x_1, x_2, \ldots, x_C]$, where $x_i \in \mathbb{R}^{H\times W\times C}$ represents the feature of the ith channel. The input feature $X = [x_1, x_2, \ldots, x_C]$ is spatially compressed by global average pooling to obtain the feature $M \in \mathbb{R}^{1\times1\times C}$, and the result M_i for the ith channel feature can be represented as

$$M_i = \frac{1}{H\times W}\sum_{m=1}^{H}\sum_{n=1}^{W} x_i(m,n) \tag{2}$$

where, H and W represent the height and width of feature x_i, respectively. The feature $M \in \mathbb{R}^{1\times1\times C}$ compressed by global average pooling, can reflect global spatial information. Then, the feature $M \in \mathbb{R}^{1\times1\times C}$, with the global receptive field, is subjected to two continuous 1×1 convolution operations to obtain $\widehat{M}_C = W_2(R(W_1 M_C))$, where $W_1 \in \mathbb{R}^{C\times\frac{C}{4}}$ and $W_2 \in \mathbb{R}^{\frac{C}{4}\times C}$ are the weights of the first and second 1×1 convolutions, respectively, and $R(\cdot)$ represents the activation function Relu. The first 1×1 convolution has the function of dimension reduction, which reduces the number of feature channels to one quarter of the original number of channels, and then the nonlinear relationship between channels is increased by Relu. The second convolution restores the number of channels, normalizes the learned activation values of each channel into the range [0, 1] through the Sigmoid activation function and obtains the channel attention feature $\zeta(\widehat{M})$, where $\zeta(\cdot)$ represents the

Sigmoid activation function. Finally, the input feature $X = [x_1, x_2, \ldots, x_C]$ is recalibrated through the channel attention feature, $\zeta(\widehat{M})$, to obtain $\widehat{X}$:

$$\widehat{X} = [\zeta(\widehat{M}_1)x_1, \zeta(\widehat{M}_2)x_2, \ldots, \zeta(\widehat{M}_C)x_C] \tag{3}$$

where, $\zeta(\widehat{M}_i)$ represents the importance of the ith channel. These activation values can be adaptively adjusted by the convolutional neural network. The channel attention module can enhance important features and suppress unimportant features.

2.3. Spatial Attention

Spatial attention first squeezes the channel to obtain the feature $\breve{M} \in \mathbb{R}^{H \times W \times 1}$. Then, different weight coefficients are assigned to different locations of the compressed feature $\breve{M} \in \mathbb{R}^{H \times W \times 1}$, through an activation function which enhances the target areas of interest and suppresses the unimportant areas. The process of spatial attention is depicted in Figure 3. Assuming that the input feature is $X = [x^{1,1}, x^{1,2}, \ldots, x^{m,n}, \ldots, x^{H,W}]$, $x^{m,n} \in \mathbb{R}^{1 \times 1 \times C}$ represents the feature at the corresponding spatial location (m, n), where $m \in \{1, 2, \ldots, H\}$ and $n \in \{1, 2, \ldots, W\}$, and the convolution kernel is $f_{sq} \in \mathbb{R}^{5 \times 5 \times C \times 1}$. The calculation process of feature $\breve{M} \in \mathbb{R}^{H \times W \times 1}$ after channel compression is:

$$\breve{M} = f_{sq} * X \tag{4}$$

where $\breve{M} \in [\breve{M}^{1,1,}, \breve{M}^{1,2}, \ldots, \breve{M}^{m,n}, \ldots, \breve{M}^{H,W}]$ and $\breve{M}^{m,n} \in \mathbb{R}^{1 \times 1 \times 1}$ represents the linear combination of all channels at spatial position (m, n). Then, the Sigmoid activation function $\zeta(\cdot)$ is used to normalize it into the range [0, 1] to obtain the spatial attention feature $\zeta(\breve{M}^{m,n})$. Finally, the input feature $X = [x^{1,1}, x^{1,2}, \ldots, x^{m,n}, \ldots, x^{H,W}]$ is recalibrated through the spatial attention feature $\zeta(\breve{M}^{m,n})$ to obtain $\breve{X}$; that is:

$$\breve{X} = [\zeta(\breve{M}^{1,1})x^{1,1}, \zeta(\breve{M}^{1,2})x^{1,2}, \ldots, \zeta(\breve{M}^{m,n})x^{m,n}, \ldots, \zeta(\breve{M}^{H,W})x^{H,W}] \tag{5}$$

where $\zeta(\breve{M}^{m,n})$ represents the importance at the spatial position (m, n) of the feature. This enhances the importance of regions of interest and suppresses unimportant spatial locations.

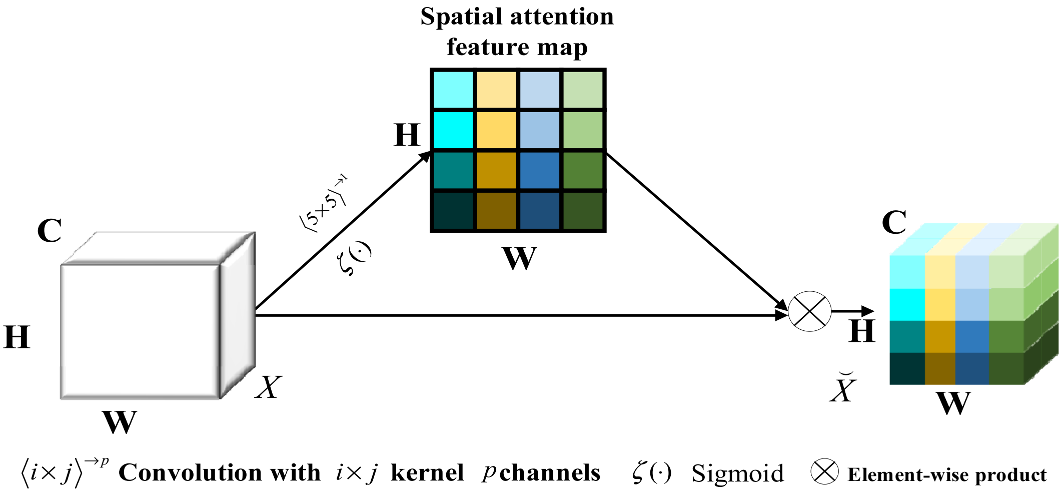
Spatial attention feature map
H
W
C
H
W
X
$\langle 5\times5\rangle^{\to 1}$
$\zeta(\cdot)$
C
H
W
$\breve{X}$
$\langle i\times j\rangle^{\to p}$ Convolution with $i\times j$ kernel p channels
$\zeta(\cdot)$ Sigmoid
$\otimes$ Element-wise product

Figure 3. Spatial Attention Module.

2.4. Group-Wise Hybrid Attention

Spatial attention ignores the information interaction between channels, while channel attention ignores the information interaction in the spatial dimension. To solve this problem, hybrid attention is proposed. The output feature $\breve{X}$ of the spatial attention module and the output feature $\widehat{X}$ of the channel attention module are fused to obtain the feature $\widetilde{X}$, which is calibrated in space and channel respectively. $\phi(p_i^{m,n})$ is used to represent the importance of the ith channel at the spatial position (m, n).

The addition of hybrid attention to the network can improve the network performance, but inevitably increases the computational cost of the network and reduces the running speed of the model. Therefore, grouping convolution was designed to extract features more efficiently based on an attention mechanism, as shown in Figure 4. Assuming that the input features are $X = [x_1^{1,1}, x_1^{1,2}, \ldots, x_i^{m,n}, \ldots, x_C^{H,W}]$, first, the input features are grouped along the channel dimension to obtain $X_1 = [x_1^{1,1}, x_1^{1,2}, \ldots, x_i^{m,n}, \ldots, x_{\frac{C}{4}}^{H,W}]$, $X_2 = [x_{\frac{C}{4}}^{1,1}, x_{\frac{C}{4}}^{1,2}, \ldots, x_i^{m,n}, \ldots, x_{\frac{C}{2}}^{H,W}]$, $X_3 = [x_{\frac{C}{2}}^{1,1}, x_{\frac{C}{2}}^{1,2}, \ldots, x_i^{m,n}, \ldots, x_{\frac{3C}{4}}^{H,W}]$ and $X_4 = [x_{\frac{3C}{4}}^{1,1}, x_{\frac{3C}{4}}^{1,2}, \ldots, x_i^{m,n}, \ldots, x_C^{H,W}]$. Then, for the four grouped features X_1, X_2, X_3 and X_4, channel attention and spatial attention are utilized to calibrate the features, respectively, and the enhanced results $\widetilde{X}_1$, $\widetilde{X}_2$, $\widetilde{X}_3$ and $\widetilde{X}_4$ are obtained. The specific process is as follow. When the grouped feature is $X_1 = [x_1^{1,1}, x_1^{1,2}, \ldots, x_i^{m,n}, \ldots, x_{\frac{C}{4}}^{H,W}]$, where $x_i^{m,n}$ represents the feature with spatial position (m, n) in the ith channel, the result for $\widetilde{X}_1$ after applying the hybrid attention is shown in formula (6). In formula (6), $\phi(p_i^{m,n})x_i^{m,n}$ represents the result of feature enhancement of each feature $x_i^{m,n}$ in X_1 in the spatial and channel dimensions, respectively.

$$\widetilde{X}_1 = [\phi(p_1^{1,1})x_1^{1,1}, \phi(p_1^{1,2})x_1^{1,2}, \ldots, \phi(p_i^{m,n})x_i^{m,n}, \ldots, \phi(p_{\frac{C}{4}}^{H,W})x_{\frac{C}{4}}^{H,W}] \tag{6}$$

When the grouped feature is $X_2 = [x_{\frac{C}{4}}^{1,1}, x_{\frac{C}{4}}^{1,2}, \ldots, x_i^{m,n}, \ldots, x_{\frac{C}{2}}^{H,W}]$, the result of $\widetilde{X}_2$ after hybrid attention is shown in formula (7). In formula (7), $\phi(p_i^{m,n})x_i^{m,n}$ represents the enhanced result of each feature $x_i^{m,n}$ in X_2 in the spatial and channel dimensions, respectively.

$$\widetilde{X}_2 = [\phi(p_{\frac{C}{4}}^{1,1})x_{\frac{C}{4}}^{1,1}, \phi(p_{\frac{C}{4}}^{1,2})x_{\frac{C}{4}}^{1,2}, \ldots, \phi(p_i^{m,n})x_i^{m,n}, \ldots, \phi(p_{\frac{C}{2}}^{H,W})x_{\frac{C}{2}}^{H,W}] \tag{7}$$

When the grouped feature is $X_3 = [x_{\frac{C}{2}}^{1,1}, x_{\frac{C}{2}}^{1,2}, \ldots, x_i^{m,n}, \ldots, x_{\frac{3C}{4}}^{H,W}]$, the result of $\widetilde{X}_3$ after hybrid attention is shown in formula (8). In formula (8), $\phi(p_i^{m,n})x_i^{m,n}$ represents the result of feature enhancement of each feature $x_i^{m,n}$ in X_3 in the spatial and channel dimensions, respectively.

$$\widetilde{X}_3 = [\phi(p_{\frac{C}{2}}^{1,1})x_{\frac{C}{2}}^{1,1}, \phi(p_{\frac{C}{2}}^{1,2})x_{\frac{C}{2}}^{1,2}, \ldots, \phi(p_i^{m,n})x_i^{m,n}, \ldots, \phi(p_{\frac{3C}{4}}^{H,W})x_{\frac{3C}{4}}^{H,W}] \tag{8}$$

When the grouped feature is $X_4 = [x_{\frac{3C}{4}}^{1,1}, x_{\frac{3C}{4}}^{1,2}, \ldots, x_i^{m,n}, \ldots, x_C^{H,W}]$, the result of $\widetilde{X}_4$ after hybrid attention is shown in formula (9). In formula (9), $\phi(p_i^{m,n})x_i^{m,n}$ represents the result of feature enhancement for each feature $x_i^{m,n}$ in X_4 in the spatial and channel dimensions, respectively.

$$\widetilde{X}_4 = [\phi(p_{\frac{3}{4}C}^{1,1})x_{\frac{3}{4}C}^{1,1}, \phi(p_{\frac{3C}{4}}^{1,2})x_{\frac{3C}{4}}^{1,2}, \ldots, \phi(p_i^{m,n})x_i^{m,n}, \ldots, \phi(p_C^{H,W})x_C^{H,W}] \tag{9}$$

Finally, the enhanced features $\widetilde{X}_1$, $\widetilde{X}_2$, $\widetilde{X}_3$, and $\widetilde{X}_4$ are fused along the channel direction to obtain the output feature Y, as shown in formula (10). In formula (10), $\oplus$ represents feature fusion along the channel dimension.

$$Y = \widetilde{X}_1 \oplus \widetilde{X}_2 \oplus \widetilde{X}_3 \oplus \widetilde{X}_4 \tag{10}$$

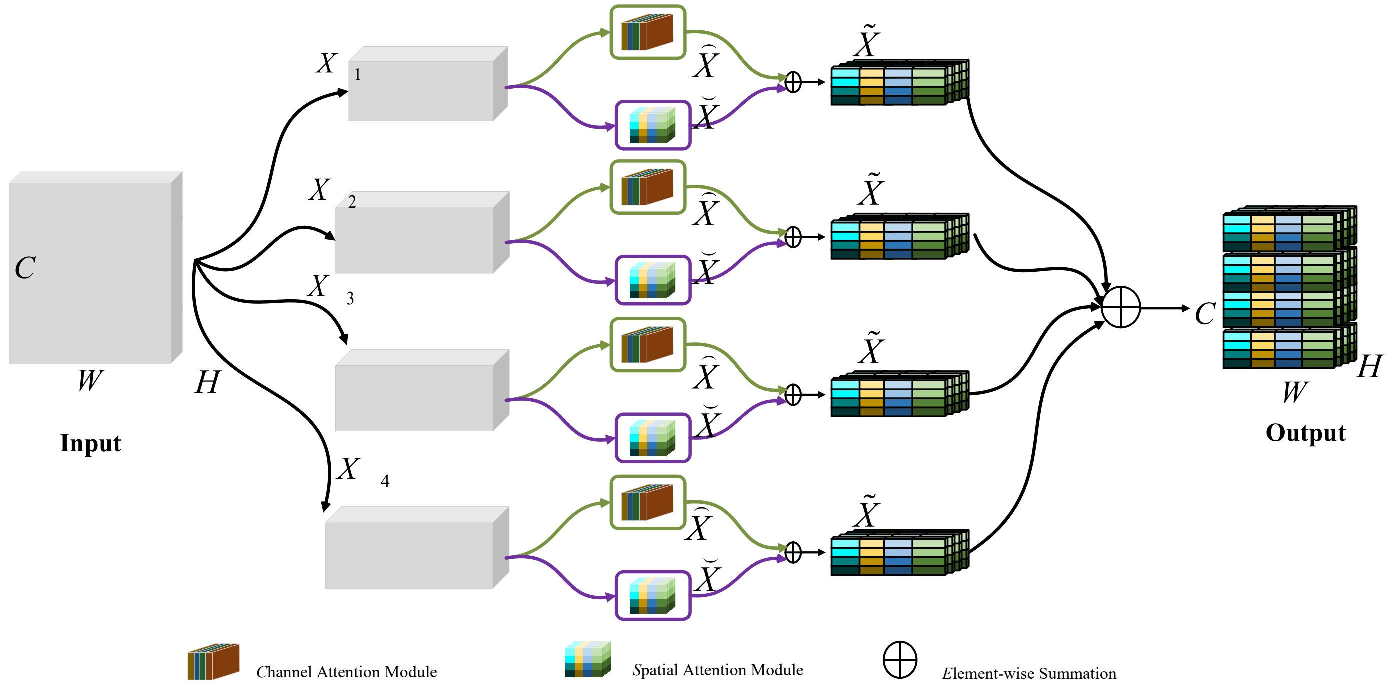

Figure 4. Group-wise hybrid attention module (GWHAM). W, H, and C are the width, height and number of channels of the feature, respectively. $\widehat{X}$ and $\breve{X}$ are the output features of channel attention and spatial attention, respectively. $\widetilde{X}$ is the hybrid attention feature after fusion.

2.5. Lightweight Convolution Neural Network Based on Group-Wise Hybrid Attention (LCNN-GWHA)

The structure of the proposed lightweight modular LCNN-GWHA method is shown in Figure 5. The structure is mainly composed of convolution, the group-wise hybrid attention module, a global average pooling layer, and the classifier. First, the shallow feature information is extracted through two consecutive convolution operations and then the deeper features are extracted through six group-wise hybrid attention modules (GWHAM). The output features of the last convolution are mapped to each category using global average pooling (GAP). The use of global average pooling does not increase the weight parameters and can effectively avoid the over-fitting phenomenon in the training process. Finally, the softmax function classifier is adopted to classify the features.

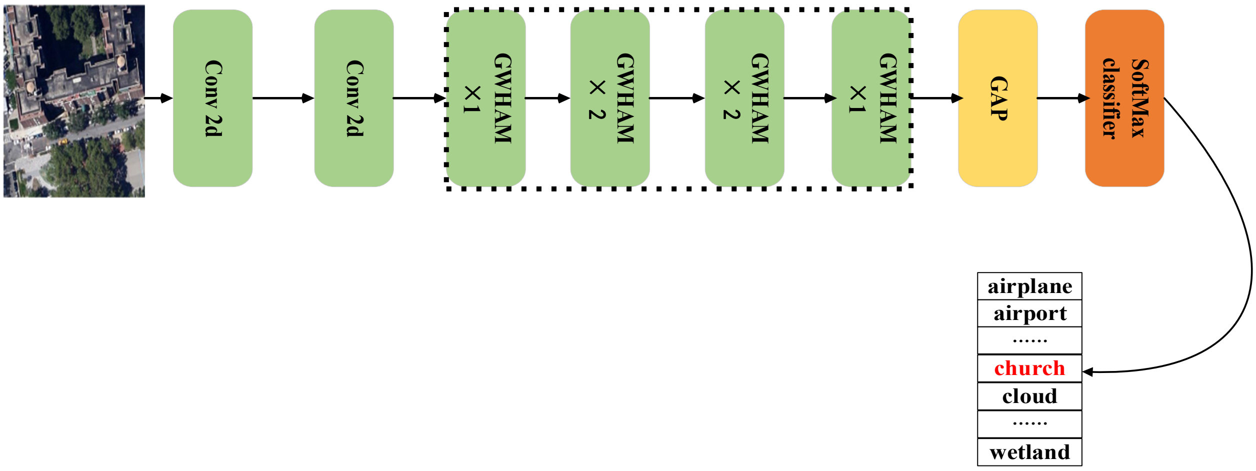

Figure 5. Overall flowchart of the proposed LCNN-GWHA method. (GWHAM refers to the group-wise hybrid attention modules, and GAP denotes global average pooling).

If a fully connected layer (FC) with the classification number n is used to classify the average pooled output result $g_i \in G$, and the classification result is $Q \in [q_1, q_2, \ldots, q_i, \ldots, q_N] \equiv FC(g_i)$, the output result $S = [s_1, s_2, \ldots, s_i, \ldots, s_N]$ from softmax can be represented as:

$$s_i = \frac{e^{Q[i-1]}}{\sum_{k=0}^{N-1} e^{Q[k]}} \tag{11}$$

where Q_i represents the i-th element in Q (the index starts from 0). Cross-entropy is adopted as the loss function. Assuming that $T = [t_1, t_2, \ldots, t_i, \ldots, t_N]$ represents the encoding result of the input sample label. Then, the loss function can be represented as:

$$L = -\sum_{i=1}^{N} t_i \log(s_i) \tag{12}$$

where N represents the number of categories, s_i represents the output result of Softmax, and the input sample label adopts the one-hot coding rule.

3. Experiments

In this section, some evaluation indicators are adopted to evaluate the proposed LCNN-GWHA method. The proposed LCNN-GWHA method was compared with various state-of-the-art methods on four challenging datasets. To make a fair comparison, both the proposed method and those methods used for comparison were carried out under the same experimental environment and super parameters. The experimental results indicate that the proposed method can classify remote sensing scene images more accurately and has obvious advantages in terms of parameter quantity and running speed.

3.1. Dataset Settings

Experiments were performed on four commonly used datasets: UCM21 [28], RSSCN7 [29], AID [30], and NWPU45 [31]. In Table 1, the number of images per category, the number of scene categories, the total number of images, the spatial resolution of images, and the size of images in the four datasets are listed. To avoid memory overflow during the training process, bilinear interpolation was used to resize the input images to 256 × 256.

Table 1. Description of four datasets.

Datasets	Number of Images Per Class	Number of Classes	Total Number of Images	Spatial Resolution (m)	Image Size
UCM21	100	21	2100	0.3	256 × 256
RSSCN7	400	7	2800	-	400 × 400
AID	200–400	30	10,000	0.5–0.8	600 × 600
NWPU45	700	45	31,500	0.2–30	256 × 256

3.2. Setting of the Experiments

The stratified sampling method was adopted to divide the datasets to avoid the risk of sampling bias. In addition, so that the proposed method and the compared method used the same training samples, random seeds were set during the division of training and test samples. For the UCM21 [28] dataset, the training proportion was set to 80%; For the RSSCN7 [29] dataset, the training proportion was set to 50%; For the AID [30] dataset, the training proportions were set to 20% and 50%, respectively. Finally, for the NWPU45 [31] dataset, the training proportions were set to 10% and 20%, respectively. The parameters and equipment configuration used in the experiments are listed in Table 2, while the training parameters used for the proposed LCNN-GWHA method are given in Table 3.

Table 2. Experimental environment and parameter settings.

Item	Contents
Processor	AMD Ryzen 7 4800 H with Radeon Graphics@2.90 GHz
Memory	16 GB
Operating system	Windows10
Solid state hard disk	512 GB
Software	PyCharm Community Edition 2020.3.2
GPU	NVIDIA GeForce RTX2060 6 GB
Keras	v2.2.5
Initial study rate	0.01
Momentum	0.9

Table 3. Training Parameters for Proposed LCNN-GWHA Methods.

Input	Operator	Repeated Times	Stride	Output Channels	Output
256 × 256 × 3	Conv 2d 3 × 3	1	2	32	128 × 128 × 32
128 × 128 × 32	Conv 2d 3 × 3	1	2	64	64 × 64 × 64
64 × 64 × 64	GWHAM	1	2	128	32 × 32 × 128
32 × 32 × 128	GWHAM	2	2	256	16 × 16 × 256
16 × 16 × 256	GWHAM	2	2	512	8 × 8 × 512
8 × 8 × 512	GWHAM	1	2	512	4 × 4 × 512
4 × 4 × 512	Avgpool	1	-	512	1 × 1 × 512
1 × 1 × 512	Dense	1	-	7	1 × 1 × 7

3.3. Performance of the Proposed Model

Table 4 details the performance of the proposed method on the four data sets with various training ratios. In order to verify the performance of the proposed method, the overall accuracy (OA), average accuracy (AA), kappa coefficient (kappa), and F1 score (F1) were adopted as evaluation indices for the experiments. The OA represents the percentage of correct classification in the test set; AA represents the ratio of the number of correctly predicted samples in each category to the total number of samples in the category; the F1 score is the weighted average of accuracy and recall, which is used to measure the robustness of the model, and the Kappa coefficient is used for consistency evaluation, in terms of whether the predicted results are consistent with the actual classification results. It can be seen from Table 4 that the OA and AA of the proposed method on the four data sets reached more than 90%, and the difference between the OA and AA was less than 1%, indicating that the proposed method has strong generalization ability. The Kappa coefficient was more than 90%, which demonstrates that the predicted value obtained by the proposed method was almost consistent with the real value. The F1 value results also proved that the proposed method has strong robustness.

Table 4. Performance Indices for the Proposed LCNN-GWHA Model on Four Datasets.

Datasets	OA (%)	Kappa (%)	AA (%)	F1 (%)
RSSCN7	97.78	97.42	97.70	97.71
UCM21	99.76	99.75	99.49	99.52
AID (50/50)	97.64	97.55	97.05	97.16
AID (20/80)	93.85	93.63	93.60	93.67
NWPU45 (20/80)	94.26	94.13	93.95	94.10
NWPU45 (10/90)	92.24	92.04	92.15	92.20

3.3.1. Experimental Results of the RSSCN7 Dataset

The comparison results for the RSSCN7 dataset are shown in Table 5. In this dataset, the proportion of samples used for training was 50% of the total number of samples. The proposed method had 0.3 M parameters and 97.78% classification accuracy. It had the highest accuracy and the least number of parameters compared with all of the methods used for comparison. The OA of the proposed method was 2.57% higher than that of ADFF [32], 2.24% higher than that of Coutourlet CNN [33], and 3.07% higher than that of SE-MDPMNet [34]. The confusion matrix for the proposed method of the RSSCN7 dataset is shown in Figure 6. It can be seen from Figure 6 that the proposed method achieved 99% classification accuracy for 'Forest', 'Parking', and 'RiverLake' categories, indicating that these scenarios had high interclass differences and intraclass similarities. 'Grass' was a scenario with a minimum classification accuracy of 95%, some of which were incorrectly classified into 'Forest' and 'Field' scenarios, as the three scenarios are similar and have small intraclass differences, resulting in the incorrect classification of grasslands.

Table 5. Performance comparison of the proposed model with some advanced methods on the RSSCN7 dataset.

Network Model	OA (%)	Number of Parameters
VGG16+SVM Method [30]	87.18	130 M
Variable-Weighted Multi-Fusion Method [35]	89.1	-
TSDFF Method [36]	92.37 ± 0.72	-
ResNet+SPM-CRC Method [37]	93.86	23 M
ResNet+WSPM-CRC Method [37]	93.9	23 M
LCNN-BFF Method [38]	94.64 ± 0.21	6.2 M
ADFF [32]	95.21 ± 0.50	23 M
Coutourlet CNN [33]	95.54 ± 0.17	12.6 M
SE-MDPMNet [34]	94.71 ± 0.15	5.17 M
Proposed Method	97.78 ± 0.12	0.3 M

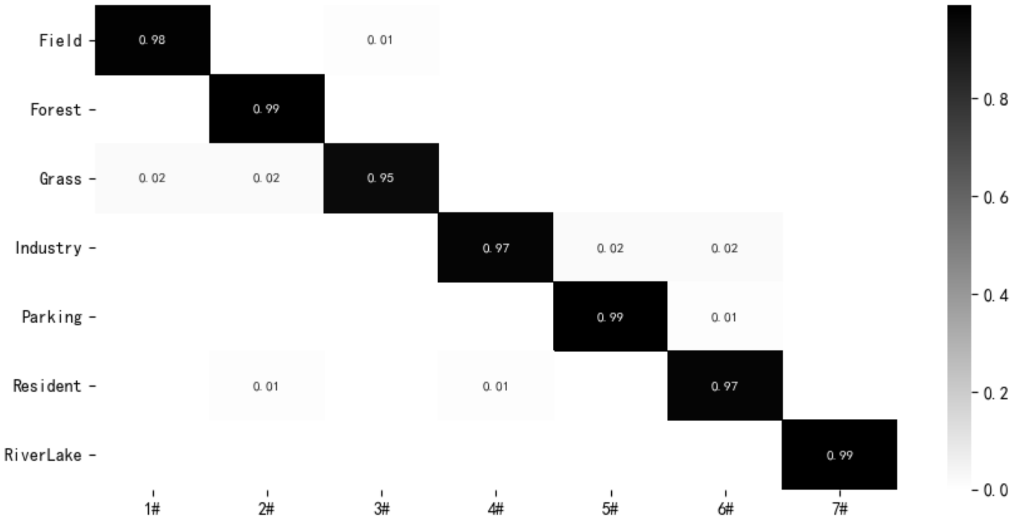

Figure 6. Confusion matrix of the proposed LCNN-GWHA method of the RSSCN7 Dataset (50/50).

3.3.2. Experimental Results of the UCM21 Dataset

The division proportion for the UCM21 dataset was set as training/test = 8:2, and the experimental results on the UCM21 dataset are shown in Table 6. It can be seen from Table 6 that the OAs of some methods on this dataset exceeded 99%. In this case, the number of parameters was an important evaluation index. The parameter amount of the proposed method was 0.31 M and, the classification accuracy was 99.76%, 5.89 M less than that of the LCNN-BFF method [38] parameters with 99.29% accuracy, and 21.69 M less than that of the Inceptionv3+CapsNet method [39] parameters with 99.05% accuracy. The proposed method achieves high classification accuracy while greatly reducing the number of parameters of the model.

Table 6. Performance Comparison of the Proposed Model with Some Advanced Methods on the UCM21 Dataset.

Network Model	OA (%)	Number of Parameters
Variable-Weighted Multi-Fusion [35]	97.79	-
ResNet+WSPM-CRC [37]	97.95	23 M
ADFF [32]	98.81 ± 0.51	23 M
LCNN-BFF [38]	99.29 ± 0.24	6.2 M
VGG16 with MSCP [40]	98.36 ± 0.58	-
Gated Bidirectional+global feature [41]	98.57 ± 0.48	138 M
Feature Aggregation CNN [42]	98.81 ± 0.24	130 M
Skip-Connected CNN [43]	98.04 ± 0.23	6 M
Discriminative CNN [44]	98.93 ± 0.10	130 M
VGG16-DF [45]	98.97	130 M
Scale-Free CNN [46]	99.05 ± 0.27	130 M
Inceptionv3+CapsNet [39]	99.05 ± 0.24	22 M
DDRL-AM [47]	99.05 ± 0.08	-
Semi-Supervised Representation Learning [48]	94.05 ± 1.2	210 M
Multiple Resolution BlockFeature [49]	94.19 ± 1.5	-
Siamese CNN [50]	94.29	-
Siamese ResNet50 with R.D [51]	94.76	-
Bidirectional Adaptive Feature Fusion [52]	95.48	130 M
Multiscale CNN [53]	96.66 ± 0.90	60 M
VGG_VD16 with SAFF [54]	97.02 ± 0.78	15 M
Proposed Method	99.76 ± 0.25	0.3 M

The confusion matrix of the proposed method on the UCM21 dataset with a training:test = 8:2 is shown in Figure 7. As can be seen from Figure 7, except for the 'mediumresidential' scene, all other scenes were fully recognized. This was because the two scenes 'mediumresidential' and 'mobilehomepark' were very similar in appearance, resulting in confusion in classification.

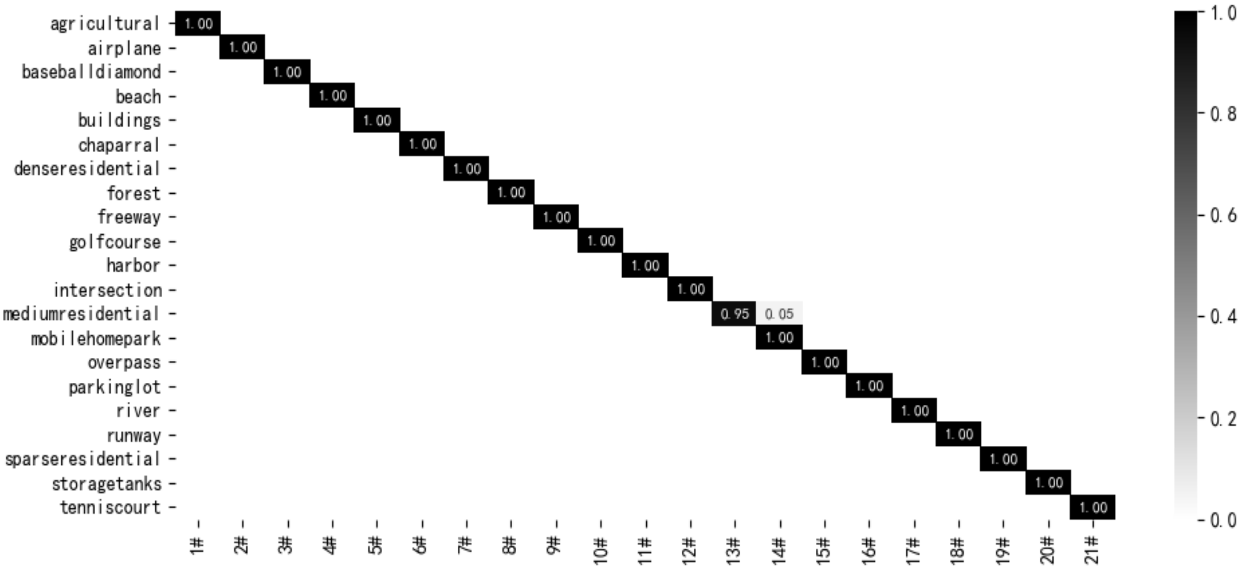

Figure 7. Confusion Matrix for the LCNN-GWHA Method on the UCM21 Dataset (80/20).

3.3.3. Experimental Results on the AID Dataset

For the AID dataset, experiments were performed with training test = 2:8 and training test = 5:5, respectively. The experimental results are shown in Table 7. It can be seen that when the training ratio was 20%, our method achieved the best performance with the least parameters, and the classification accuracy reached 93.85%, which is 0.58% higher than that of Inception V3 [55], and 1.46% higher than that of ResNet50 [55]. When the training proportion was 50%, compared with the Discriminative CNN [44] method, the Inception V3 [55] method, and the Skip Connected CNN [43] method, the proposed method had great advantages in classification accuracy, with the amount of model parameters being only 0.2, 0.6, and 1.6% of the abovementioned methods, respectively.

Table 7. Performance Comparison of the Proposed Model with Some Advanced Methods on the AID Dataset.

Network Model	OA (20/80) (%)	OA (50/50) (%)	Number of Parameters
VGG16+CapsNet [39]	91.63 ± 0.19	94.74 ± 0.17	130 M
VGG_VD16 with SAFF [54]	90.25 ± 0.29	93.83 ± 0.28	15 M
Discriminative CNN [44]	90.82 ± 0.16	96.89 ± 0.10	130 M
Fine-tuning [30]	86.59 ± 0.29	89.64 ± 0.36	130 M
Skip-Connected CNN [43]	91.10 ± 0.15	93.30 ± 0.13	6 M
LCNN-BFF [38]	91.66 ± 0.48	94.64 ± 0.16	6.2 M
Gated Bidirectional [41]	90.16 ± 0.24	93.72 ± 0.34	18 M
Gated Bidirectional+global feature [41]	92.20 ± 0.23	95.48 ± 0.12	138 M
TSDFF [36]	-	91.8	-
AlexNet with MSCP [40]	88.99 ± 0.38	92.36 ± 0.21	-
VGG16 with MSCP [40]	91.52 ± 0.21	94.42 ± 0.17	-
ResNet50 [55]	92.39 ± 0.15	94.69 ± 0.19	25.61 M
InceptionV3 [55]	93.27 ± 0.17	95.07 ± 0.22	45.37 M
Proposed Method	93.85 ± 0.16	97.64 ± 0.28	0.3 M

The confusion matrix for the proposed method on the AID dataset with training test = 5:5 is shown in Figure 8. The three scenarios of 'Meadow, 'Viaduct' and 'Sparse Residential' achieved 100% correct classification. The lowest classification accuracy, for the 'industrial' category, was 94%, as the 'Industrial' and 'Commercial' areas have similar architectural styles, resulting in some 'Industrial' scenes being incorrectly classified as 'Commercial'. Moreover, 'Desert' and 'BareLand' were also easily confused, as they have similar surface appearance, resulting in low classification accuracy for desert areas. Nevertheless, compared with other state-of-the-art classification methods, the proposed method still achieved higher classification accuracy.

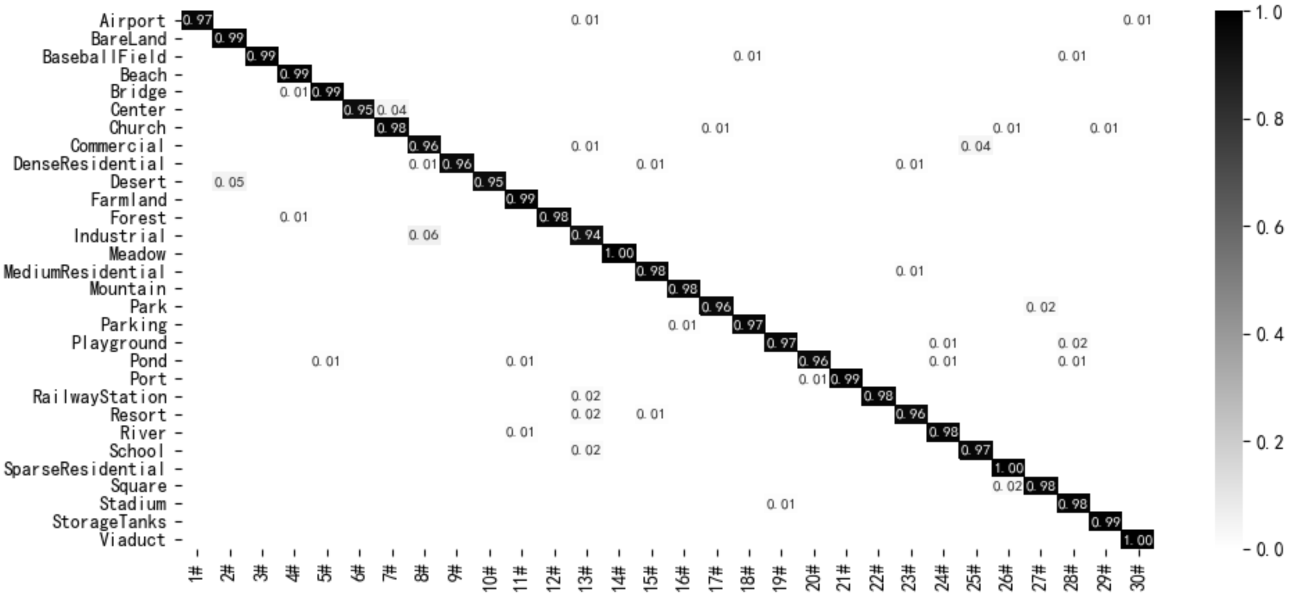

Figure 8. Confusion Matrix for the LCNN-GWHA Method on the AID (50/50) Dataset.

3.3.4. Experimental Results on the NWPU45 Dataset

For the NWPU45 dataset, experiments were carried out with training test = 2:8 and training test = 1:9, respectively. The experimental results are shown in Table 8. We can see that when the training proportion was 10%, the proposed method achieved 92.24% classification accuracy with 0.3 M parameters, which is 2.01% higher than that of LiG with RBF kernel [56] with 2.07 M parameters, 7.91% higher than that of Skip-Connected CNN [43] with 6 M parameters, and 5.71% higher than that of the LCNN-BFF method [38] with 6.2 M parameters. When the training proportion was 20%, the OA of the proposed method was 94.26%, which is 1.01% higher than that of LiG with RBF kernel [56], 2.37% higher than that of Discriminative with VGG16 [44], and 2.53% higher than that of the LCNN-BFF Method [38]. The experimental results demonstrate that the proposed method could extract more significant features with fewer parameters in datasets with rich image changes, as well as high similarity between classes and intra-class differences.

Table 8. Performance Comparison of the Proposed Model with Some Advanced Methods on the NWPU45 Dataset.

Network Model	OA (10/90) (%)	OA (20/80) (%)	Number of Parameters
R.D [51]	-	91.03	-
AlexNet with MSCP [40]	81.70 ± 0.23	85.58 ± 0.16	-
VGG16 with MSCP [40]	85.33 ± 0.17	88.93 ± 0.14	-
VGG_VD16 with SAFF [54]	84.38 ± 0.19	87.86 ± 0.14	15 M
Fine-tuning [30]	87.15 ± 0.45	90.36 ± 0.18	130 M
Skip-Connected CNN [43]	84.33 ± 0.19	87.30 ± 0.23	6 M
LCNN-BFF [38]	86.53 ± 0.15	91.73 ± 0.17	6.2 M
VGG16+CapsNet [39]	85.05 ± 0.13	89.18 ± 0.14	130 M
Discriminative with AlexNet [44]	85.56 ± 0.20	87.24 ± 0.12	130 M
Discriminative with VGG16 [44]	89.22 ± 0.50	91.89 ± 0.22	130 M
ResNet50 [55]	86.23 ± 0.41	88.93 ± 0.12	25.61 M
InceptionV3 [55]	85.46 ± 0.33	87.75 ± 0.43	45.37 M
Contourlet CNN [33]	85.93 ± 0.51	89.57 ± 0.45	12.6 M
LiG with RBF kernel [56]	90.23 ± 0.13	93.25 ± 0.12	2.07 M
Proposed Method	92.24 ± 0.12	94.26 ± 0.25	0.31 M

The confusion matrix for the proposed method on the training test = 2:8 NWPU45 dataset is shown in Figure 9. As the NWPU45 dataset has high intraclass dissimilarity and interclass similarity, none of the classes were completely correctly classified. However, there were 44 scenarios that achieved a classification accuracy of over 90%, and thus achieved good classification results. As shown in Figure 9, the worst accuracy scenarios were 'church' and 'palace' with accuracies of 90% and 89%, respectively, as they had very similar buildings, which caused confusion when classifying. In addition, the classification accuracy of 'roundabout' scenes was lower than 91%, as irregular intersections can easily be identified as 'intersection' scenes, resulting in incorrect classification. Nevertheless, the proposed method still gave good classification results for each scenario.

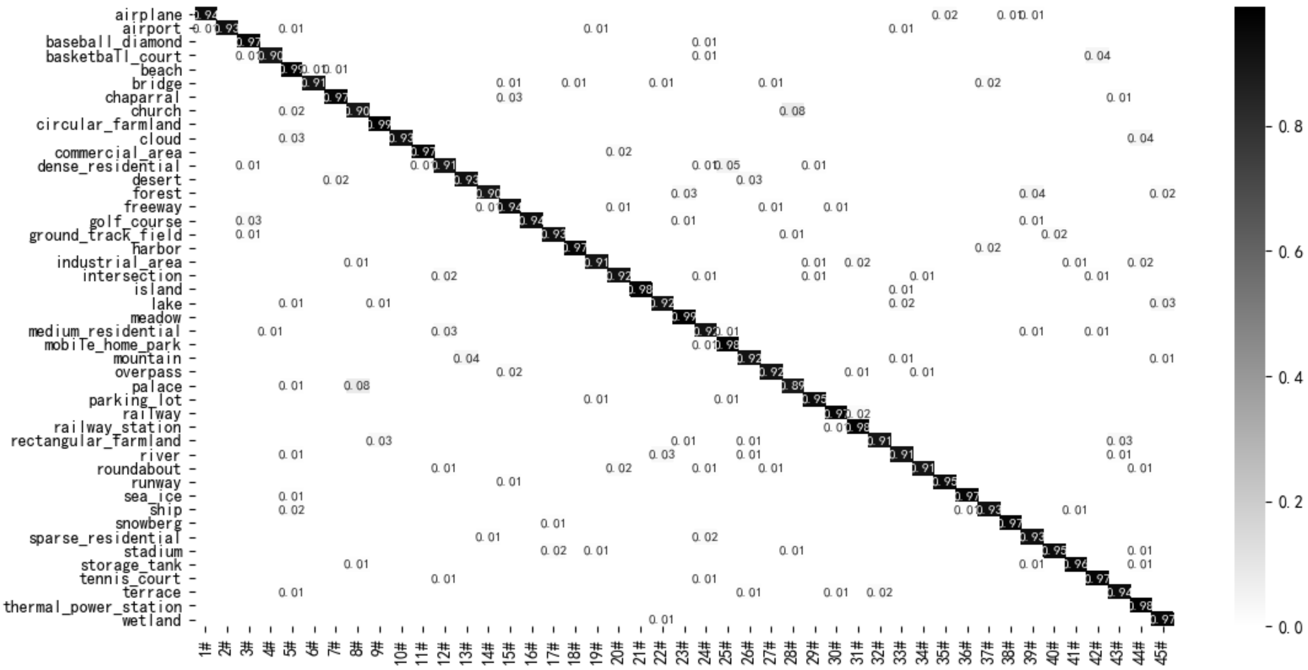

Figure 9. Confusion Matrix for the LCNN-GWHA Method on the NWPU45 (20/80) Dataset.

3.4. Speed Comparison of Models

To verify the advantage of our method in terms of speed, experiments were performed on the UCM21 dataset using the ATT evaluation index. The ATT refers to the average training time required by a model to process an image. Because the results of ATT have a great relationship with the performance of the computer, other algorithms for comparison are rerun on the same computer. In order to reduce random effects, the average value of ten experiments is taken as the final result for each method. The experimental results of our method were compared with those of advanced methods, as detailed in Table 9.

Table 9. ATT Comparison of the Proposed Model with Advanced Methods on UCM21 Datasets.

Network Model	Time Required to Process Each Image(s)
Siamese ResNet_50 [51]	0.053
Siamese AlexNet [51]	0.028
Siamese VGG-16 [51]	0.039
GBNet+global feature [41]	0.052
GBNet [41]	0.048
LCNN-BFF [38]	0.029
Proposed Method	0.010

It can be seen that under the same experimental equipment, the proposed method took 0.010 s to process a remote sensing image, which was the shortest time compared with the other methods. It was 0.018 s faster than that the Siamese ALexNet [51] method and 0.019 s faster than that the LCNN-BFF [38] method.

3.5. Comparison of Computational Complexity of Models

The floating-point operations (FLOPs) evaluation index is used to measure the complexity of models. Experiments were performed on the AID dataset with a training proportion of 50%. The experimental results are shown in Table 10. It can be seen that both the OA and FLOPs of the proposed method were the best compared with those of the methods used for comparison. Compared with the lightweight network models, MobileNetv2 [34] and SE-MDPMNet [34], on the premise that the FLOPs value has great advantages, the classification accuracy of the proposed method was 1.68% and 0.5% higher than these methods, respectively, thus verifying that the proposed LCNN-GWHA method can achieve a good trade-off between classification accuracy and running speed.

Table 10. Complexity Evaluation of some Models.

Network Model	OA (%)	Number of Parameters	FLOPs
CaffeNet [30]	89.53	60.97 M	715 M
VGG-VD-16 [30]	89.64	138.36 M	15.5 G
GoogLeNet [30]	86.39	7 M	1.5 G
MobileNetV2 [34]	95.96	3.5 M	334 M
SE-MDPMNet [34]	97.14	5.17 M	3.27 G
Proposed Method	97.64	0.31 M	12.6 M

4. Discussion

To the performance of the proposed LCNN-GWHA method more intuitively, three visualization methods were explored. The UCM21 dataset was selected for these experiments. First, the channel attention, spatial attention, and mixed attention used in the LCNN-GWHA method were visualized, as shown in Figure 10. In Figure 10, different colors represent the degree of attention to the region, and where the yellow color represents a high degree of attention to the region.

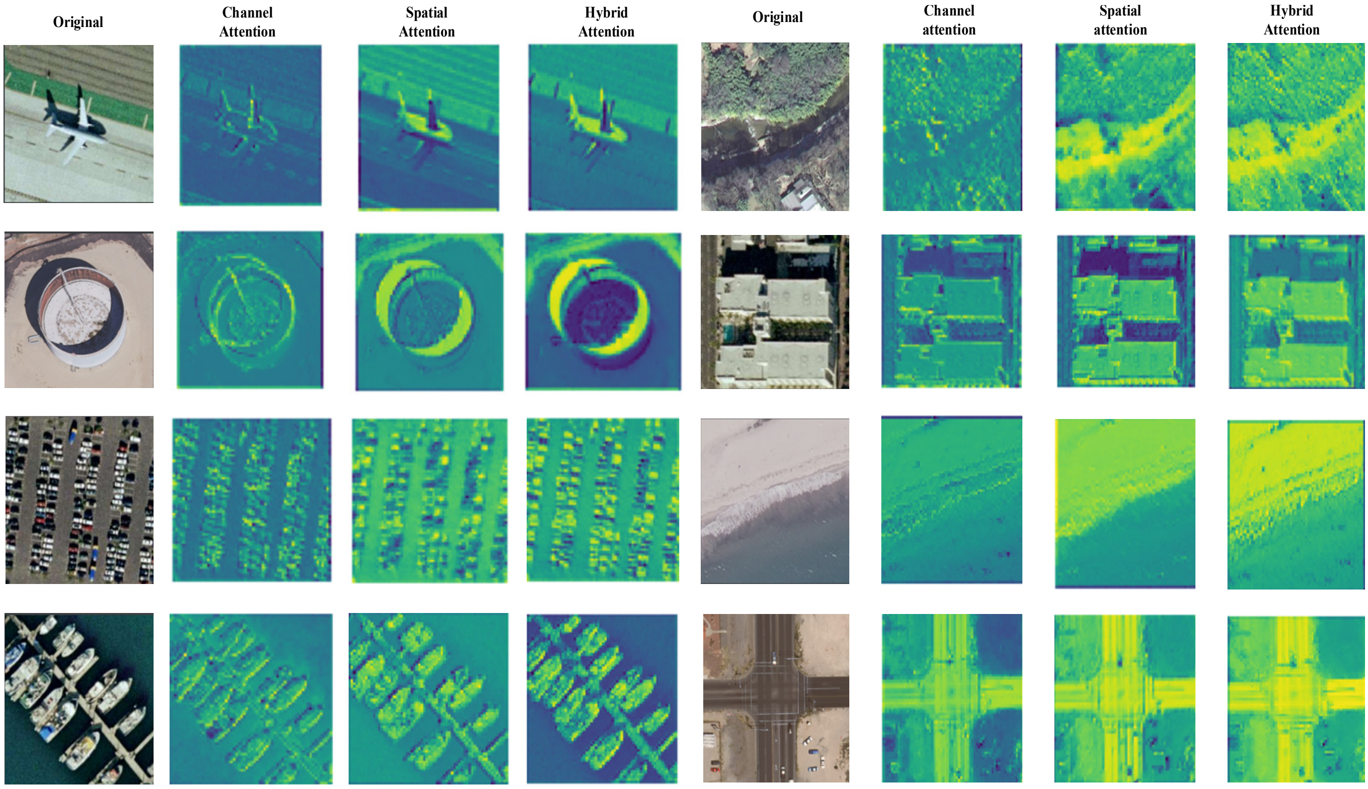

Figure 10. Attention Visualization Results.

From the visualization results in Figure 10, it can be seen that the points of interest under the different attention mechanisms were different, as well as the areas of enhancement. Channel attention enhanced the feature points of interest, while spatial attention enhanced the background area of interest. For hybrid attention, this added spatial attention on the basis of channel attention, which made it more active and allowed learning more meaningful features. Therefore, both the background area and feature points were enhanced at the same time.

Next, class activation map (CAM) visualization was used to visualize the entire network feature extraction ability of the proposed LCNN-GWHA method. This method uses the gradient of any target and then generates a rough attention map from the last layer of the convolution network to show important areas in the image predicted by the model. Some images were randomly selected from the UCM21 dataset for visual analysis. The VGG_VD16 with SAFF method is adopted for CAM visual comparison with the proposed LCNN-GWHA method. The visual comparison results are shown in Figure 11. The LCNN-GWHA method can better cover important objects with a wide range of highlights. Especially for the 'Tenniscourt' scenario, the proposed LCNN-GWHA method perfectly covers the main target, while the coverage area of VGG_VD16 with SAFF method deviates severely, resulting in classification errors. This is because the LCNN-GWHA method proposed has strong target positioning and recognition ability due to the enhancement of features by hybrid attention.

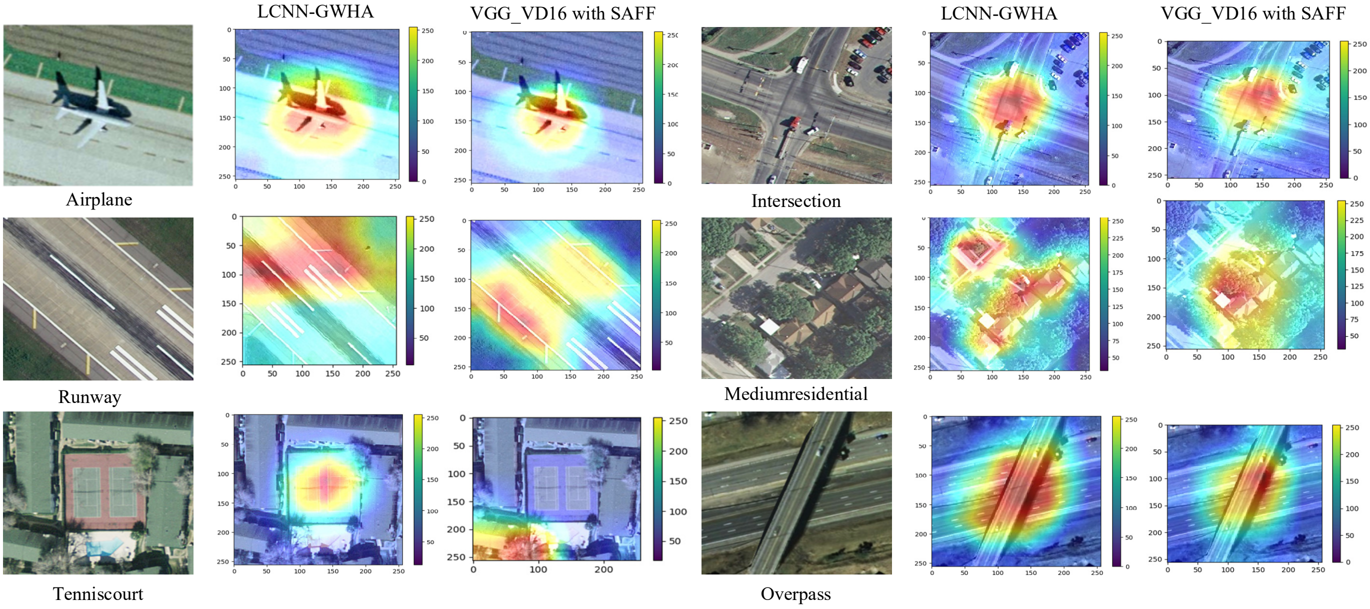

Figure 11. Class activation map (CAM) visualization results of the LCNN-GWHA method and the VGG_VD16 with SAFF method on UCM21 dataset.

In addition, some random classification experiments were conducted on the UCM21 data set to further prove the effectiveness of the proposed LCNN-GWHA method. The experimental results are shown in Figure 12. We can see that the predictive confidence of the model exceeded 99%, and some of the individual cases reached 100%. This proves that the proposed LCNN-GWHA method could extract image features more effectively.

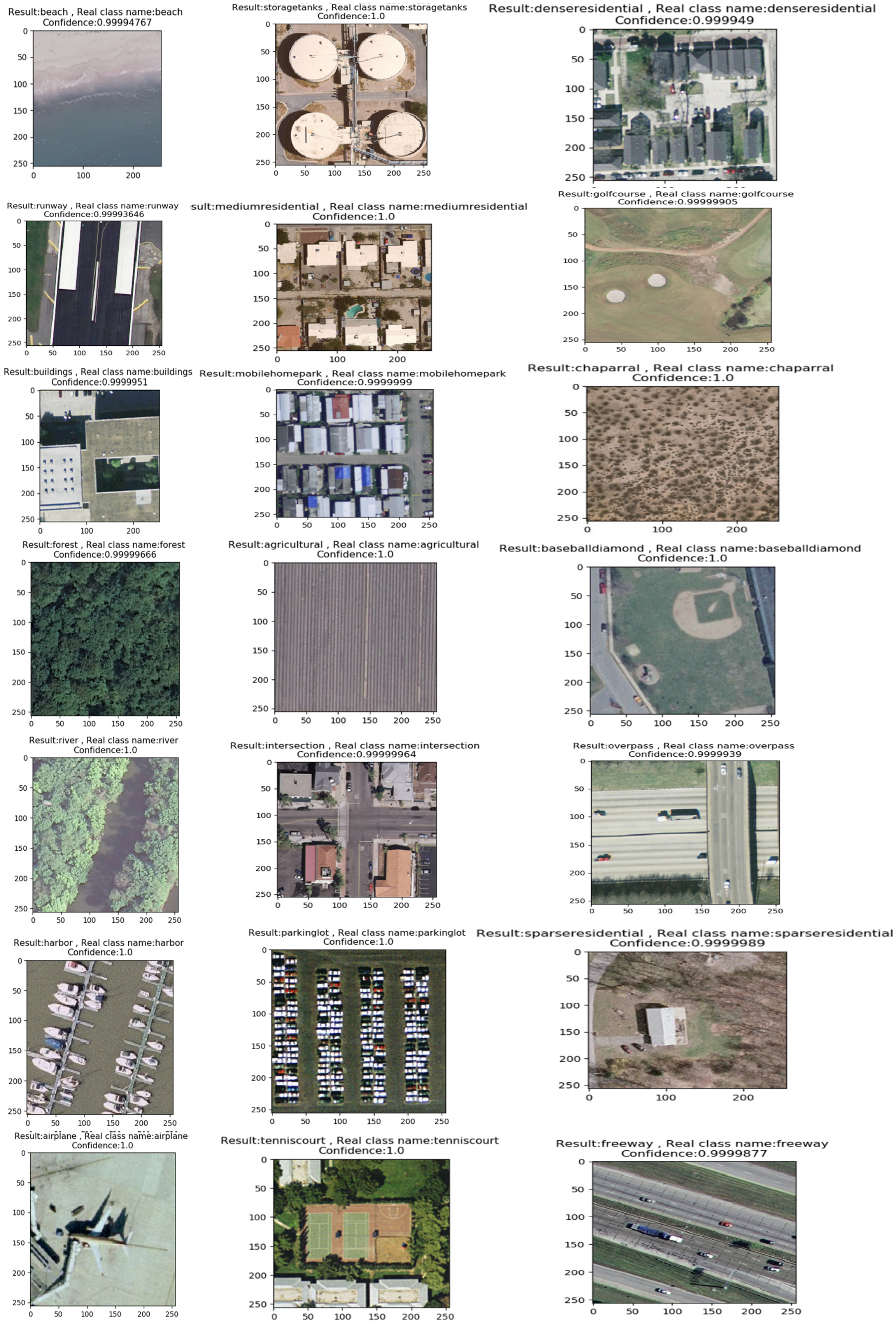
Result:beach , Real class name:beach
Confidence:0.99994767
Result:storagetanks , Real class name:storagetanks
Confidence:1.0
Result:denseresidential , Real class name:denseresidential
Confidence:0.999949
Result:runway , Real class name:runway
Confidence:0.99993646
sult:mediumresidential , Real class name:mediumresidential
Confidence:1.0
Result:golfcourse , Real class name:golfcourse
Confidence:0.99999905
Result:buildings , Real class name:buildings
Confidence:0.9999951
Result:mobilehomepark , Real class name:mobilehomepark
Confidence:0.9999999
Result:chaparral , Real class name:chaparral
Confidence:1.0
Result:forest , Real class name:forest
Confidence:0.99999666
Result:agricultural , Real class name:agricultural
Confidence:1.0
Result:baseballdiamond , Real class name:baseballdiamond
Confidence:1.0
Result:river , Real class name:river
Confidence:1.0
Result:intersection , Real class name:intersection
Confidence:0.99999964
Result:overpass , Real class name:overpass
Confidence:0.9999939
Result:harbor , Real class name:harbor
Confidence:1.0
Result:parkinglot , Real class name:parkinglot
Confidence:1.0
Result:sparseresidential , Real class name:sparseresidential
Confidence:0.9999989
Result:airplane , Real class name:airplane
Confidence:1.0
Result:tenniscourt , Real class name:tenniscourt
Confidence:1.0
Result:freeway , Real class name:freeway
Confidence:0.9999877
0
50
100
150
200
250

Figure 12. Random classification prediction results.

5. Conclusions

In this paper, we present a lightweight end-to-end convolutional neural network for remote sensing scene image classification that combines the advantages of channel attention, spatial attention, and channel grouping. Channel attention is introduced to enhance important features, spatial attention is introduced to enhance the regions of interest, and the two kinds of attention are fused to generate a hybrid attention module with higher activation, which can extract more meaningful features. In order to make the proposed model lightweight, group-wise hybrid attention is proposed, and hybrid attention is introduced into each group, which not only ensures high classification accuracy but also greatly reduces the computational complexity. Experiments were carried out on four data sets with various training ratios. The experimental results demonstrate that the proposed method is robust and has higher classification accuracy than state-of-the-art methods. In addition to spatial attention and channel attention, self -attention is another effective method to improve network performance. In future work, we will propose a more efficient lightweight convolutional neural network for remote sensing scene classification, based on the use of a self-attention mechanism.

Author Contributions: Conceptualization, C.S.; data curation, C.S., X.Z. and J.S.; formal analysis, L.W.; methodology, C.S.; software, X.Z.; validation, C.S., X.Z. and J.S.; writing—original draft, X.Z.; writing—review & editing, C.S. and L.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded in part by the Heilongjiang Science Foundation Project of China under Grant LH2021D022, in part by the National Natural Science Foundation of China (41701479, 62071084), and in part by the Fundamental Research Funds in Heilongjiang Provincial Universities of China under Grant 135509136.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data associated with this research are available online. The UC Merced dataset is available for download at http://weegee.vision.ucmerced.edu/datasets/landuse.html (accessed on 18 November 2021). RSCCN dataset is available for download at https://sites.google.com/site/qinzoucn/documents (accessed on 18 November 2021). NWPU dataset is available for download at http://www.escience.cn/people/JunweiHan/NWPU-RESISC45.html (accessed on 18 November 2021). AID dataset is available for download at https://captain-whu.github.io/AID/ (accessed on 18 November 2021).

Acknowledgments: We would like to thank the handling editor and the anonymous reviewers for their careful reading and helpful remarks.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
2. Toshev, A.; Szegedy, C. DeepPose: Human Pose Estimation via Deep Neural Networks. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1653–1660.
3. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, June 2015; pp. 3431–3440.
4. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 91–99. [CrossRef] [PubMed]
5. Zheng, X.; Chen, X.; Lu, X.; Sun, B. Unsupervised Change Detection by Cross-Resolution Difference Learning. *IEEE Trans. Geosci. Remote. Sens.* **2021**, *18*, 1–16. [CrossRef]
6. Zheng, X.; Wang, B.; Du, X.; Lu, X. Mutual Attention Inception Network for Remote Sensing Visual Question Answering. *IEEE Trans. Geosci. Remote. Sens.* **2021**, *18*, 1–14. [CrossRef]
7. Luo, F.; Zou, Z.; Liu, J.; Lin, Z. Dimensionality reduction and classification of hyperspectral image via multi-structure unified discriminative embedding. *IEEE Trans. Geosci. Remote Sens.* **2021**, *18*, 1. [CrossRef]

8. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
9. Luo, F.; Huang, H.; Ma, Z.; Liu, J. Semi-supervised Sparse Manifold Discriminative Analysis for Feature Extraction of Hyperspectral Images. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 6197–6221. [CrossRef]
10. Luo, F.; Zhang, L.; Zhou, X.; Guo, T.; Cheng, Y.; Yin, T. Sparse-Adaptive Hypergraph Discriminant Analysis for Hyperspectral Image Classification. *IEEE Geosci. Remote Sens. Lett.* **2020**, *17*, 1082–1086. [CrossRef]
11. Zheng, X.; Gong, T.; Li, X.; Lu, X. Generalized Scene Classification From Small-Scale Datasets With Multitask Learning. *IEEE Trans. Geosci. Remote Sens.* **2021**, *18*, 1–11. [CrossRef]
12. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
13. He, K.; Zhang, X.; Ren, S.; Sun, J. Identity Mappings in Deep Residual Networks. In Proceedings of the 14th European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 630–645. [CrossRef]
14. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv* **2015**, arXiv:1502.03167.
15. Carreira, J.; Madeira, H.; Silva, J.G. Xception: A technique for the experimental evaluation of dependability in modern computers. *IEEE Trans. Softw. Eng.* **1998**, *24*, 125–136. [CrossRef]
16. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, M.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
17. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L. MobileNetV2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520. [CrossRef]
18. Xie, S.N.; Girshick, R.; Dollar, P.; Tu, Z.W.; He, K.M. Aggregated Residual Transformations for Deep Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 5987–5995. [CrossRef]
19. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; p. 18326147.
20. Ma, N.; Zhang, X.; Zheng, H.T.; Sun, J. Shufflflenet v2: Practical guidelines for efficient cnn architecture design. *arXiv* **2018**, arXiv:1807.11164.
21. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
22. Li, X.; Wang, W.; Hu, X.; Yang, J. Selective kernel networks. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 510–519.
23. Woo, S.; Park, J.; Lee, J.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the 2018 European Conference on Computer Vision, Munich, Germany, 8–14 September 2018.
24. Wang, Q.; Liu, S.; Chanussot, J.; Li, X. Scene Classification With Recurrent Attention of VHR Remote Sensing Images. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 1155–1167. [CrossRef]
25. Tong, W.; Chen, W.; Han, W.; Li, X.; Wang, L. Channel-Attention-Based DenseNet Network for Remote Sensing Image Scene Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 4121–4132. [CrossRef]
26. Yu, D.; Guo, H.; Xu, Q.; Lu, J.; Zhao, C.; Lin, Y. Hierarchical Attention and Bilinear Fusion for Remote Sensing Image Scene Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 6372–6383. [CrossRef]
27. Alhichri, H.; Alswayed, A.S.; Bazi, Y.; Ammour, N.; Alajlan, N.A. Classification of Remote Sensing Images Using EfficientNet-B3 CNN Model With Attention. *IEEE Access* **2021**, *9*, 14078–14094. [CrossRef]
28. Yang, Y.; Newsam, S. Bag-of-visual-words and spatial extensions for land-use classification. In Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, San Jose, CA, USA, 3–5 November 2010; pp. 270–279.
29. Zou, Q.; Ni, L.; Zhang, T.; Wang, Q. Deep Learning Based Feature Selection for Remote Sensing Scene Classification. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 2321–2325. [CrossRef]
30. Xia, G.-S.; Hu, J.; Hu, F.; Shi, B.; Bai, X.; Zhong, Y.; Zhang, L.; Lu, X. AID: A Benchmark Data Set for Performance Evaluation of Aerial Scene Classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3965–3981. [CrossRef]
31. Cheng, G.; Han, J.; Lu, X. Remote Sensing Image Scene Classification: Benchmark and State of the Art. *Proc. IEEE.* **2017**, *105*, 1865–1883. [CrossRef]
32. Li, B.; Su, W.; Wu, H.; Li, R.; Zhang, W.; Qin, W.; Zhang, S. Aggregated Deep Fisher Feature for VHR Remote Sensing Scene Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2019**, *12*, 3508–3523. [CrossRef]
33. Liu, M.; Jiao, L.; Liu, X.; Li, L.; Liu, F.; Yang, S. C-CNN: Contourlet Convolutional Neural Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 2636–2649. [CrossRef] [PubMed]
34. Zhang, B.; Zhang, Y.; Wang, S. A Lightweight and Discriminative Model for Remote Sensing Scene Classification With Multidilation Pooling Module. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2019**, *12*, 2636–2653. [CrossRef]
35. Zhao, F.; Mu, X.; Yang, Z.; Yi, Z. A novel two-stage scene classification model based on feature variable significance in high-resolution remote sensing. *Geocarto Int.* **2019**, *35*, 1603–1614. [CrossRef]

36. Liu, Y.; Liu, Y.; Ding, L. Scene classification based on two-stage deep feature fusion. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 183–186. [CrossRef]
37. Liu, B.-D.; Meng, J.; Xie, W.-Y.; Shao, S.; Li, Y.; Wang, Y. Weighted Spatial Pyramid Matching Collaborative Representation for Remote-Sensing-Image Scene Classification. *Remote Sens.* **2019**, *11*, 518. [CrossRef]
38. Shi, C.; Wang, T.; Wang, L. Branch Feature Fusion Convolution Network for Remote Sensing Scene Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 5194–5210. [CrossRef]
39. Zhang, W.; Tang, P.; Zhao, L. Remote sensing image scene classification using CNN-CapsNet. *Remote Sens.* **2019**, *11*, 494. [CrossRef]
40. He, N.; Fang, L.; Li, S.; Plaza, A.; Plaza, J. Remote Sensing Scene Classification Using Multilayer Stacked Covariance Pooling. *IEEE Trans. Geosci. Remote. Sens.* **2018**, *56*, 6899–6910. [CrossRef]
41. Sun, H.; Li, S.; Zheng, X.; Lu, X. Remote Sensing Scene Classification by Gated Bidirectional Network. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 82–96. [CrossRef]
42. Lu, X.; Sun, H.; Zheng, X. A Feature Aggregation Convolutional Neural Network for Remote Sensing Scene Classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 7894–7906. [CrossRef]
43. He, N.; Fang, L.; Li, S.; Plaza, J.; Plaza, A. Skip-Connected Covariance Network for Remote Sensing Scene Classification. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *31*, 1461–1474. [CrossRef] [PubMed]
44. Cheng, G.; Yang, C.; Yao, X.; Guo, L.; Han, J. When Deep Learning Meets Metric Learning: Remote Sensing Image Scene Classification via Learning Discriminative CNNs. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 2811–2821. [CrossRef]
45. Boualleg, Y.; Farah, M.; Farah, I.R. Remote Sensing Scene Classification Using Convolutional Features and Deep Forest Classifier. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 1944–1948. [CrossRef]
46. Xie, J.; He, N.; Fang, L.; Plaza, A. Scale-Free Convolutional Neural Network for Remote Sensing Scene Classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 6916–6928. [CrossRef]
47. Li, J.; Lin, D.; Wang, Y.; Xu, G.; Zhang, Y.; Ding, C.; Zhou, Y. Deep Discriminative Representation Learning with Attention Map for Scene Classification. *Remote Sens.* **2020**, *12*, 1366. [CrossRef]
48. Yan, P.; He, F.; Yang, Y.; Hu, F. Semi-Supervised Representation Learning for Remote Sensing Image Classification Based on Generative Adversarial Networks. *IEEE Access* **2020**, *8*, 54135–54144. [CrossRef]
49. Wang, C.; Lin, W.; Tang, P. Multiple resolution block feature for remote-sensing scene classification. *Int. J. Remote Sens.* **2019**, *40*, 6884–6904. [CrossRef]
50. Liu, X.; Zhou, Y.; Zhao, J.; Yao, R.; Liu, B.; Zheng, Y. Siamese Convolutional Neural Networks for Remote Sensing Scene Classification. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 1200–1204. [CrossRef]
51. Zhou, Y.; Liu, X.; Zhao, J.; Ma, D.; Yao, R.; Liu, B.; Zheng, Y. Remote sensing scene classification based on rotation-invariant feature learning and joint decision making. *EURASIP J. Image Video Process.* **2019**, *2019*, 3. [CrossRef]
52. Lu, X.; Ji, W.; Li, X.; Zheng, X. Bidirectional adaptive feature fusion for remote sensing scene classification. *Neurocomputing* **2019**, *328*, 135–146. [CrossRef]
53. Liu, Y.; Zhong, Y.; Qin, Q. Scene Classification Based on Multiscale Convolutional Neural Network. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 7109–7121. [CrossRef]
54. Cao, R.; Fang, L.; Lu, T.; He, N. Self-Attention-Based Deep Feature Fusion for Remote Sensing Scene Classification. *IEEE Geosci. Remote Sens. Lett.* **2021**, *18*, 43–47. [CrossRef]
55. Li, W.; Wang, Z.; Wang, Y.; Wu, J.; Wang, J.; Jia, Y.; Gui, G. Classification of High-Spatial-Resolution Remote Sensing Scenes Method Using Transfer Learning and Deep Convolutional Neural Network. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 1986–1995. [CrossRef]
56. Xu, C.; Zhu, G.; Shu, J. A Lightweight Intrinsic Mean for Remote Sensing Classification With Lie Group Kernel Function. *IEEE Geosci. Remote Sens. Lett.* **2021**, *18*, 1741–1745. [CrossRef]