remote sensing

MDPI

Article

A Remote-Sensing Scene-Image Classification Method Based on Deep Multiple-Instance Learning with a Residual Dense Attention ConvNet

Xinyu Wang [1,2], Haixia Xu [1,2], Liming Yuan [1,2], Wei Dai [1,2] and Xianbin Wen [1,2]*

[1] School of Computer Science and Engineering, and Key Laboratory of Computer Vision, Tianjin University of Technology, Tianjin 300384, China
[2] System of the Ministry of Education, Tianjin 300384, China
* Correspondence: xbwen@email.tjut.edu.cn

Abstract: The spatial distribution of remote-sensing scene images is highly complex in character, so how to extract local key semantic information and discriminative features is the key to making it possible to classify accurately. However, most of the existing convolutional neural network (CNN) models tend to have global feature representations and lose the shallow features. In addition, when the network is too deep, gradient disappearance and overfitting tend to occur. To solve these problems, a lightweight, multi-instance CNN model for remote sensing scene classification is proposed in this paper: MILRDA. In the instance extraction and classifier part, more discriminative features are extracted by the constructed residual dense attention block (RDAB) while retaining shallow features. Then, the extracted features are transformed into instance-level vectors and the local information associated with bag-level labels is highlighted by the proposed channel-attention-based multi-instance pooling, while suppressing the weights of useless objects or backgrounds. Finally, the network is constrained by the cross-entropy loss function to output the final prediction results. The experimental results on four public datasets show that our proposed method can achieve comparable results to other state-of-the-art methods. Moreover, the visualization of feature maps shows that MILRDA can find more effective features.

Keywords: remote-sensing scene image classification; convolutional neural network (CNN); multiple instance learning (MIL); attention mechanisms

Citation: Wang, X.; Xu, H.; Yuan, L.; Dai, W.; Wen, X. A Remote-Sensing Scene-Image Classification Method Based on Deep Multiple-Instance Learning with a Residual Dense Attention ConvNet. *Remote Sens.* **2022**, *14*, 5095. https://doi.org/10.3390/rs14205095

Academic Editor: Giuseppe Scarpa

Received: 24 August 2022
Accepted: 3 October 2022
Published: 12 October 2022

1. Introduction

High-resolution remote-sensing scene-image classification, as a fundamental task in remote-sensing-image understanding, has received increasing attention in the past few years [1–3]. Thanks to the development of satellite and remote-sensing technologies, remote-sensing scene-image classification plays an important role in real-life applications, such as urban construction and planning [4,5], land cover and land use (LCLU) [6], vegetation mapping [7], remote monitoring and intelligent decision making [8,9].

Difficulties in the study of remote-sensing scene-image classification are influenced by the characteristics of the images themselves. Compared to natural images, remote-sensing scene images contain more objects at different scales because they are taken from a bird's eye view. As shown in Figure 1a, remote-sensing images often contain many objects which are diverse in size. As shown in Figure 1b, objects in natural images are usually medium-sized and centered, but in remote-sensing images, objects generally have multiple scales and show a dense distribution at any location in the image. In addition, remote-sensing images often contain a large amount of useless background information due to the angle of imaging, making it more difficult to capture key object features than natural images. In addition, the remote-sensing scene images contain a large number of similar scenes, and have the characteristics of small inter-class differences and large intra-class differences.

As shown in Figure 2, the categories in (a) are the same, but their architectural styles are obviously different, and the shapes of the main objects are not similar; the categories in (b) are different but very similar, such as "runway," "freeway," "railway" and "intersection," which are sometimes difficult to distinguish with the naked eye. Another example involves "terrace," "meadow," "wetland" and "forest," which mostly contain similar or identical objects, but they have completely different semantic labels. The above problems pose difficulties for accurate classification of remote-sensing scene images.

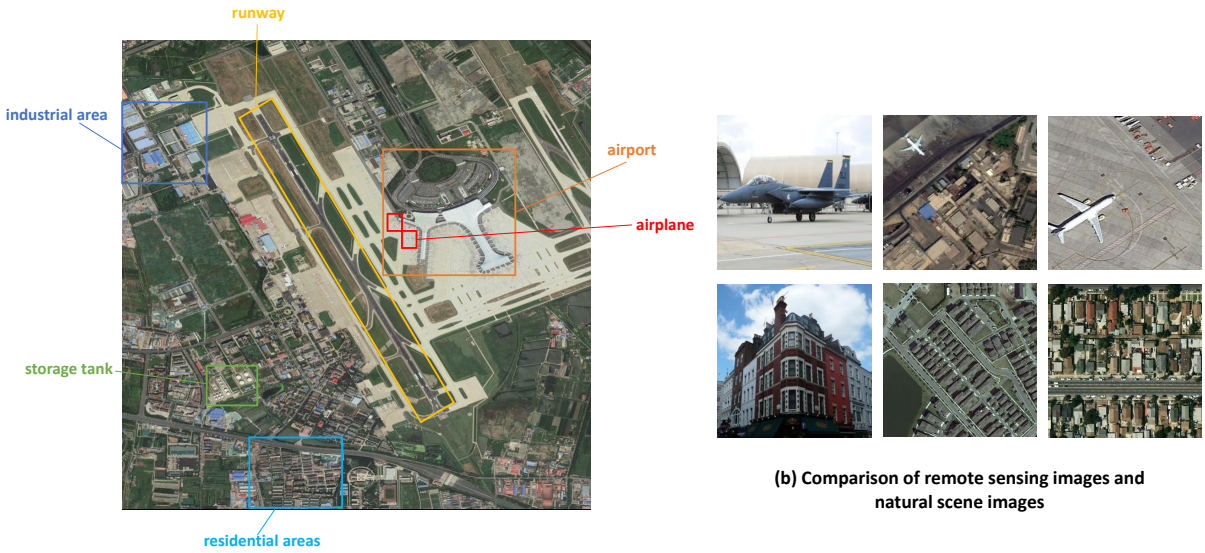

Figure 1. Comparison of remote-sensing images and natural images.

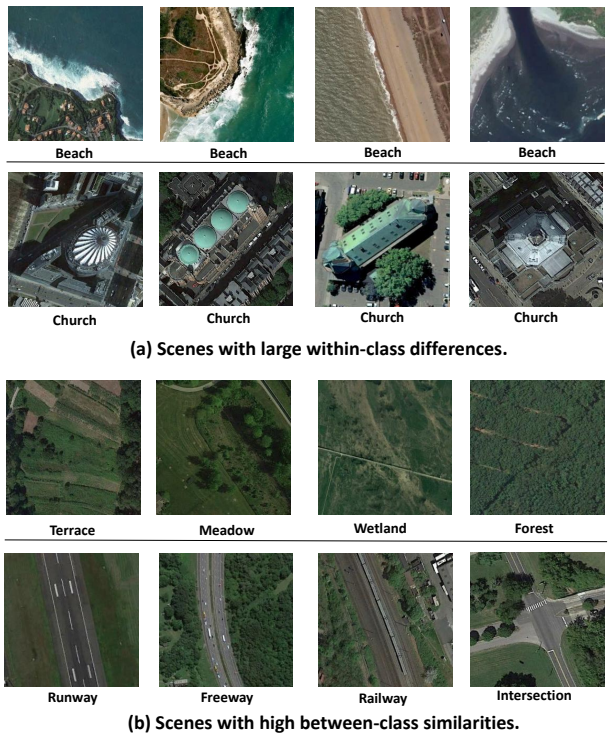

Figure 2. Examples of different classes in remote-sensing scene images.

The early research methods are mainly based on low-level features, which represent the features of remote-sensing scene images by selecting different feature description operators.

Some of the widely used methods include SIFT [10], CH [11], HOG [12], GIST [13] and LBP [14]. However, with the rapid development of remote-sensing imaging technologies and platforms, the internal information contained in remote-sensing images is becoming more and more complex, and a single shallow feature is no longer applicable. To overcome the limitations of low-level feature description, the researchers proposed a method based on mid-level features. This type of approach obtains global features by encoding the extracted features (such as BoVW [15], LDA [16] and pLSA [17]). However, these methods rely on a large amount of a priori information and sparse local features, and thus have limited ability to characterize remote sense images.

With the rapid development of deep learning methods (high-level features) since 2012, deep CNN models have become able to automatically learn and extract representative features from given data, and have achieved many impressive results in several fields [18]. Compared with traditional feature extraction methods, deep learning methods have stronger recognition and feature description capabilities [19]. Remote-sensing scene-image classification methods based on deep learning can be broadly classified into three categories, namely, fine tuning, full training and using a CNN as a feature extractor. The fine-tuning-based methods usually target CNN models pre-trained on large natural-image datasets beforehand, and use remote-sensing image datasets. CNN models require a large amount of data for training to reach their true potential, but remote-sensing datasets have the problem of small samples, so fine-tuning-based methods are generally effective. Full-training-based methods usually redesign the CNN structure based on remote-sensing scene-image features or improve the currently available superior models [20–22]. The new model can extract key features directly from remote-sensing scene images, and thus works better than existing models such as VGGNet [23], AlexNet [24] and ResNet [25]. Using a CNN as a feature extractor usually fuses multiple layers of features from a CNN model to obtain a more comprehensive feature representation map [26–29]. Although such methods outperform existing CNN models, they require CNN models that have already been trained on remote sensing datasets, and thus they lack flexibility.

Although the CNN has achieved some good results in the study of remote-sensing scene-image classification, the following problems still exist:

(1) Insufficient description of key semantic feature representations: The remote-sensing scene-image contains many objects or redundant backgrounds inside the image that are not related to labels, and also has the characteristics of large intra-class differences and small inter-class differences, but the CNN focuses on global features, which are easily disturbed by useless information and affect the final performance.

(2) Too many parameters make it difficult to train: Although the deeper CNNs have stronger feature representation capabilities, the small sizes of remote-sensing scene-image datasets tend to cause parameter redundancy, resulting in low accuracy. Meanwhile, the problem of gradient disappearance easily arises during the training process, which generates a high computational cost.

(3) Loss of shallow features: Although the discriminative power of deep features is stronger, retaining shallow features is more helpful to enriching the diversity of features. For remote-sensing scene images with complex spatial information, retaining shallow features is more helpful to describing different spatial structures and improving the final classification performance.

In recent years, multiple-instance learning (MIL) is often combined with a CNN. This combined approach can effectively distinguish the local semantic information associated with the scene labels [30]. MIL was originally designed for drug activity prediction [31]. Its effectiveness has since also been demonstrated in a range of computer-vision tasks, such as image recognition [32], saliency detection [33] and target detection [34]. In MIL, training samples are specified as bags, each containing multiple instances, each with a predefined semantic label. A bag is labeled as a positive bag if it contains at least one positive instance, and vice versa. In general, there are no specific instance labels, and each instance can only be judged to belong to or be deployed in one bag category [35], which makes MIL well suited

for learning from weakly labeled data [36,37]. In the past few years, the combination of MIL and trainable CNNs has become a new trend. For example, Wang et al. [36] used max pooling and mean pooling to aggregate instance representations in the network. However, this method is applicable to medical images or natural images, and does not adapt to create remote-sensing scene maps that contain complex spatial information.

To solve the above problems, this paper proposes a framework for remote-sensing scene-image classification based on the CNN and MIL. The main objectives include the following.

(1) Improved utilization of shallow features: Deep CNNs usually cannot retain shallow features, but shallow features help to improve feature diversity and enhance the performance of the final classification decision. Therefore, our model should effectively improve the feature reuse rate, improve feature propagation and make full use of the limited samples of remote-sensing scene images.

(2) Enhance the extraction of key features: The commonly used deep CNN models are inadequate in key local feature extraction. Since remote-sensing scene images contain a large amount of redundant background information and have high inter-class similarity, our model needs to improve the feature representation of key objects.

(3) Improved parameter utilization: Although increasing the depth of the CNN model helps to extract deeper features with more discriminative rows, it can easily cause parameter redundancy and overfitting. Therefore, our model should minimize the parameters while ensuring the feature extraction capability of the image-depth semantic information.

In summary, we first constructed a feature extraction module, RDAB, based on local residuals and dense connectivity, and converted the extracted features into local instance vectors. Then, the correlation weights were generated by aggregating the instance information through MIL pooling based on channel attention. Finally, the whole network is constrained by a cross-entropy loss function, so that the whole model outputs the final result directly under the supervision of bag-level labels.

The main contributions of this paper are as follows.

(1) We constructed an end-to-end lightweight network, MILRDA, for remote-sensing scene-image classification. Additionally, it has much smaller parameters and computational complexity compared to existing CNN models.

(2) We constructed the feature extraction module RDAB with local residuals and dense connections, which performs feature reuse and retains shallow features, which helps the network generate more discriminative information.

(3) The constructed MIL pooling based on channel attention and aggregating relevant instances, helps to suppress redundant background information of remote-sensing scene images while highlighting major instance weights and outputting prediction results directly under the supervision of bag-level labeling.

The rest of the article is organized as follows. Section 2 introduces our proposed framework and describes its component parts in detail. Section 3 describes the experimental results and compares them with those of other methods. In Section 4, we discuss the proposed approach. Section 5 summarizes the proposed method.

2. Methodology

Figure 3 shows the architecture of the proposed MILRDA method. MILRDA consists of three parts: (1) instance extraction and classifier, (2) MIL pooling and (3) a bag-Level classification layer. In this framework, we first extract features with the proposed convolution module and then feed the extracted features into the instance-level classifier to obtain instance-level feature vectors. The instance-level classifier here is made up of a series of 1×1 convolutions that are proportional in number to the number of remote-sensing scene images (for example, the UCM dataset corresponds to 21 convolutional groups, and the AID dataset corresponds to 30). Then, we use the proposed MIL pooling with channel attention to obtain the bag-level class probabilities. Finally, the true labels of the scene images are predicted by the softmax classifier. The network as a whole creates an end-to-end structure.

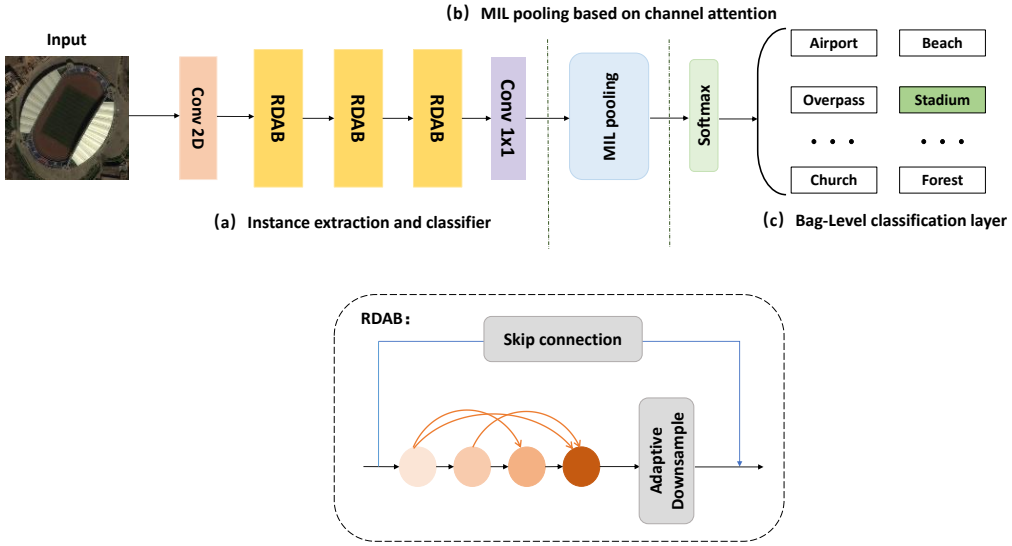

Figure 3. Overall structure of the proposed MILRDA model. The model consists of three parts: (**a**) instance extraction and classifer, (**b**) MIL pooling based on channel attention and (**c**) bag-level classification layer.

For remote-sensing scene classification tasks, each image in a training set T is converted into a collection of local patches, which are referred to as instances. Let x_i denote each local patch that maps to the class label $y_i = h(x_i)$ of instance Z_i through the instance-level classifier h; each instance Z_i is a local piece of image X_i. Then, the instance label is changed into an image (bag) label under the common MIL assumptions, which are based on the MIL pooling function f_{MIL}, denoted as:

$$Y_j = f_{MIL}(y_1, y_2, \ldots, y_n) = \begin{cases} 1, & \text{if } \exists y_i = 1 \\ 0, & \text{else} \end{cases} \tag{1}$$

This indicates that the negative image has solely negative patches, whereas the positive image has at least one positive patch. Since the instance-level label y_i is an unknown hidden variable during the training process, it is crucial to establish the image-to-instance mapping h, and to establish the f_{MIL} that transforms the instance label to the bag label. Deep convolutional neural networks have shown powerful capabilities in the field of computer vision, and we constructed a deep CNN to learn hidden variables, and the pooling function f_{MIL} is a module based on channel attention to better highlight local key regions of images under class label supervision.

2.1. Instance Extraction and Classifier

Scene classification performance is somewhat impacted by the influence of feature extraction. Stronger feature representation may be attained with deeper CNN structures; however, these structures also come with issues, including gradient disappearance, parameter redundancy and challenging training [38]. We built a residual dense attention block (RDAB) for feature extraction to solve this problem and transformed it into an instance feature vector. The complete structure of the block is shown in Figure 4. It consists of a dense connection layer, an attention-based adaptive downsampling layer and local residual connection.

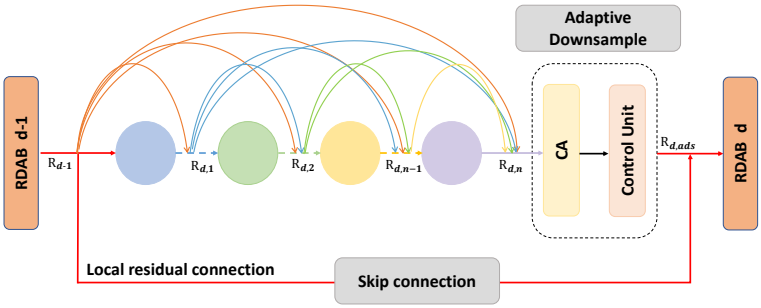

Figure 4. Structure of RDAB. The RDAB module consists of a dense connection block, a skip connection and an adaptive downsampling layer.

(1) Dense Connection Layer

It is known that deep neural networks can be optimized with dense connections for more efficient feature extraction [39]. When training deep neural networks, a large number of trainable parameters are often required, but the small data size of remote-sensing-image datasets makes it difficult to train the networks effectively. The densely connected structure provides feature reuse, which to some extent mitigates the remote sensing datasets' limited sample learning difficulty and boosts training effectiveness. The output feature maps from each layer during feature extraction can be used as inputs for all succeeding layers. We set the number of densely connected layers in the three RDABs to four for multi-level feature representation, which makes the network structure more organized. The dense connection layers consist of 1×1 and 3×3 convolution operations; let $\mathbf{R}_{d-1}$ be the input of the d-th RDAB, and $\mathbf{R}_{d,n}$ stands for the output of the n-th dense connection layer in the d-th RDAB. The whole process of dense connection can be expressed as:

$$\mathbf{R}_{d,n} = H([\mathbf{R}_{d-1}, \mathbf{R}_{d,1}, \ldots, \mathbf{R}_{d,n-1}]) \quad (2)$$

where H denotes the convolution; BN, ReLU three consecutive composite functions; $[\mathbf{B}_{k-1}, \mathbf{B}_{k,1}, \ldots, \mathbf{B}_{k,d-1}]$ denotes the successive operations of the feature map generated by the $d-1$th RDAB. The output feature map channel of $\mathbf{R}_{d,n}$ is $N_0 + N \times (n-1)$, where N_0 is the input feature map channel of $\mathbf{R}_{d-1}$, and N is the growth rate of each dense connection layer.

(2) Attention-Based Adaptive Downsampling

The final output number of feature channels, after the features have passed through numerous tightly coupled layers, is the total of the earlier channels. To ease the network training burden and improve the features while drawing attention to the weights of important regions, we created an attention-based control unit called the adaptive downsampler. The original control unit (CU) consists of 1×1 convolution and average pooling [39]. We placed a coordinate attention (CA) at the front end to highlight key discriminative features while reducing the number of feature channels and improving the efficiency of sampling. CA is a light and high-efficiency attention mechanism that embeds location information into the channel [40]. Compared with the original channel attention mechanism, CA allows lightweight CNNs to acquire critical information at a larger scale. Referring to the experimental procedure of CA, this mechanism is introduced into the constructed residual densely connected module in this paper. CA generates attention weights by encoding channel information in horizontal and vertical coordinates, which are then aggregated. The complete structure is shown in Figure 5, which contains two parts: coordinate information embedding (CIE) and coordinate attention generation (CAG).

CIE: Encoding using the global pooling makes it difficult to retain location information [41]. Therefore, the global average pooling is first decomposed into a bi-directional average set of channels, and the association between long distances is obtained by location information. The outputs of the c-th channel with height h and width w are expressed, respectively, as:

$$z_c^h(h) = \frac{1}{W} \sum_{0 \le i \le W} f_c(h, i) \tag{3}$$

$$z_c^w(w) = \frac{1}{H} \sum_{0 \le j \le H} f_c(j, w) \tag{4}$$

where W and H denote the width and height of the feature map F, and f_c represents the pixel value of c-th channel in the feature map. This operation compresses all pixels of each channel into a single feature vector that gets long dependencies along both directions, helping the network to better capture important information.

CAG: The features obtained in both directions are concatenated and then channel compressed by a shared 1×1 convolutional layer:

$$z = \delta\left(Conv_{1\times1}\left(\left[z^h, z^w\right]\right)\right) \tag{5}$$

where $[.., ..]$ represents the concatenation operation for manipulating the spatial dimension, and δ represents the non-linear and BatchNorm, which encode spatial information in both horizontal and vertical directions. The resulting tensor is then split into its component pieces, z^h and z^w, and its dimensionality is changed using convolution operation, yielding

$$Z^h = \sigma\left(Conv_h\left(z^h\right)\right) \tag{6}$$

$$Z^w = \sigma(Conv_w(z^w)) \tag{7}$$

where $Conv_h$ and $Conv_w$ represent two 1×1 convolutional operations that convert z^h and z^w into a tensor with the same number of channels as the input features. σ is the sigmoid activation function. The outputs Z^h and Z^w are re-weighted and fused as attention weights with the original input features to get $F^{'}$, which can be represented as:

$$F_c^{'}(i, j) = f_c(i, j) \times Z_c^h(i) \times Z_c^w(j) \tag{8}$$

where $f_c(i, j)$ denotes the c-th channel of the input feature map. In the H and W directions, $Z_c^h(i)$ and $Z_c^w(j)$ are the attention weights for the i-th and j-th positions. The output flow of the whole adaptive downsample can be expressed as:

$$\mathbf{R}_{d,\,\mathrm{ads}} = W([\mathbf{R}_{d-1}, \mathbf{R}_{k,1}, \ldots, \mathbf{R}_{d,n-1}, \ldots, \mathbf{R}_{d,n}]) \tag{9}$$

where $\mathbf{R}_{d,\,\mathrm{ads}}$ is the output of adaptive downsample, and W is the operation of CA and CU.

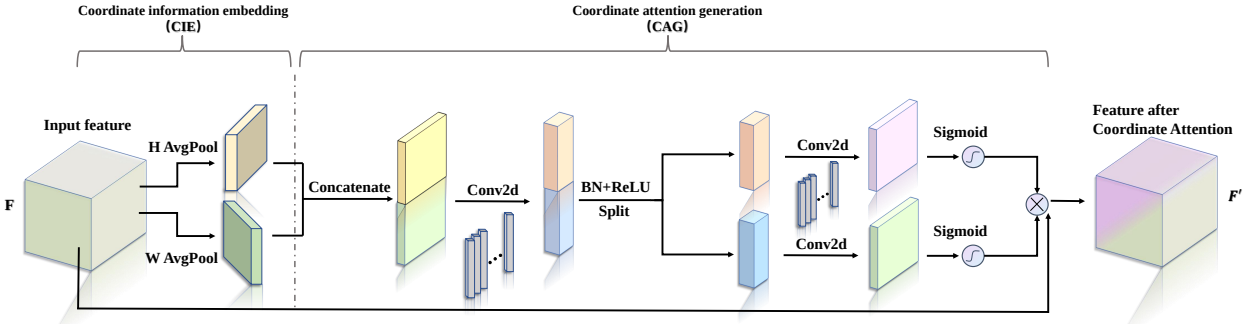

Figure 5. The architecture of the coordinate attention (CA). It contains two parts, coordinate information embedding (CIE) and coordinate attention generation (CAG).

(3) Local Residual Connection

To further ensure that the feature information transmitted by RDAB is not lost and to improve the efficiency and use of the transferred features, driven by the idea of RDN [42], a local residual connection is added between the RDAB input and output. This skip

connection technique can address the issue of gradient disappearance in deep networks and achieve the fusion of local features of densely connected blocks, which to some extent increases the variety of features. The 1×1 convolution is utilized for the local residual connection in order to preserve the consistency of the RDAB input and output dimensions, and the output R_d of the d-th RDAB can be written as follows:

$$\mathbf{R}_d = \mathbf{R}_{d-1} + \mathbf{R}_{d,\,\mathrm{ads}} \tag{10}$$

The output of RDAB can connect with all preceding layers and directly access the original input features, which not only increases feature reuse but also creates implicit deep supervision.

(4) Instance-level Classifier

For remote-sensing scene-image classification tasks, when MIL is introduced, an instance-level classifier needs to be built to sample the local image patches [43]. Specifically, in MILRDA, the image is obtained as a multi-channel feature map after a series of convolutional-feature-extraction operations. Each position on the feature map corresponds to a local feature vector. The feature maps are fed into a 1×1 convolution layer consistent with the number of scenes needed to build an instance-level classifier. Additionally, bag-level semantic labels can be given to the local instances in each feature map.

2.2. MIL Pooling Based on Channel Attention

The MIL pooling converts the instance feature vector to a label at the bag level. The remote-sensed scene images contain many occurrences unrelated to the bag level labels and are available in various sizes. In other words, the instances in the feature map can cover one or more categories (channels). Therefore, there is nonlinear dependence between the different channels. To solve this problem, we constructed MIL pooling based on channel attention [41] that combines CNN and MIL to suppress irrelevant instances while highlighting important regions. The module structure is shown in Figure 6.

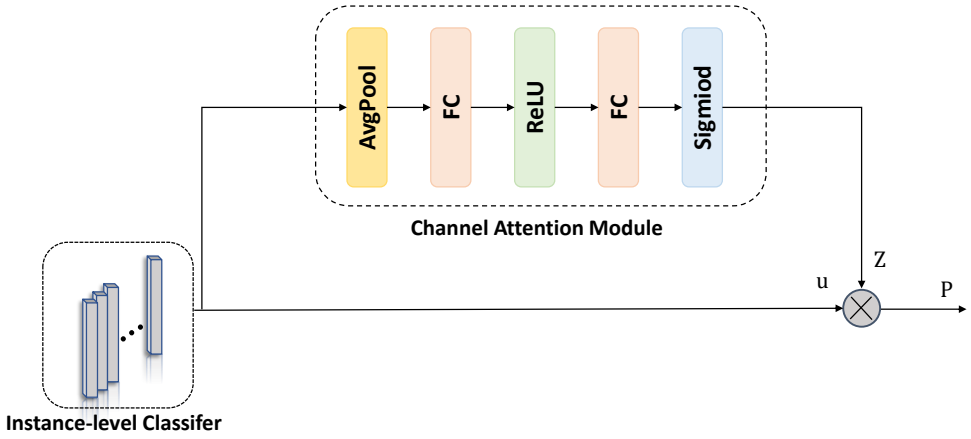

Figure 6. Structure of MIL pooling based on channel attention.

The MIL pooling based on channel attention first initializes channel weights by global average pooling of u:

$$Z_c = \frac{1}{k_1 \times k_2} \sum_{i=1}^{k_1 \times k_2} u_{c,i} \tag{11}$$

Then , the nonlinearity between different channels is captured by two fully connected layers:

$$S_c = \sigma\left(Fc^{(2)}\delta\left(Fc^{(1)}Z\right)\right) \tag{12}$$

where σ is the sigmod function and *delta* is the ReLU function to limit the range of instance weights. There is a skip connection between the output of the input and the sigmoid layer. The final outputs, feature maps u'_c, are obtained by channelwise multiplication:

$$u'_c = Z_c \cdot u_c + u_c \tag{13}$$

After obtaining the instance weights, the class-score vector P_c at the bag level is calculated by weighted average. Each channel of P represents an image class. The probability of the input image belonging to class c is P_c.

$$P_c = \frac{\exp(u_c\prime)}{\sum_{i=1}^{p} \exp(u_i\prime)} \tag{14}$$

2.3. Bag-Level Classification

The softmax classifier receives the output bag-level scores and converts them into conditional probabilities for each class. Then, we calculate the loss between the bag level probability P_c and the true label Y_c by the cross-entropy loss function:

$$Loss = -\sum_{c=1}^{p} Y_c \log P_c \tag{15}$$

Here, the loss between the bag-level prediction and the true label is obtained by direct minimization of the global optimization [43].

3. Experiment and Results

We experimented with the proposed method on four challenging datasets and compare it with some state-of-the-art methods. The experimental results show that our proposed method outperforms existing CNN models and some other state-of-the-art methods.

3.1. Datasets Description

We used four publicly available datasets: UC Merced Land Use Dataset(UCM) [44], SIRI-WHU [45], AID [24] and NWPU-RESISC45 Dataset (NWPU) [46]. Table 1 shows the basic information on these datasets, including the number of classes, the images per class, the total number of images, the spatial resolution and the image size.

Table 1. Information on the four datasets.

Datasets	No.of Classes	Image Per Class	No. of Images	Spatial Resolution (in meters)	Image Size	Training Ratio Setting
UCM	21	100	2100	0.3	256 × 256	50%, 80%
SIRI-WHU	12	200	2400	2	200 × 200	50%, 80%
AID	30	220–400	10,000	0.5–8	600 × 600	20%, 50%
NWPU	45	700	31,500	0.2–30	256 × 256	10%, 20%

3.1.1. UCM Dataset

The UCM dataset was manually extracted from large images of urban areas in the USGS National Map Urban Area Imagery collection for use in various urban areas across the country. The dataset contains 21 categories. There are 100 images per category. The spatial resolution is 0.3 m, and the image size is 256 × 256. The small size of this dataset and the fact that there are many categories make it quite challenging. Figure 7 shows the categories included in this dataset.

Figure 7. Examples of different categories in the UCM dataset.

3.1.2. SIRI-WHU Dataset

The SIRI-WHU dataset is from Google Earth and contains mainly different urban areas in China. The dataset contains 12 categories of images; there are 200 images per category. The image size is 200 × 200, and the spatial resolution is 2 m. Although the number of categories is small, the SIRI-WHU dataset contains a large number of similar images, and the sample size is small, so it is challenging. Some examples are shown in Figure 8.

Figure 8. Examples of different categories in the SIRI-WHU dataset.

3.1.3. AID Dataset

Compared with the UCM dataset, the AID dataset contains more categories and images, and the spatial resolution is variable. Therefore, it presents greater classification difficulties. The dataset contains 30 categories and 220 to 400 images per category. Spatial resolutions range from 0.5 to 8 m, and the image size is 600 × 600. Some scene images are shown in Figure 9.

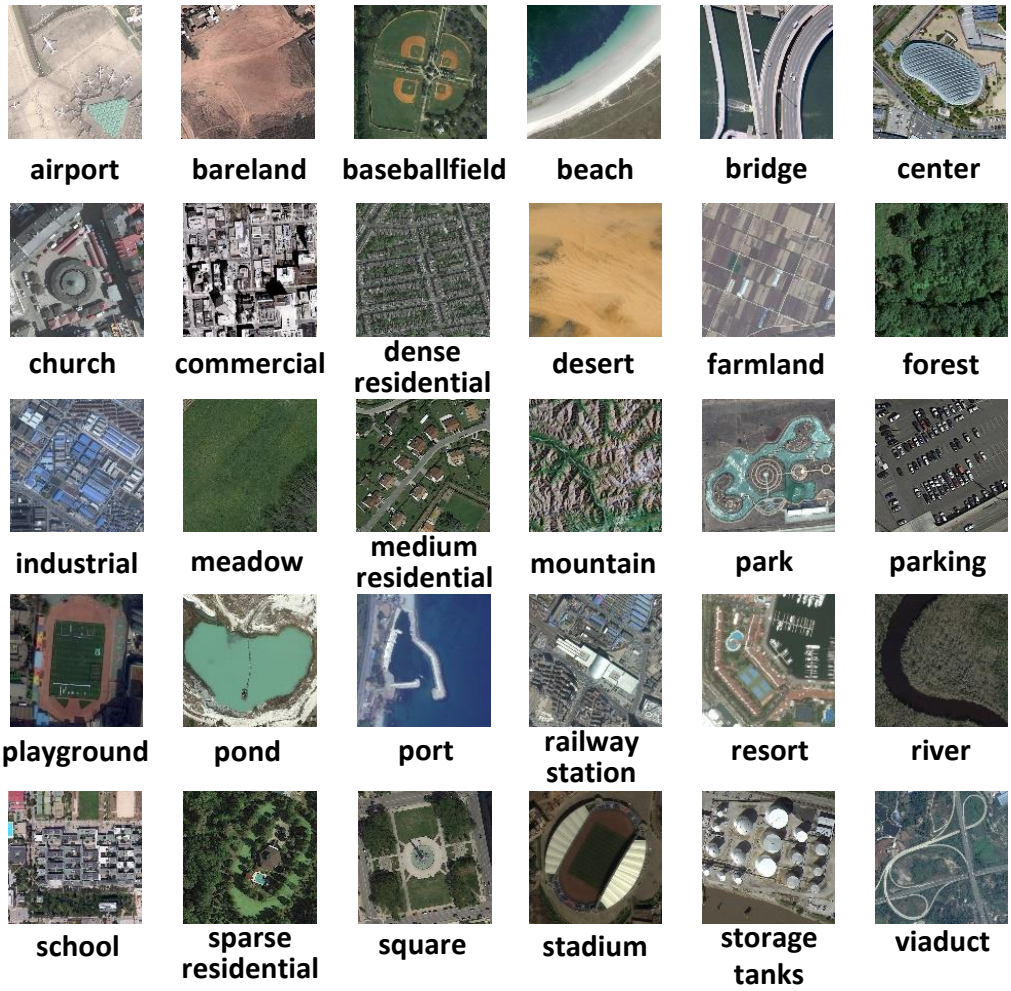

Figure 9. Examples of different categories in the AID dataset.

3.1.4. NWPU Dataset

The NWPU dataset is the most challenging. It contains more than 100 areas around the world, and images from satellite photography, aerial photography and Geographic Information Systems (GIS). Additionally, it was collected from different angles, in different lighting conditions and at different times of day and seasons, so the similarity between different categories is very high. The dataset has a size of 256 × 256 per image and 45 categories of scene images. There are 700 images per category, and a total of 31,500 images. It has large spatial resolution variation: 0.2–30 m. Some examples are shown in Figure 10.

3.2. Experimental Settings

To accurately evaluate the performance of the proposed method against those of other experimental methods [1,24,47–49], we selected training-to-test set ratios that are equal to those used in the majority of prior research. For the UCM and SIRI-WHU datasets, the training ratios were set to 50% and 80% (50% or 80% of the randomly selected images from the dataset were used for training, and the remaining images were used for testing.); for the AID dataset, 20% and 50%; and for the NWPU dataset, 10% and 20%. To obtain real and reliable experimental results, we randomly divided the four datasets according to the training ratios and repeated the experiment ten times. The mean and standard deviation were calculated as the final results of the MILRDA network.

Figure 10. Examples of different categories in the NWPU dataset.

The input size for all images was set to 224 × 224 pixels. The Adam optimizer was used to optimize the parameters, and its initial learning rate was set to 0.001. Training was carried out until the network converged; if the training loss did not decrease for ten consecutive epochs, the learning rate was divided by 10. Each of our methods was implemented in this study using the TensorFlow and Keras framework. All of the solutions were evaluated using a workstation equipped with a GeForce TiTan V GPU, 64 GB of memory and a Xeon(R) Gold 5222 CPU.

3.3. Results and Comparison

3.3.1. Experiments on the UCM Dataset

Table 2 shows the results of the proposed method and some state-of-the-art methods on the UCM dataset. Previous studies [24,46] have shown that deep learning-based methods are far superior to traditional handcrafted feature-based methods, so we do not compare the proposed approach with handcrafted feature methods here. As can be observed, the proposed MILRDA had the highest OAs under two training ratios, 98.19% and 98.81%. Under both training ratios, the enhanced methods based on the traditional VGGNet, AlexNet and GoogLeNet networks, such as TEX-Net with VGG [47], D-CNN with AlexNet [23] and DSFATN [48], perform bettered than the original methods. Some methods redesigned for remote-sensing scene images, such as ADFF [50], LSENet [22] and CIPAL [20], performed better. However, they still were 0.97%, 0.25% and 6.59% worse than our proposed MILRDA method, respectively. The numbers of parameters for DC-Net [43], although 0.16 M less than MILRDA, were 3.67% and 2.6% lower than our method for the two training ratios, respectively. since the UCM dataset is a small dataset with only 2100 images in total, the performance improvement from this small increase in parameters is within a reasonable range. Although Inception-v3-CapsNet [49] achieved 0.24% higher accuracy than our proposed method at high training ratios, it had 21.34M more parameters than our method. However, our method was still 0.6% more accurate than Inception-v3-CapsNet [49] at lower training ratios. This is because the residuals and dense connectivity

included in our method improve the feature extraction rate to some extent. In addition, for CNN-based methods, the results are better when the training ratio is high, because the network can learn more features. Figure 11 shows the CM on the UCM dataset. Of the 21 categories, all achieved accuracy of 95% or higher; 16 of them achieved 100% accuracy.

Table 2. Performance comparison on the UCM dataset(—: not reported).

Methods	OA(50/50)(%)	OA(80/20)(%)	Number of Parameters
AlexNet [24]	93.98 ± 0.67	95.02 ± 0.81	60 M
VGGNet-16 [24]	94.14 ± 0.69	95.21 ± 1.20	130 M
GoogLeNet [24]	92.70 ± 0.60	94.31 ± 0.89	7 M
TEX-Net with VGG [47]	94.22 ± 0.50	95.31 ± 0.69	130 M
D-CNN with AlexNet [23]	—	96.67 ± 0.10	60 M
CCP-net [51]	—	97.52 ± 0.97	130 M
ADFF [50]	97.22 ± 0.45	98.81 ± 0.51	23 M
DSFATN [48]	—	98.25	143 M
Inception-v3-CapsNet [49]	97.59 ± 0.16	99.05 ± 0.24	22 M
WSPM-CRC [52]	—	97.95	23 M
SAFF with AlexNet [28]	—	96.13 ± 0.97	60 M
DFAGCN [53]	—	98.48 ± 0.42	130 M
Fine-tune MobileNetV2 [54]	97.88 ± 0.31	98.13 ± 0.33	3.5 M
DC-Net [43]	94.52 ± 0.63	96.21 ± 0.67	0.5 M
GBNet [29]	97.05 ± 0.19	98.57 ± 0.48	18 M
LSENet [22]	97.94 ± 0.35	98.69 ± 0.53	130 M
RSNet [55]	—	96.78 ± 0.60	1.22 M
CIPAL [20]	91.96 ± 0.91	96.58 ± 0.76	1.53 M
ORRCNN [56]	96.58	96.42	—
MILRDA (ours)	98.19 ± 0.54	98.81 ± 0.12	0.66 M

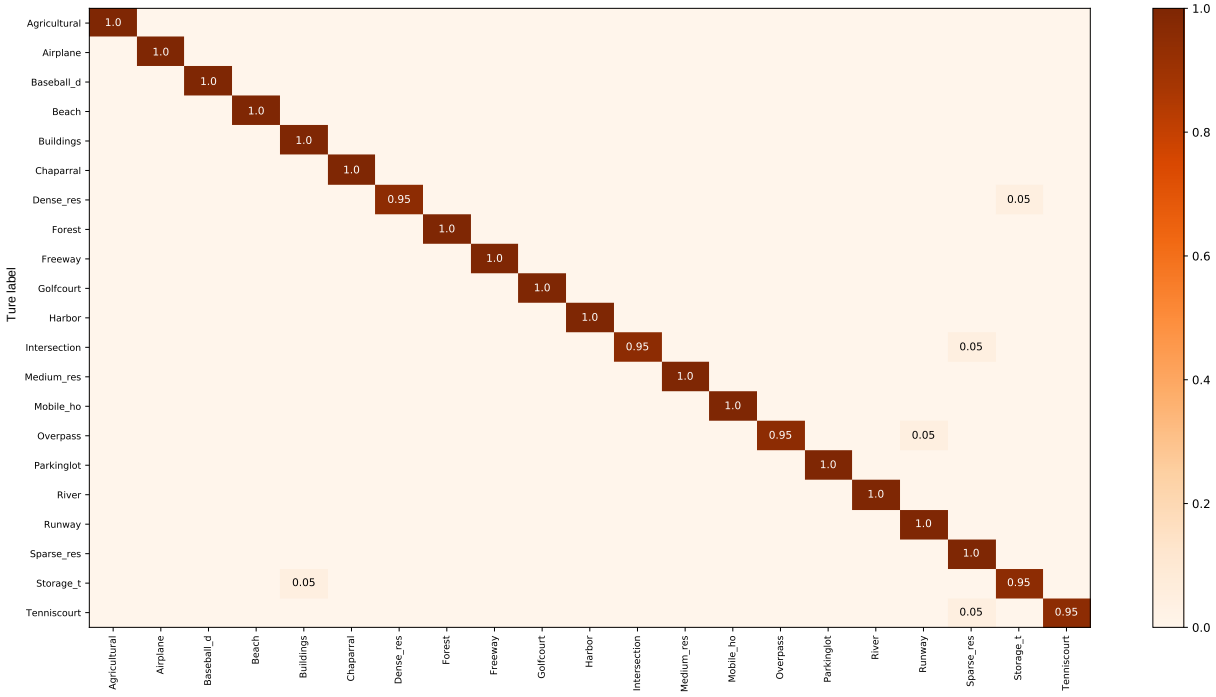

Figure 11. CM of the UCM dataset for an 80% training ratio (only displays results greater than 0.01).

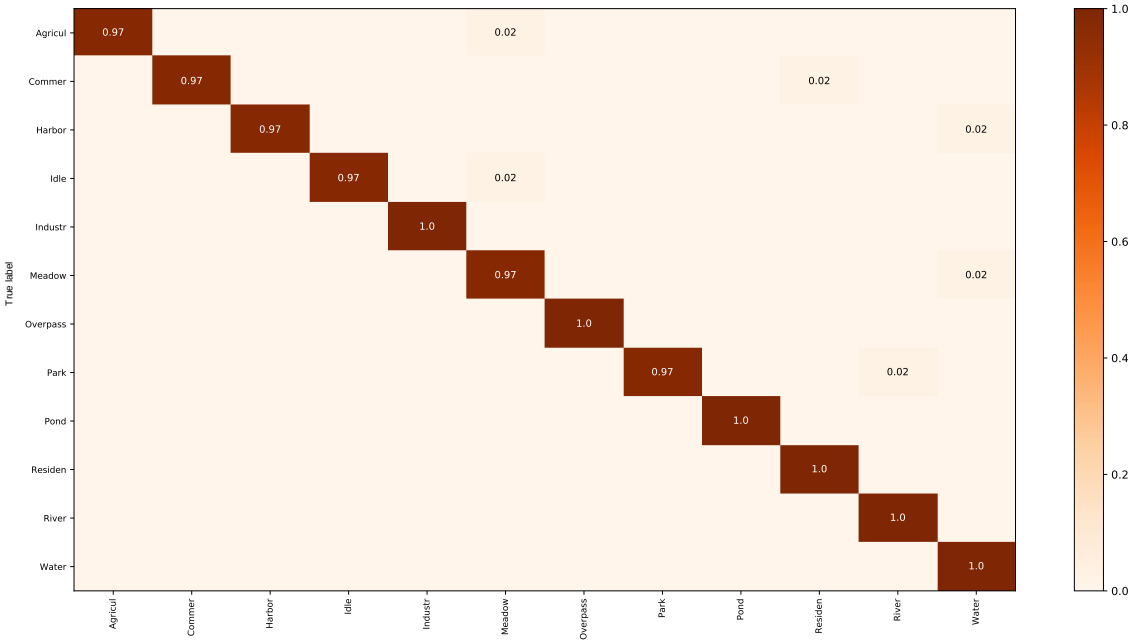

Figure 12. CM of the SIRI-WHU dataset for a 80% training ratio (only displays results greater than 0.01).

3.3.2. Experiments on the SIRI-WHU Dataset

Table 3 shows the results of the proposed method and some state-of-the-art methods on the SIRI-WHU dataset. The proposed method outperformed the other methods at all training ratios. With two training ratios of 50% and 80%, MILRDA achieved accuracies of 97.16 and 98.54. Compared to some large-scale methods, such as Siamese AlexNet [25], Siamese VGG16 and Siamese ResNet50, it was superior by 13.91%, 2.66% and 1.41%. Compared with the lightweight method SE-MDPMNet [54], our method was only 0.02% less accurate at the large training ratio but 0.2% more accurate at the small training ratio. In addition, ours had 4.51M fewer parameters than SE-MDPMNet [54]. Since MILRDA constructs a feature extraction backbone based on RDAB, which can perform feature reuse well, while suppressing redundant backgrounds and increasing key object weights through attention-based MIL pooling, our method is more effective on small sample datasets. Figure 12 shows the CM on the SIRI-WHU dataset. It can be seen that all categories achieved accuracy of 95% or more; six categories reached 100%.

Table 3. Performance comparison on the SIRI-WHU dataset (—-: not reported).

Methods	OA(50/50)(%)	OA(80/20)(%)	Number of Parameters
AlexNet [25]	82.50	88.33	60 M
VGGNet-16 [25]	94.92	96.25	130M
ResNet-50 [25]	94.67	95.63	26M
DMTM [45]	91.52	—-	—-
Siamese AlexNet [25]	83.25	88.96	60M
Siamese VGG16 [25]	94.50	97.30	130 M
Siamese ResNet50 [25]	95.75	97.50	26 M
Fine-tune MobileNetV2 [54]	95.77 ± 0.16	96.21 ± 0.31	3.5M
SE-MDPMNet [54]	96.96 ± 0.19	98.77 ± 0.19	5.17M
LPCNN [57]	—-	89.88	—-
SICNN [58]	—-	93.00	—-
Pre-trained-AlexNet-SPP-SS [26]	—-	95.07 ± 1.09	—-
SRSCNN [59]	93.44	94.76	—-
MILRDA (ours)	97.16 ± 0.37	98.75 ± 0.18	0.66M

3.3.3. Experiments on the AID Dataset

Table 4 shows the results of the different methods on the AID dataset. Our proposed method achieved accuracies of 91.95% and 95.46% at the two training ratios. Compared to the first two datasets, the AID dataset is more difficult to classify, so the accuracy of all methods was significantly lowered. Some redesigned models, such as DC-Net [43], GBNet [29] and LCNN-BFF [60], are more accurate than some improved methods based on pre-trained models, such as TEX-Net with VGG [47] and VGG16+MSCP [61], but these were still 5.58%, 1.74% and 0.82% less accurate than our proposed MILRDA. The light-weight models MIDC-Net [43] and RANet [62], although they had less parameters than MILRDA, had lower accuracy by 3.44% and 3.83%. Our proposed MILRDA method achieved a good balance between the number of parameters and accuracy. Figure 13 shows the CM of the AID dataset at a 50% training ratio. Of the 30 categories, for 13, category accuracy of 95% or higher was achieved; for 28, 90% or higher was achieved; and for 4 of them, 100% accuracy was achieved. Several categories with similar structures and landforms, such as schools and commercial buildings, towns centers and churches, are prone to mislabelling, but accuracy of over 80% was still achieved.

Table 4. Performance comparison on the AID dataset (—: not reported).

Methods	OA(20/80)(%)	OA(50/50)(%)	Number of Parameters
AlexNet [24]	86.86 ± 0.47	89.53 ± 0.31	60 M
VGGNet-16 [24]	86.59 ± 0.29	89.64 ± 0.36	130 M
GoogLeNet [24]	83.44 ± 0.40	86.39 ± 0.55	7 M
TEX-Net with VGG [47]	87.32 ± 0.37	90.00 ± 0.33	130 M
D-CNN with AlexNet [23]	85.62 ± 0.10	94.47 ± 0.12	60 M
Fusion by addition [63]	—	91.87 ± 0.36	—
WSPM-CRC [52]	—	95.11	23 M
DFAGCN [53]	—	94.88 ± 0.22	130 M
SAFF with AlexNet [28]	87.51 ± 0.36	91.83 ± 0.27	60 M
VGG16+MSCP [61]	91.52 ± 0.21	94.42 ± 0.17	130 M
AlexNet+MSCP [61]	88.99 ± 0.38	92.36 ± 0.21	60 M
GBNet [29]	90.16 ± 0.24	93.72 ± 0.34	18 M
DC-Net [43]	87.37 ± 0.41	91.49 ± 0.22	0.5 M
LCNN-BFF [60]	91.66 ± 0.48	94.64 ± 0.16	6.2 M
Skip-connected CNN [64]	91.10 ± 0.15	93.30 ± 0.13	6 M
CIPAL [20]	91.22 ± 0.83	93.45 ± 0.31	1.53 M
ORRCNN [56]	86.42	92.00	
LCPP [65]	90.96 ± 0.33	93.12 ± 0.28	
MILRDA (ours)	91.95 ± 0.19	95.46 ± 0.26	0.66 M

3.3.4. Experiments on the NWPU Dataset

Table 5 shows the results of the proposed MILRDA method compared with those of other methods. Some transfer-learning-based methods perform better with completely new data because knowledge from large datasets (such as ImageNet) is fully learned during pre-training. The MILRDA method achieved accuracies of 91.56% and 92.87% at the two training ratios, respectively. Some lightweight networks, such as SCCov [64], LCNN-BFF [60] and Contourlet CNN [66], also achieved good accuracy, but still 2.26%, 5.03% and 5.63% lower accuracy than our proposed method. In addition, the MILRDA method had the highest accuracy but the smallest number of parameters among all listed methods. Since the instance feature classifier in MILRDA contains a residual dense-connectivity structure for maximum feature reuse, CA helps to extract key local features, and MIL pooling based on channel attention helps to suppress background information (other categories) and highlight the weights of important instances under bag label supervision. Figure 14 shows the CM of the NWPU dataset at the 20% training ratio. Of the 45 categories, an accuracy rate of 90% or higher was achieved for 32; for five of them, 100% accuracy was

reached. Two scene categories, circular farmland and rectangular farmland, are prone to mislabelling because they contain similar objects, and the categories river and terrace are prone to classification errors due to similar scene topographic features, but still resulted in 75% accuracy.

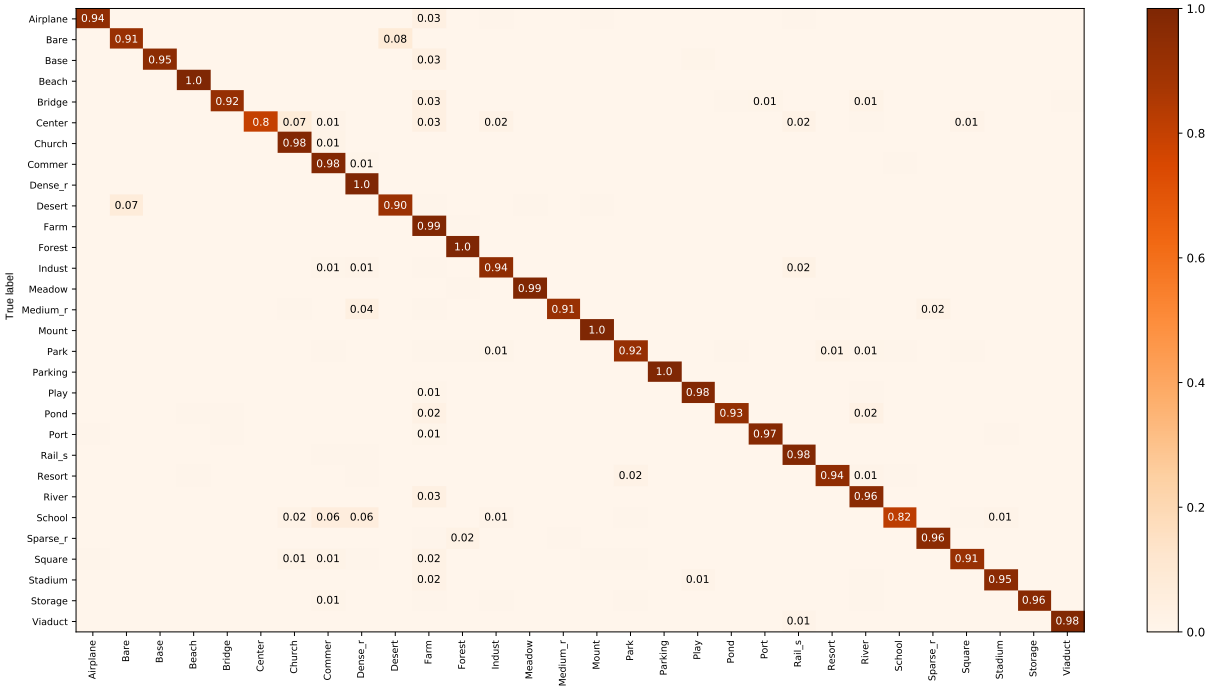

Figure 13. CM of the AID dataset for a 50% training ratio (only displays results greater than 0.01).

Table 5. Performance comparison on the NWPU dataset (—-: not reported).

Methods	OA(10/90)(%)	OA(20/80)(%)	Number of Parameters
AlexNet [46]	76.69 ± 0.21	79.85 ± 0.13	60M
VGGNet-16 [46]	76.47 ± 0.18	79.79 ± 0.15	130M
GoogLeNet [46]	76.19 ± 0.38	78.48 ± 0.26	7M
Fine-tuned VGG16 [46]	87.15 ± 0.45	90.36 ± 0.18	130M
Fine-tuned AlexNet [46]	81.22 ± 0.19	85.16 ± 0.18	60M
Fine-tuned GoogLeNet [46]	82.57 ± 0.12	86.02 ± 0.18	7M
DFAGCN [53]	—-	89.29 ± 0.28	130M
TFADNN [67]	87.78 ± 0.11	90.86 ± 0.24	130M
SAFF with AlexNet [28]	80.05 ± 0.29	84.00 ± 0.17	60M
Contourlet CNN [66]	85.93 ± 0.51	89.57 ± 0.45	12.6M
Inception-v3-CapsNet [49]	89.03 ± 0.21	92.60 ± 0.11	22M
SCCov [64]	89.30 ± 0.35	92.10 ± 0.25	13M
MF^{2}Net [68]	90.17 ± 0.25	92.73 ± 0.21	—-
LCNN-BFF [60]	86.53 ± 0.15	91.73 ± 0.17	6.2M
ACNet [27]	91.09 ± 0.13	92.42 ± 0.16	130M
ACR-MLFF [69]	90.01 ± 0.33	92.45 ± 0.20	26M
AMB-CNN [70]	88.99 ± 0.14	92.42 ± 0.14	5.6M
MILRDA (ours)	91.56 ± 0.18	92.87 ± 0.26	0.66M

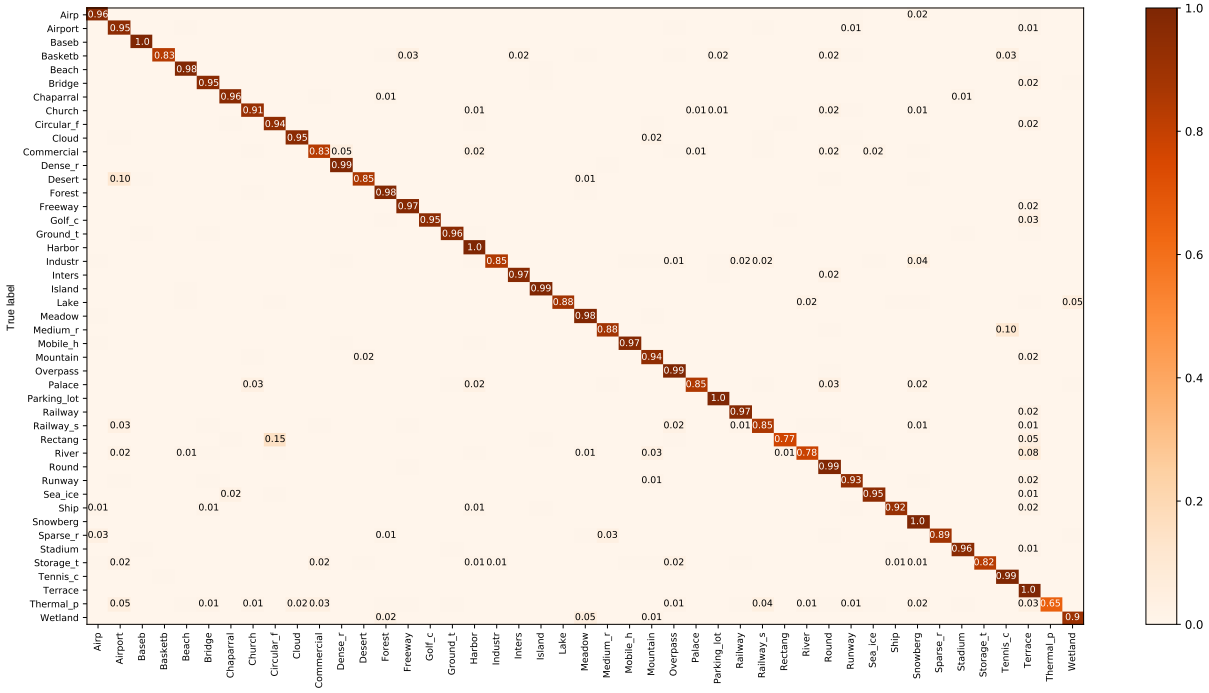

Figure 14. CM of the NWPU dataset for a 20% training ratio (only displays results greater than 0.01).

3.3.5. Prediction Time

Table 6 shows the prediction time for a single image from each of two datasets. Since the prediction time is closely related to the configuration of the computer, all experiments were performed on the same equipment. The prediction time of MILRDA was significantly shorter than those of the four baseline models that are widely used for remote-sensing scene-image classification. Additionally, these results show that reducing the parameters of the model is positively associated with improving the time efficiency. It is worth noting that the prediction time did not increase significantly when the size of the dataset became larger, indicating that our method can be applied to datasets with larger numbers of images.

Table 6. Prediction times of different methods on two datasets (in milliseconds).

Methods	AID (20%)	NWPU (10%)
AlexNet [46]	8.91	9.84
GoogLeNet [46]	5.33	6.17
VGGNet-16 [46]	12.56	13.11
ResNet-50 [25]	6.18	7.33
MILRDA (ours)	2.81	3.07

3.3.6. Model Size

Table 7 shows how the MILRDA method compares with other commonly used existing CNNs and some lightweight methods in terms of parameters and complexity. We use floating-point operations (FLOPs) to indicate model complexity, and in general, smaller values indicate lighter models. As can be seen in the table, our proposed method has far fewer parameters than some commonly used deep CNNs, such as AlexNet, GoogleNet, ResNet-50 and VGGNet-16. MILRDA's parameters total only 0.66 M, which is 60.24 M less than the deep CNN model AlexNet, but the FLOPs are slightly more numerous—about 0.22 G. This is due to the fact that MILRDA contains residual and dense connection that produce more floating-point operations, but as can be seen in Tables 4 and 5, our method is much more accurate than AlexNet for AID and NWPU datasets. Some recently proposed lightweight methods, SE-MDPMNet, Contourlet CNN and LCNN-BFF, still have far more

parameters and FLOPs than our proposed methods. In other words, the MILRDA method achieves a balance in terms of accuracy and model size.

Table 7. Comparing the size and complexity of different models.

Methods	Parameters (In Million)	Model size (In MByte)	Computational Complexity (In GFlops)
AlexNet [46]	60.9	232.7	0.72
GoogLeNet [46]	6.8	25.9	2.1
VGGNet-16 [46]	136	527.6	15.5
ResNet-50 [25]	26	87.2	2.91
SE-MDPMNet [54]	5.17	19.72	3.27
Contourlet CNN [66]	12.6	48	15.5
LCNN-BFF [60]	6.1	23.3	24.6
MILRDA (ours)	0.66	2.51	0.94

4. Discussion

The performance of our proposed MILRDA method is influenced by the number of building blocks in the RDAB. Due to space limitations, only the OA of M in UCM with a 50% training ratio and that of NWPU with a 10% training ratio are reported here, for when RDAB equals 1, 2, 3 and 4. Figure 15 shows the relevant results. The classification accuracy peaked when the number of RDAB modules is 3. With four RDAB modules, the number of convolutional layers increased, the classification accuracy decreased and overfitting occurred. When the number of RDAB modules was four, the classification accuracy could still compete with those of some of the other methods in Tables 2 and 5. This proves the effectiveness of the MILRDA method in feature extraction, and may also be related to the MIL pooling based on channel attention, which after all, highlights key local features.

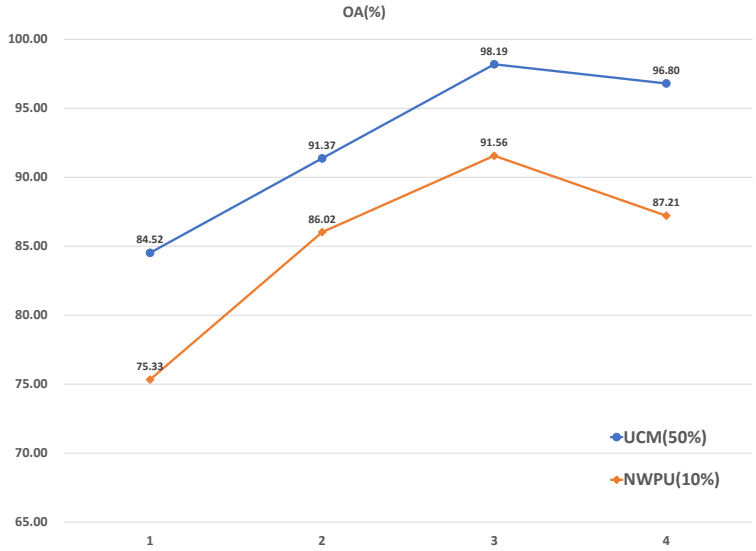

Figure 15. Effect of the number of RDAB modules on the classification accuracies in UCM and NWPU datasets.

To verify the effectiveness of MIL pooling based on channel attention, Figure 16 shows the different results on two datasets. RDANet indicates that MIL pooling is not included (the features extracted by the convolutional layer are fed directly into the final classification layer), and RDNet indicates that the CA mechanism is not included in the convolutional feature extraction backbone. Even without using any MIL pooling operation, our RDANet still achieved good results with higher classification accuracy than all existing deep CNN models, and still had some advantages over other methods (seen in Tables 2 and 5). This

was due to the fact that the features are maximally reused under the effect of residuals and dense connectivity, which improves the diversity of features while highlighting the weights of key regions within the CA mechanism. Note that even without including the CA mechanism, RDNet still achieved good results, proving the effectiveness of the model. Remote-sensing scene images contain a large amount of redundant background information; the MIL pool based on channel concerns can not only alleviate this problem, but also highlight the weights of key instances, and to a certain extent solve the research difficulties of small inter-class differences and large intra-class differences in remote-sensing scene images; in addition, the pooling is trainable, and the MILRDA model as a whole can be trained under the supervision of bag-level semantic labels.

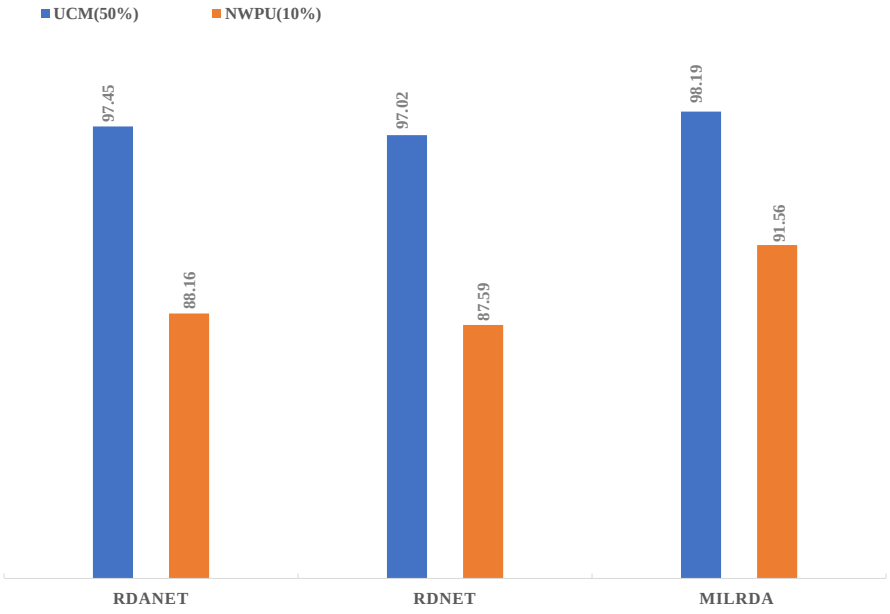

Figure 16. Effects of different structures on experimental effects.

For the sake of observation, Figure 17 shows the feature maps of some categories. With the effect of MIL pooling based on channel attention, useless background information or other categories not related to bag-level labeling were suppressed. For example, for the category "tennis court," the image contains multiple objects, such as trees, residential areas and roads. With MIL pooling, regions associated with true bag-level semantic labels were highlighted, and other instances were given reduced weights, showing the effectiveness of this pooling operation. Even in the absence of MIL pooling, our proposed method still produced good feature extraction results. This is because the CA mechanism in the feature extraction backbone can help represent the key local features while expanding the corresponding regions. However, this would include some useless background regions, further demonstrating the effectiveness of MIL pooling for suppressing the weights of irrelevant objects.

Figure 17. Visualization of feature maps of example vectors before and after MIL pooling.

5. Conclusions

In this paper, we proposed a lightweight model called MILRDA for remote-sensing scene-image classification, providing an end-to-end MIL framework with strong local semantic representation. In this framework, dense connections and residual connections are first used as the feature extraction backbone, and then an adaptive downsampling layer is formed with the control unit through a CA mechanism to highlight local features while focusing on large regions. The features extracted through the backbone network are then transformed into instance feature vectors, which are further eliminated from redundant background information while highlighting the weights of important instances

through a trainable MIL pooling layer based on channel attention. Finally, optimization and classification are performed by a cross-entropy loss function under the supervision of bag-level labels. We validated the method on four publicly available remote sensing scene datasets, and our experiments showed that our proposed method MILRDA outperforms other state-of-the-art methods. The numbers of parameters and FLOPs are much smaller in MILRDA than in existing CNNs and some lightweight methods, proving the effectiveness of the method. For future work, we will further feature framework compatibility, improve the extraction capability of the sign extraction backbone for multi-layer features and further improve performance.

Author Contributions: Conceptualization, X.W. (Xinyu Wang) and H.X.; methodology, X.W. (Xinyu Wang); software, L.Y.; validation, H.X. and W.D.; formal analysis, X.W. (Xinyu Wang); investigation, L.Y.; resources, D.W.; data curation, X.W. (Xinyu Wang); writing—original draft preparation, X.W. (Xinyu Wang); writing—review and editing, X.W. (Xianbin Wen); visualization, L.Y.; supervision, X.W. (Xianbin Wen); project administration, X.W. (Xianbin Wen); funding acquisition, X.W. (Xianbin Wen) All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the New-Generation AI Major Scientific and Technological Special Project of Tianjin (18ZXZNGX00150), and in part by the Special Foundation for Technology Innovation of Tianjin (21YDTPJC00250).

Institutional Review Board Statement: Not Applicable.

Informed Consent Statement: Not Applicable.

Data Availability Statement: The UC Merced Land Use Dataset (UCM), SIRI-WHU, AID and NWPU-RESISC45 (NWPU) datasets presented in this work are openly available.

Acknowledgments: The authors would like to thank the anonymous reviewers for their very competent comments and helpful suggestions.

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

References

1. Wang, X.; Yuan, L.; Xu, H.; Wen, X. CSDS: End-to-end aerial scenes classification with depthwise separable convolution and an attention mechanism. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 10484–10499.
2. Cheng, G.; Xie, X.; Han, J.; Guo, L.; Xia, G.S. Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 3735–3756.
3. Zhang, L.; Han, Y.; Yang, Y.; Song, M.; Yan, S.; Tian, Q. Discovering discriminative graphlets for aerial image categories recognition. *IEEE Trans. Image Process.* **2013**, *22*, 5071–5084.
4. Longbotham, N.; Chaapel, C.; Bleiler, L.; Padwick, C.; Emery, W.J.; Pacifici, F. Very high resolution multiangle urban classification analysis. *IEEE Trans. Geosci. Remote Sens.* **2011**, *50*, 1155–1170.
5. Tayyebi, A.; Pijanowski, B.C.; Tayyebi, A.H. An urban growth boundary model using neural networks, GIS and radial parameterization: An application to Tehran, Iran. *Landsc. Urban Plan.* **2011**, *100*, 35–44.
6. Chen, W.; Li, X.; He, H.; Wang, L. Assessing different feature sets' effects on land cover classification in complex surface-mined landscapes by ZiYuan-3 satellite imagery. *Remote Sens.* **2017**, *10*, 23.
7. Li, X.; Shao, G. Object-based urban vegetation mapping with high-resolution aerial photography as a single data source. *Int. J. Remote Sens.* **2013**, *34*, 771–789.
8. Ahmed, Z.; Hussain, A.J.; Khan, W.; Baker, T.; Al-Askar, H.; Lunn, J.; Al-Shabandar, R.; Al-Jumeily, D.; Liatsis, P. Lossy and lossless video frame compression: A novel approach for high-temporal video data analytics. *Remote Sens.* **2020**, *12*, 1004.
9. Kleanthous, N.; Hussain, A.; Khan, W.; Sneddon, J.; Liatsis, P. Deep transfer learning in sheep activity recognition using accelerometer data. *Expert Syst. Appl.* **2022**, *207*, 117925.
10. Hu, J.; Xia, G.S.; Hu, F.; Sun, H.; Zhang, L. A comparative study of sampling analysis in scene classification of high-resolution remote sensing imagery. In Proceedings of the 2015 IEEE International geoscience and remote sensing symposium (IGARSS), Milan, Italy, 26–31 July 2015; pp. 2389–2392.
11. Swain, M.J.; Ballard, D.H. Color indexing. *Int. J. Comput. Vis.* **1991**, *7*, 11–32.
12. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–26 June 2005; Volume 1, pp. 886–893.
13. Oliva, A.; Torralba, A. Modeling the shape of the scene: A holistic representation of the spatial envelope. *Int. J. Comput. Vis.* **2001**, *42*, 145–175.

14. Ojala, T.; Pietikainen, M.; Maenpaa, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987.
15. Zhu, Q.; Zhong, Y.; Zhao, B.; Xia, G.S.; Zhang, L. Bag-of-visual-words scene classifier with local and global features for high spatial resolution remote sensing imagery. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 747–751.
16. Hofmann, T. Unsupervised learning by probabilistic latent semantic analysis. *Mach. Learn.* **2001**, *42*, 177–196.
17. Blei, D.; Ng, A.; Jordan, M. Latent dirichlet allocation. In Proceedings of the 2001 Neural Information Processing Systems (NIPS) Conference, Vancouver, BC, Canada, 3–8 December 2001.
18. Hu, F.; Xia, G.S.; Hu, J.; Zhang, L. Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sens.* **2015**, *7*, 14680–14707.
19. Penatti, O.A.; Nogueira, K.; Dos Santos, J.A. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Boston, MA, USA, 7–12 June 2015; pp. 44–51.
20. Chen, J.; Huang, H.; Peng, J.; Zhu, J.; Chen, L.; Tao, C.; Li, H. Contextual information-preserved architecture learning for remote-sensing scene classification. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–14.
21. Tang, X.; Lin, W.; Ma, J.; Zhang, X.; Liu, F.; Jiao, L. Class-Level Prototype Guided Multiscale Feature Learning for Remote Sensing Scene Classification With Limited Labels. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–15.
22. Bi, Q.; Qin, K.; Zhang, H.; Xia, G.S. Local semantic enhanced convnet for aerial scene recognition. *IEEE Trans. Image Process.* **2021**, *30*, 6498–6511.
23. 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.
24. 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.
25. 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.
26. Han, X.; Zhong, Y.; Cao, L.; Zhang, L. Pre-trained alexnet architecture with pyramid pooling and supervision for high spatial resolution remote sensing image scene classification. *Remote Sens.* **2017**, *9*, 848.
27. Tang, X.; Ma, Q.; Zhang, X.; Liu, F.; Ma, J.; Jiao, L. Attention consistent network for remote sensing scene classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 2030–2045.
28. Cao, R.; Fang, L.; Lu, T.; He, N. Self-attention-based deep feature fusion for remote sensing scene classification. *IEEE Geosci. Remote Sens. Lett.* **2020**, *18*, 43–47.
29. Sun, H.; Li, S.; Zheng, X.; Lu, X. Remote sensing scene classification by gated bidirectional network. *IEEE Trans. Geosci. Remote Sens.* **2019**, *58*, 82–96.
30. Tang, P.; Wang, X.; Feng, B.; Liu, W. Learning multi-instance deep discriminative patterns for image classification. *IEEE Trans. Image Process.* **2016**, *26*, 3385–3396.
31. Dietterich, T.G.; Lathrop, R.H.; Lozano-Pérez, T. Solving the multiple instance problem with axis-parallel rectangles. *Artif. Intell.* **1997**, *89*, 31–71.
32. Wang, X.; Wang, B.; Bai, X.; Liu, W.; Tu, Z. Max-margin multiple-instance dictionary learning. In Proceedings of the International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013; pp. 846–854.
33. Wang, Q.; Yuan, Y.; Yan, P.; Li, X. Saliency detection by multiple-instance learning. *IEEE Trans. Cybern.* **2013**, *43*, 660–672.
34. Wang, C.; Huang, K.; Ren, W.; Zhang, J.; Maybank, S. Large-scale weakly supervised object localization via latent category learning. *IEEE Trans. Image Process.* **2015**, *24*, 1371–1385.
35. Bi, Q.; Zhou, B.; Qin, K.; Ye, Q.; Xia, G.S. All Grains, One Scheme (AGOS): Learning Multi-grain Instance Representation for Aerial Scene Classification. *arXiv* **2022**, arXiv:2205.03371.
36. Wang, X.; Yan, Y.; Tang, P.; Bai, X.; Liu, W. Revisiting multiple instance neural networks. *Pattern Recognit.* **2018**, *74*, 15–24.
37. Pinheiro, P.O.; Collobert, R. From image-level to pixel-level labeling with convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1713–1721.
38. 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.
39. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
40. Hou, Q.; Zhou, D.; Feng, J. Coordinate attention for efficient mobile network design. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 13713–13722.
41. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
42. Zhang, Y.; Tian, Y.; Kong, Y.; Zhong, B.; Fu, Y. Residual dense network for image super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2472–2481.
43. Bi, Q.; Qin, K.; Li, Z.; Zhang, H.; Xu, K.; Xia, G.S. A multiple-instance densely-connected ConvNet for aerial scene classification. *IEEE Trans. Image Process.* **2020**, *29*, 4911–4926.

44. 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, 2–5 November 2010; pp. 270–279.
45. Zhao, B.; Zhong, Y.; Xia, G.S.; Zhang, L. Dirichlet-derived multiple topic scene classification model for high spatial resolution remote sensing imagery. *IEEE Trans. Geosci. Remote Sens.* **2015**, *54*, 2108–2123.
46. Cheng, G.; Han, J.; Lu, X. Remote sensing image scene classification: Benchmark and state of the art. *Proc. IEEE* **2017**, *105*, 1865–1883.
47. Anwer, R.M.; Khan, F.S.; van de Weijer, J.; Molinier, M.; Laaksonen, J. Binary patterns encoded convolutional neural networks for texture recognition and remote sensing scene classification. *ISPRS J. Photogramm. Remote Sens.* **2018**, *138*, 74–85.
48. Gong, X.; Xie, Z.; Liu, Y.; Shi, X.; Zheng, Z. Deep salient feature based anti-noise transfer network for scene classification of remote sensing imagery. *Remote Sens.* **2018**, *10*, 410.
49. Zhang, W.; Tang, P.; Zhao, L. Remote sensing image scene classification using CNN-CapsNet. *Remote Sens.* **2019**, *11*, 494.
50. 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.
51. Qi, K.; Guan, Q.; Yang, C.; Peng, F.; Shen, S.; Wu, H. Concentric circle pooling in deep convolutional networks for remote sensing scene classification. *Remote Sens.* **2018**, *10*, 934.
52. 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.
53. Xu, K.; Huang, H.; Deng, P.; Li, Y. Deep Feature Aggregation Framework Driven by Graph Convolutional Network for Scene Classification in Remote Sensing. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 5751–5765.
54. 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.
55. Wang, J.; Zhong, Y.; Zheng, Z.; Ma, A.; Zhang, L. RSNet: The search for remote sensing deep neural networks in recognition tasks. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 2520–2534.
56. Li, Z.; Wu, Q.; Cheng, B.; Cao, L.; Yang, H. Remote sensing image scene classification based on object relationship reasoning CNN. *IEEE Geosci. Remote Sens. Lett.* **2020**, *19*, 1–5.
57. Zhong, Y.; Fei, F.; Zhang, L. Large patch convolutional neural networks for the scene classification of high spatial resolution imagery. *J. Appl. Remote Sens.* **2016**, *10*, 025006.
58. Zhong, Y.; Fei, F.; Liu, Y.; Zhao, B.; Jiao, H.; Zhang, L. SatCNN: Satellite image dataset classification using agile convolutional neural networks. *Remote Sens. Lett.* **2017**, *8*, 136–145.
59. Liu, Y.; Zhong, Y.; Fei, F.; Zhu, Q.; Qin, Q. Scene classification based on a deep random-scale stretched convolutional neural network. *Remote Sens.* **2018**, *10*, 444.
60. 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.
61. 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.
62. Bi, Q.; Qin, K.; Zhang, H.; Li, Z.; Xu, K. RADC-Net: A residual attention based convolution network for aerial scene classification. *Neurocomputing* **2020**, *377*, 345–359.
63. Chaib, S.; Liu, H.; Gu, Y.; Yao, H. Deep feature fusion for VHR remote sensing scene classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 4775–4784.
64. 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.
65. Sun, X.; Zhu, Q.; Qin, Q. A multi-level convolution pyramid semantic fusion framework for high-resolution remote sensing image scene classification and annotation. *IEEE Access* **2021**, *9*, 18195–18208.
66. Liu, M.; Jiao, L.; Liu, X.; Li, L.; Liu, F.; Yang, S. C-CNN: Contourlet convolutional neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 2636–2649.
67. Xu, K.; Huang, H.; Deng, P.; Shi, G. Two-stream feature aggregation deep neural network for scene classification of remote sensing images. *Inf. Sci.* **2020**, *539*, 250–268.
68. Xu, K.; Huang, H.; Li, Y.; Shi, G. Multilayer feature fusion network for scene classification in remote sensing. *IEEE Geosci. Remote Sens. Lett.* **2020**, *17*, 1894–1898.
69. Wang, X.; Duan, L.; Shi, A.; Zhou, H. Multilevel feature fusion networks with adaptive channel dimensionality reduction for remote sensing scene classification. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 1–5.
70. Shi, C.; Zhao, X.; Wang, L. A multi-branch feature fusion strategy based on an attention mechanism for remote sensing image scene classification. *Remote Sens.* **2021**, *13*, 1950.