*electronics*

MDPI

*Article*

# Fine-Grained Recognition of Surface Targets with Limited Data

**Runze Guo, Bei Sun, Xiaotian Qiu, Shaojing Su *, Zhen Zuo and Peng Wu**

College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China; guorunze14@nudt.edu.cn (R.G.); sunbei08@nudt.edu.cn (B.S.); qiuxtnudt@sina.com (X.Q.); z.zuo@nudt.edu.cn (Z.Z.); pengwu@nudt.edu.cn (P.W.)

* Correspondence: ssjing@nudt.edu.cn

Received: 3 November 2020; Accepted: 30 November 2020; Published: 2 December 2020


**Abstract:** Recognition of surface targets has a vital influence on the development of military and civilian applications such as maritime rescue patrols, illegal-vessel screening, and maritime operation monitoring. However, owing to the interference of visual similarity and environmental variations and the lack of high-quality datasets, accurate recognition of surface targets has always been a challenging task. In this paper, we introduce a multi-attention residual model based on deep learning methods, in which channel and spatial attention modules are applied for feature fusion. In addition, we use transfer learning to improve the feature expression capabilities of the model under conditions of limited data. A function based on metric learning is adopted to increase the distance between different classes. Finally, a dataset with eight types of surface targets is established. Comparative experiments on our self-built dataset show that the proposed method focuses more on discriminative regions, avoiding problems like gradient disappearance, and achieves better classification results than B-CNN, RA-CNN, MAMC, and MA-CNN, DFL-CNN.

**Keywords:** surface targets; multi-attention residual model; transfer learning; fine-grained recognition

## 1. Introduction

The recognition of surface targets has attracted broad interest from researchers in recent decades because of its diverse applications in marine monitoring and investigations, such as maritime patrols, rescue operations, and illegal-vessel screening, which are crucial for coastal countries [1]. Typical surface targets include rocks, reefs, buoys, fishing boats, tugboats, and different types of warships. In addition, the recognition approaches should resolve various types of environmental changes and view-points.

In the past two decades, various algorithm for surface target recognition have been proposed. These methods can be roughly divided into moment-based [2], knowledge-based [3], model-based [4], and neural network-based [5,6]. However, these methods are mainly aimed at fixed scenes, static targets, and require a large amount of labeled data, which are difficult to implement in water scenes. Therefore, previous methods do not perform well in water scenes. In addition, changing the distance and view-points results in variations in the targets' size and posture, which brings huge challenges for surface target recognition. It can be summarized that the differences in intra-class are large and those in inter-class are small. Some visually similar subcategories cannot even be recognized by untrained-personnel. With the development of deep learning, convolutional neural networks (CNNs) are widely used to detect and classify objects. However, the CNN model is more prone to overfitting because of complicated and similar appearance of surface targets. Moreover, a majority of existing methods used in detection and recognition are based on remote sensing images [7] rather than visible light images. With this regard, there are a few works aimed at visible light images of surface targets.

Therefore, there is no complete and large-scale dataset that limits the improvement of networks' learning ability.

Therefore, as shown in Figure 1, a surface target fine-grained recognition algorithm based on deep learning is proposed under limited-data conditions. First, a dataset of eight types of targets including rocks, reefs, buoys, fishing boats, tugboats and other types of warships is established in response to the lack of a public surface target dataset. Thereafter, owing to serious background information interference, changing the posture and scale of surface targets, a new multi-attention residual model is proposed, which allows the deep learning network to focus on the areas with distinguishing characteristics, thus avoiding gradient disappearance and gradient explosion. Moreover, transfer learning [8] is adopted to improve feature expression capabilities under the condition of limited training data. The parameters of the network pretrained on large-scale datasets are transferred to the task of surface target recognition. The parameters of the first convolutional layers are frozen, and the rest of the parameters should be fine-tuned on our self-built dataset. Finally, for the problem of small differences between subclasses and large differences within classes, a loss function based on metric learning [9] is introduced, which is suitable for multidimensional targets. It can target diverse dimensions and enrich the feature information of surface targets to make the neural network converge better and faster.

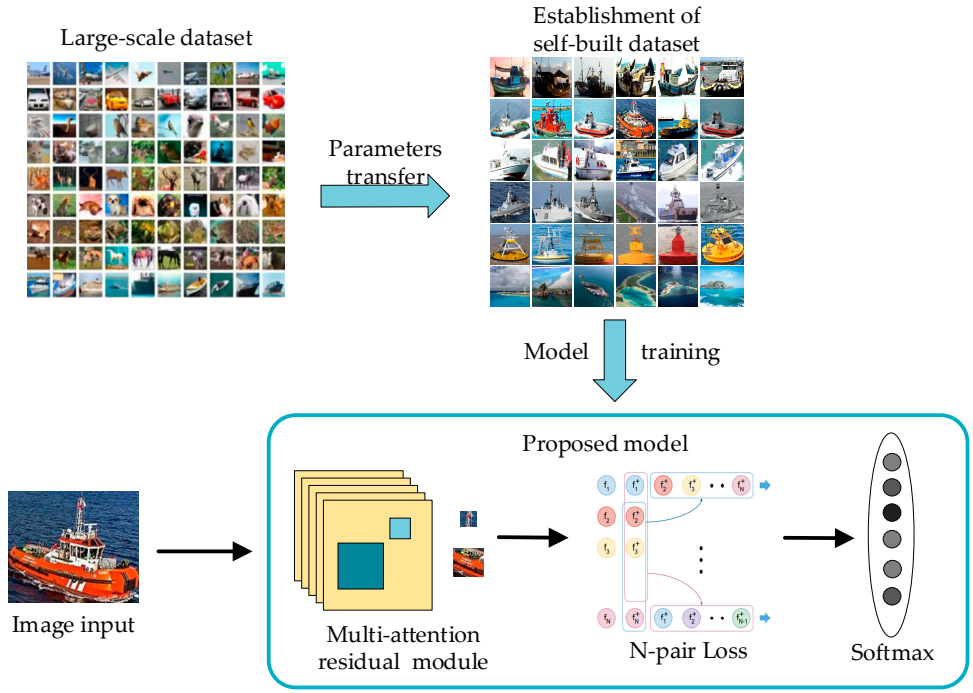


**Figure 1.** Illustration of the proposed algorithm. At the top, a limited self-built dataset is established and parameters of large-scale datasets are transferred into the former. At the bottom, an image is input into the multi-attention residual model, and then N-pair loss based on metric learning is used for classification.

In summary, the main contributions of this paper are as follows:

1. Establishing a dataset of surface targets, including different categories.
2. Introducing a multi-attention mechanism based on residual networks, and fusing the channel attention module and spatial attention module to focus on the discriminative area and further improve the classification accuracy.
3. Transfer learning and the N-pair loss function are adopted to make the networks express richer features and converge better and faster.

After training and testing, the accuracy of this model reached 90.5% on self-built dataset. Experimental results show that the method proposed in this paper can accurately identify surface targets, and is superior to most mainstream fine-grained recognition methods. The remainder of this paper is organized as follows. Section 2 introduces the research progress of fine-grained classification. Section 3 presents the multi-attention residual models and other methods used in this study. The training process and experimental results are described in Section 4, and Section 5 presents the conclusions.

## 2. Literature Review

The recognition of surface targets, especially classification of different ships, is essentially a fine-grained classification [10]. The purpose of fine-grained classification is to distinguish different subcategories of objects within the same category. This article refers to the division of different subcategories of surface targets. Owing to changes in the background and shooting angle of view, the appearance of objects of the same subcategory may vary greatly. At the same time, the visual differences of objects of different categories may be relatively small. Therefore, fine-grained classification is a challenging task. According to the differences in the structure and number of neural networks, the existing fine-grained classification methods based on deep learning can be divided into the following three types: (1) Integrating single or multiple deep neural networks for direct classification; (2) applying deep neural networks as feature extractors to locate and align different parts of fine-grained objects better; and (3) using attention mechanisms to enhance the focus on distinguished regions.

In this section, we first introduce the development of a single CNN and the integration of multiple networks. Thereafter, a method of positioning and alignment based on part detection is presented. The last part of this section reviews the attention-based methods.

### *2.1. CNN and Ensemble of CNN*

CNN has a long history in the field of computer vision. It was first proposed by LeCun et al. [11], and applied to recognition tasks. The superior performance of LeNet-5 in recognition tasks prompted researchers to apply it to fine-grained image classification. It serves as a backbone of neural networks even today. Later, AlexNet [12] containing eight learnable layers, was the ILSVRC-2012 competition Winner, with a test error rate of 15.3%. The large number of trainable parameters and the introduction of activation functions promoted the development of deep learning. VGGNet [10] achieved the highest accuracy of the ILSVRC14 classification and positioning tasks. Its depth increased by using 3 to 3 filters, and its layers included 16–19 floors. An innovation of VGGNet is that all hidden layers are equipped with rectification nonlinearity in order to reduce the computational burden and overfitting. GoogLeNet [13] created the latest level of classification and detection in ILSVRC14. GoogLeNet has a total of 22 deep layers and adopts the "inception module." The convolution kernel of $1 \times 1$ reduces the dimensions and improves the utilization of computing resources. It can be seen that with the development of deep learning, the number of layers in the networks is increasing. However, owing to the problems of gradient disappearance and gradient explosion, deeper networks will cause performance degradation.

ResNet [14], using a "shortcut" to fuse low-level features with high-level features, was proposed in order to solve this problem. In ResNet, the final output not only has the features passed down through the multilayer network but also has the initial features of the network. Therefore, the residual block could reduce the influence of gradient disappearance. With increasing number of layers of deep learning networks, their ability to express features improves, but a single CNN has a limited effect on fine-grained classification with small visual differences. Thus, the CNN integration is proposed.

The subset feature learning network [15] is one of the representatives of the ensemble network, which consists of two main parts: a general CNN and several specific CNNs. The general CNN is trained and fine-tuned on the total object dataset, whereas specific CNNs are trained on K subsets of visually similar subcategories. Learning a separate CNN for each K subsets of the target dataset makes it easier to distinguish visually similar species. However, when there are many subcategories,

the number of CNNs will be very large, which will undoubtedly increase the complexity of the structure and will not be conducive to implementation. Similar to the subset feature learning network, the MixDCNN [16] system also learns K-specific CNNs. However, it does not need to divide the dataset into subsets of K similar images in advance. The image is fed into all K CNNs, and the output from each CNN is combined to form a classification decision. MixDCNN needs to train K CNNs, and the parameters that need to be trained are quite large, so it consumes significantly more computing resources. The bilinear model [17] is an integrated network composed of two feature extractors. The output is fused before entering the classifier to obtain an image descriptor. The bilinear model used for image classification consists of four tuples, $\beta = (f_A, f_B, P, C)$, where $f_A$ and $f_B$ are the feature functions, $P$ is the pooling function, and $C$ is the classification function. The difference in the feature function is reflected in the composition of the convolutional layer and the pooling layer, which is related to the characteristics of the image. Compared with the previous networks, the bilinear model occupies less network and computing resources, which is more conducive to implementation.

### *2.2. Methods Based on Location Detection and Alignment*

In fact, for different subcategories, subtle differences in appearance only exist in several parts, such as the head and torso of birds or the deck and sides of ships. Therefore, it is important to locate and detect related parts to reduce the influence of posture and shooting angle changes in fine-grained classification. A majority of methods based on location detection and alignment follow the process shown in Figure 2.

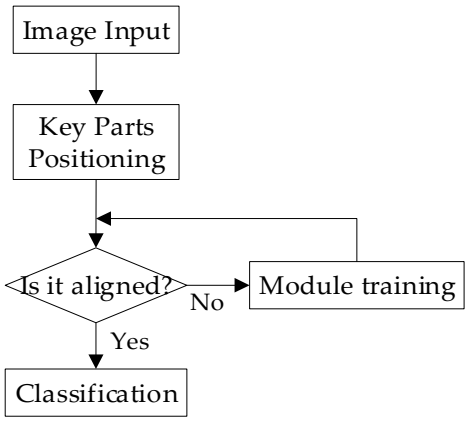


**Figure 2.** Process of methods based on detection and alignment. First, images are input into convolutional neural networks (CNNs), through which the key points or regions can be extracted and located. Thereafter, the module determines whether the images are aligned with prototypes. If not aligned, the images would be trained again until they are aligned. Finally, the images can be classified accurately.

Zhang et al. [18] proposed a part-based R-CNN, which uses the R-CNN [19] algorithm to first detect the object level (such as ships) and its local areas (deck, cabin, etc.,) on fine-grained images. Then, the image patch is divided as input, and train by CNN separately. Eventually, the features of object level and local area are fused to obtain the feature description of fine-grained image. The algorithm has strong practicability and high accuracy. Its shortcomings are as follows: (1) The bottom-up region generation method used generates a large number of irrelevant regions; (2) a bounding box and part annotation are required in training and testing, which limits the application of part-based R-CNN in actual scenes.

The pose normalized CNN [20] uses the DPM to predict 2D positions and 13 key points of semantic parts, or directly uses the provided object frame and part labelling information to learn pose prototypes. The images of different parts are wrapped, and different DCNNs (AlexNet) are used to extract their features. The features of each part and the entire image are stitched to train the classifier.

The innovation of the pose normalized CNN is to use the prototype to perform the posture alignment operation on the image and to extract the features of different network layers for different local areas in an order to construct a more discriminative feature representation. In addition, the interference of different postures of subcategories is further considered to reduce the impact of intra-class variance, thereby achieving better performance. The pose normalized CNN achieved a 2% higher classification accuracy than the based R-CNN when using the same amount of labelled information.

With the wide application of the FCN [21], the Mask CNN [22] was subsequently proposed. The model is divided into two modules, the first is a part localization block, and the second is the feature learning of global and local image blocks. It should be pointed out that the difference from the previous two works is that a part-based segmentation model is learned with the help of the FCN in the Mask-CNN. After obtaining the Part Mask, the corresponding image block can be obtained by cropping. At the same time, two masks can be combined to form a complete object mask. Moreover, part mask and object mask also play the role of "selecting useful convolutional descriptor", which can remove the interference of background convolutional descriptors, and then describe the feature description of fine-grained images. Although the method based on part detection and alignment largely avoids the influence of posture and shooting position and improves the accuracy of fine-grained recognition, the acquisition of annotation information, such as bounding boxes and part annotation is very expensive. To a certain extent, this limits the practical application of such algorithms.

### *2.3. Methods Based on Visual Attention*

The vision attention mechanism is a signal processing mechanism unique to human vision. Specifically, when people are looking at something, the vision system first obtains the target area that needs attention by quickly scanning the global image, and then suppresses irrelevant information to obtain the target of interest. Because the method based on the vision attention model can locate distinguishing areas in the image without additional annotation information (such as the target location bounding box and the location labelling information of key parts), it has been widely used in recent years, especially in the field of fine-grained detection and classification of images.

The two level attention algorithm [23] was the first attempt to achieve fine-grained image recognition without relying on additional annotation information, but only using category labels. After that, two-level attention has been widely used in the field of fine-grained recognition. Two-level attention combines three types of attention: bottom-up attention for generating candidate image blocks, object-level top-down attention for selecting related blocks to form specific objects, and component-level bottom-up attention for locating discriminative components. A specific DCNN is trained by integrating these types of attention mechanisms to extract foreground objects and components with strong features. The model is easy to generalize and does not require bounding boxes and component annotations. Overall, the two-level attention model solves the problem of how to detect local areas when there are only category labels. However, the accuracy of the local areas obtained by the clustering algorithm is limited.

Another representative work is the recurrent attention convolutional neural network (RA-CNN) proposed in 2017 [24]. This model imitates the RPN (region proposal network) network in a faster-RCNN [25], and proposes to use the APN (attention proposal network) network to locate the discriminative regions in images. It uses the rank loss function (rank loss) in the training process to ensure that the attention model is positioned more accurately. However, the RA-CNN is not very robust and cannot handle unusual gestures, which are common in tasks of fine-grained recognition.

Later, a three-linear attention sampling network (TASN) [26] was proposed to solve the problems above. The proposed TASN consists of a three-linear attention module, an attention-based sampler and a feature extractor, which are used to learn the subtle feature representations from a few hundred fine-grained image recognition suggestions. The partial network and the main network are denoted as "teacher" and "student" respectively, and the TASN is optimized in the teacher–student approach. The part network learns fine-grained features from images that retain details. The main net takes the

retained structure of an image as input, and refines a specific part (guided by the partial net) in each iteration. Therefore, fine-grained features can be efficiently extracted into a single main network.

However, the existing fine-grained algorithms are not suitable for surface target recognition for the following reasons: (1) As the data scale of surface targets is limited, the neural network cannot obtain better feature expression ability; (2) owing to different shooting angles and the incompleteness of distinguishing parts, the diversity of postures is not easy to align; (3) the scale inconsistency and the blur of local information caused by different shooting distances are not conducive to the training and learning of key information.

## 3. Methodology

This section includes four subsections: (1) surface target dataset; (2) proposed model; (3) transfer learning; (4) loss function based on metric learning.

### *3.1. Surface Target Dataset*

Currently, there is no public large-scale dataset on surface targets. In order to obtain a reliable and complete dataset, we not only download highly recognizable pictures from the website, but also use Hikvision DS-9008/9016HQH-XT (Hikvision, Hangzhou, China) to take images of various ships in the field to form a surface and marine target dataset. Hikvision DS-9008/9016HQH-XT (Hikvision, Hangzhou, China) supports third-party cameras and has access to HD-TVI cameras. In addition, 800W pixel high-definition network video preview, storage, and playback are allowed in this camera. The source of data is not only the captured pictures but also the intercepted video frame images. Moreover, both HDMI1 and VGA output resolutions can reach up to 1080p, which can get more high-quality pictures.

As shown in Figure 3, there are a total of eight types of targets including rocks, reefs, buoys, fishing boats, tugboats, and various warships. Images in Figure 3a are downloaded from websites, and images in Figure 3b are taken or intercepted in real scenes. It should be noted that the original size and posture of each image are different. In order to facilitate the processing of the model, the size of each picture is normalized.

**(a)** **(b)**

**Figure 3.** Samples of self-built dataset: (**a**) Examples of images that are downloaded from the website; (**b**) examples of images that are collected from real scenes and intercepted from videos.

### *3.2. Proposed Model*

#### 3.2.1. ResNet-50

The features learned by deep networks can be enriched by the stack of network layers (depth). Theoretically, better results can be achieved when the model is deeper. However, there exists degradation problems in deep networks. As the depth of the network increases, the accuracy of the network becomes saturated or even decreases. Residual learning is introduced to solve the degradation problem. For a stacked layer structure like Figure 4, when the input is $x$, the learned feature is $H(x)$. Now we believe that it can learn the residual $F(x) = H(x) - x$, so that the original learning feature is actually $F(x) + x$. The reason for this is that residual learning is easier than direct learning of original features. When the residual is 0, the stacked layer only performs identity mapping at this time. In fact, the network performance does not decrease. Since the residual will not be 0, which would make the stacked layer learn new features based on the input features so as to have better performance.

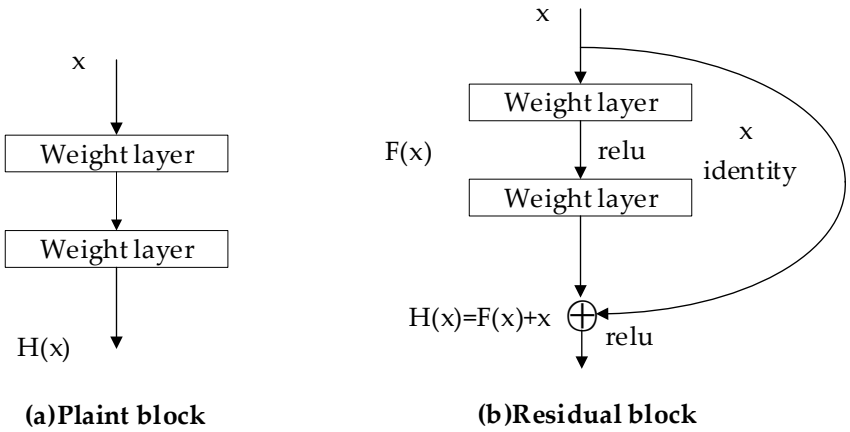


**Figure 4.** Structures of convolutional blocks: (**a**) the plaint block; (**b**) the residual block.

The proposal of ResNet effectively solves the problem of the disappearance of the deep network gradient, greatly improves the classification accuracy, and has good portability, which is helpful for the introduction of transfer learning. For the recognition of surface targets, it is necessary to learn the subtle differences. The required training parameters are huge, which makes it impossible to use neural networks with fewer layers. Therefore, this study selects ResNet-50 as the part of model feature extraction. ResNet-50 contains five groups of conv1-conv5 convolutional blocks, a total of 49 convolutional layers and a fully connected layer, which is 50 layers together.

#### 3.2.2. Multi-Attention Model

Attention mechanism is a common feature-enhancement strategy in the field of image classification and recognition, derived from the simulation of the unique visual signal processing mechanism of the human brain. When people observe and recognize objects, they will focus their attention on the salient parts of the target while ignoring some global and background information. This selective attention mechanism is similar to extracting discriminative parts in fine-grained image classification tasks. Therefore, the attention mechanism has been widely used in the field of fine-grained image classification in recent years. In the task of surface target recognition, there is a large amount of irrelevant or disturbing information, so it is very important to focus on the areas with distinguishing characteristics. In addition, for objects such as ships, different shooting distances and angles will lead to changes in scale and posture. Grasping its inherent and spatial characteristics also plays an important role in accurate recognition of surface targets. To this end, we introduce a multidimensional attention mechanism, as shown in Figure 5: extracting the attention weight maps of the channel and

space dimensions respectively, and distributing the weights to the original feature maps for feature fusion, and finally entering the next stage of feature extraction.

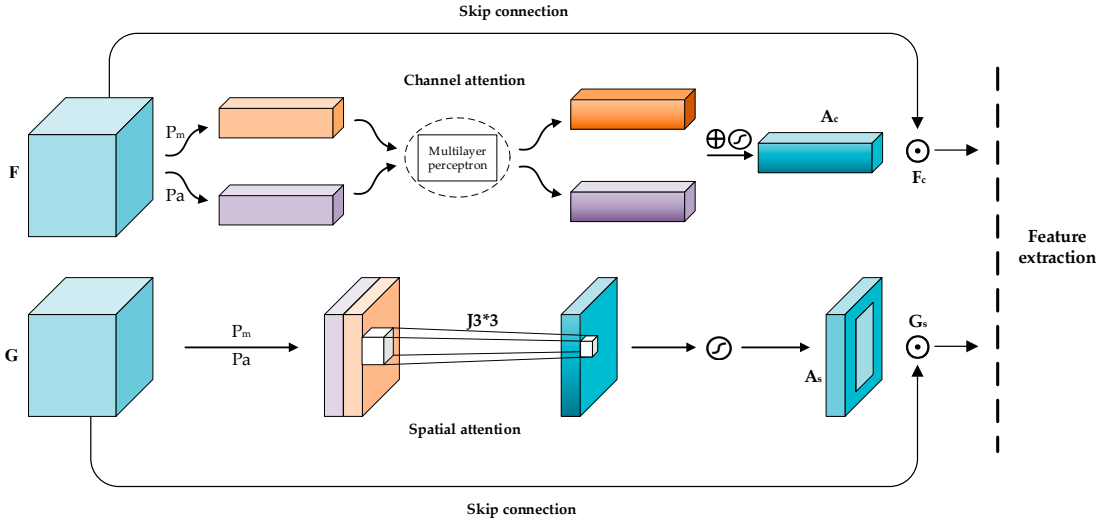


**Figure 5.** The structure of the proposed multi-dimensional model.

The following are the feature extraction and fusion processes of channel attention and spatial attention modules:

1. Channel attention module

The convolution feature map generated by the residual model contains different feature channels. In fine-grained image classification, some feature channels may represent different information in the image, such as color and size, and channels may contain irrelevant or redundant information, such as background and noise. Therefore, focusing on effective feature channels that contain discriminative local information and giving them a higher weight distribution can significantly improve the accuracy of fine-grained classification.

The feature extraction and fusion steps of the channel attention module are as follows:

- Take the convolutional feature map $F$ generated by the residual network as the original input, let $F \in R^{w \times h \times t}$, where $w \times h$ represents the spatial dimension, and $t$ represents the number of channels. In order to effectively extract the channel attention, $F$ will be compressed in the spatial dimension, and the feature of the same channel is compressed into a real number. This step can be achieved by pooling operation.
- Adopt a multi-scale pooling method: use the maximum pooling function $p_m$ and the average pooling function $p_a$ respectively to reduce the dimensionality to obtain two sized feature vectors $1 \times 1 \times t$. Then, input the two vectors into the same shared network to obtain the attention of the channel dimension weight distribution. The shared network is composed of a multi-layer perceptron with a hidden layer.
- Perform the corresponding element summation operation on the two output vectors after redistributing the attention weight, and use the Sigmod activation function to map the merged feature vector to generate the channel attention weight, $A_c \in R^{1 \times 1 \times t}$.
- Finally, the attention weight $A_c$ and the original feature map $F$ are fused. Here, a fusion method of multiplying the corresponding elements is used to obtain the fused attention feature map $F_c$, $F_c \in R^{w \times h \times t}$. Replace the original input features $F$ with $F_c$ to achieve the attention extraction of the channel dimension.

The channel attention extraction and fusion process can be represented as follows:

$$A_c = \sigma\{M[p_m(F)] \oplus M[p_m(F)]\} \quad (1)$$

$$F_c = F \odot A_c \tag{2}$$

In Equation (1), $p_m$ and $p_a$ are the maximum pooling and average pooling functions, respectively. $M$ represents a shared network containing multi-layer perceptron, while $\oplus$ represents the summation of the corresponding elements of the vector. In Equation (2), $\sigma$ indicates the Sigmod activation function. $A_c$ is the channel attention weights, $A_c \in R^{1\times1\times t}$. $\odot$ shows the multiplication operation of the corresponding elements of the vector, while $F_c$ is the attention feature map after step fusion, $F_c \in R^{w\times h\times t}$.

2. Spatial attention module

Different from the channel attention module, the spatial attention module is aimed at focusing on the spatial position information of targets, which is a supplement to the channel attention. It is also important for the learning of variable scale and posture data. It is necessary to introduce spatial position information for surface targets, since their posture would change with the shooting angle.

The feature extraction and fusion steps of the spatial attention module are as follows:

- The convolutional feature map $G$ generated by the residual network is used as the original input, $G \in R^{w\times h\times t}$. $w \times h$ represents the spatial dimension, while $t$ is the number of channels. $G$ is compressed to obtain the spatial attention information along the channel axis, through which a list of channel values is compressed into a channel. This step can be achieved by the pooling of the channel dimension.
- The same multi-scale pooling method is adopted: the maximum pooling function $p_m$ and the average pooling function $p_a$ are used for dimensionality reduction to obtain two feature maps, size of which is $w \times h \times 1$. The two feature maps are spliced along the channel axis using the corresponding element summation method. Get a new feature map of size.
- Use a $3 \times 3$ convolution kernel to convolve the spliced feature map, compress it again to $w \times h \times 1$, and use the Sigmod activation function to map the convolved feature map to generate a spatial attention map $A_s$, $A_s \in R^{w\times h\times1}$.
- Finally, the spatial attention map $A_s$ and the original feature map $G$ are fused using the corresponding element point multiplication method, and the fused spatial attention feature map $G_s$ is obtained, $G_s \in R^{w\times h\times t}$. The original input feature $G$ is replaced with $G_s$ to achieve the attention extraction of the spatial features.

The spatial attention extraction and fusion can be expressed by the following equations.

$$A_s = \sigma\left\{J^{3*3}[p_m(G) \oplus p_m(G)]\right\} \tag{3}$$

$$G_s = G \odot A_s \tag{4}$$

In Equation (3), $p_m$ and $p_a$ are the maximum pooling and average pooling functions, respectively. $\oplus$ represents the summation operation of corresponding elements, and $J^{3*3}$ means the convolution operation using a $3 \times 3$ convolution kernel of size. $\sigma$ is the Sigmod activation function, and $A_s$ is the spatial attention map, $A_s \in R^{w\times h\times1}$. In Equation (4), $\odot$ indicates the dot product operation of the corresponding element, while $G_s$ is the attention feature map after fusion, $G_s \in R^{w\times h\times t}$.

After adding two attention modules with different dimensions, the network has acquired richer attention features.

### *3.3. Transfer Learning*

Training of deep learning networks requires large datasets. The amount and quality of data directly affect the generalization function and robustness of the learned model. In particular, when the training sample is insufficient, the generalization function of the model is poor, but it is expensive to collect a sufficient amount of high-quality data for surface targets. Transfer learning allows us

to handle data-constrained scenarios by using the data labels of other related tasks or domains that already exist. This method stems from the fact that people could obtain useful knowledge from familiar source domains when faced with unfamiliar target tasks. In the task of surface target recognition, a large amount of effective data cannot be obtained because of the high cost of collection and labeling. Direct training with limited data may result in severe overfitting or failure to converge. Therefore, this article introduces transfer learning to combine the general features learned under public datasets with specific features of surface targets under a self-built small sample dataset as Figure 6. First, the model is pretrained on a large-scale public dataset (we choose ImageNet here). Thereafter, part of the parameters is transferred into our self-built small dataset for tuning.

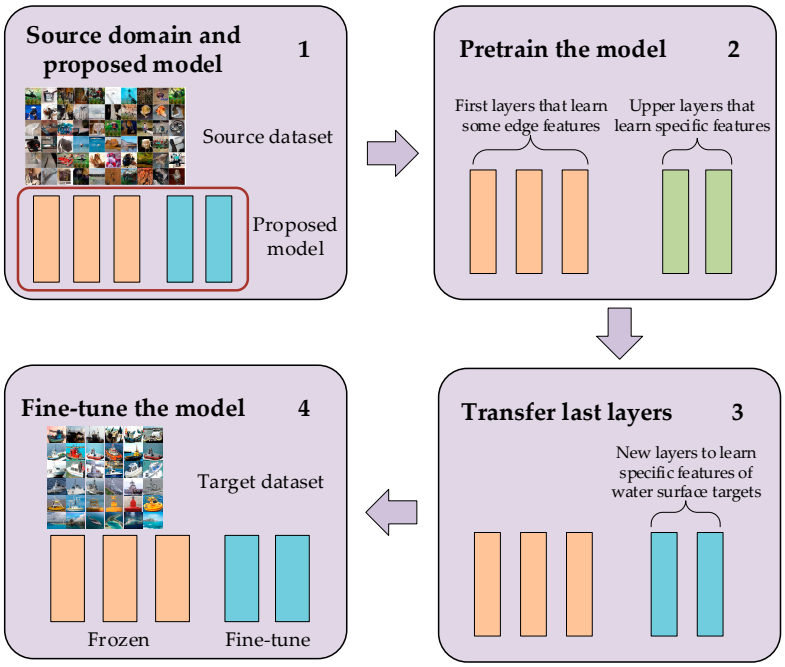


**Figure 6.** Steps of transfer learning. First, suitable source dataset should be found. Then, the model is pretrained on the source domain. Eventually, parameters are transferred and fine-tuned on our self-built dataset.

The deep learning network used in this task is inherently transferable. CNNs have a good hierarchical structure. Generally speaking, ordinary CNNs have a hierarchical structure of convolution-pooling-convolution-full connection. When the depth is considerable, the CNN can extract the features of each level. For the recognition of surface targets, especially when recognizing ships, the shallow layer of convolutional layers usually extracts the color, contour, and edge features of the image, while the upper layer of convolutional layers extracts features that have obvious identification elements such as sizes, postures, decks, and barrels.

According to the data size of the own sample and the similarity between the own dataset and the source dataset, transfer learning methods can be divided into four categories:

- When the amount of data is small and the datasets are highly similar, the linear classifier only needs to be trained.
- When the amount of data is large and the similarity of the datasets is high, multiple layers need to be fine-tuned. In other words, network weights should be pre-trained to initialize the weights of the network.
- When the amount of data is small and the datasets are very different, most of the network needs to be reinitialized.

- The amount of data is very large, and the datasets are very different, the datasets need to be fine-tuned with multiple layers.

In order to enlarge the feature expression ability of the deep learning network, we choose ImageNet as the source dataset, which contains a total of 1000 different objects and various types of ships. Because the self-built small sample dataset has limited data and is quite different from the source dataset, it is necessary to initialize most of the network according to the above method 3, that is, to freeze the shallow features and fine-tune multiple layers. The parameters of which layer needs to be frozen are determined by the results of the experiment.

### *3.4. Loss Function Based on Metric Learning*

This study uses N-pair loss in metric learning. The purpose of metric learning is to learn a data-embedding expression technology that can make similar data closer and dissimilar data more far away in the embedding space. In recent years, deep metric learning has attracted widespread attention. It can learn a non-linear embedded expression and has achieved great success in the field of face recognition and image retrieval. When recognizing visually similar surface targets, metric learning works well. In order to increase the difference between different subclasses, we choose to use N-pair loss.

In metric learning, triplet loss was the first that was widely used. It targets both a pair of positive samples and a pair of negative samples as shown in Figure 7a. By introducing triplet loss, neural networks can make the distance between positive samples as small as possible, and make the distance between negative sample pairs as large as possible, so as to increase the inter-class differences and reduce the intra-class differences. However, this method can only focus on a pair of negative sample pairs, and lacks the ability to distinguish various types of samples, which is not suitable for situations where there are many types of targets. In order to improve this situation, (N + 1)-tuplet loss that selects multiple negative sample pairs is proposed. In other words, a pair of positive sample pairs selects all other samples of different categories as negative samples and combine them to obtain negative sample pairs, as shown in Figure 7b.

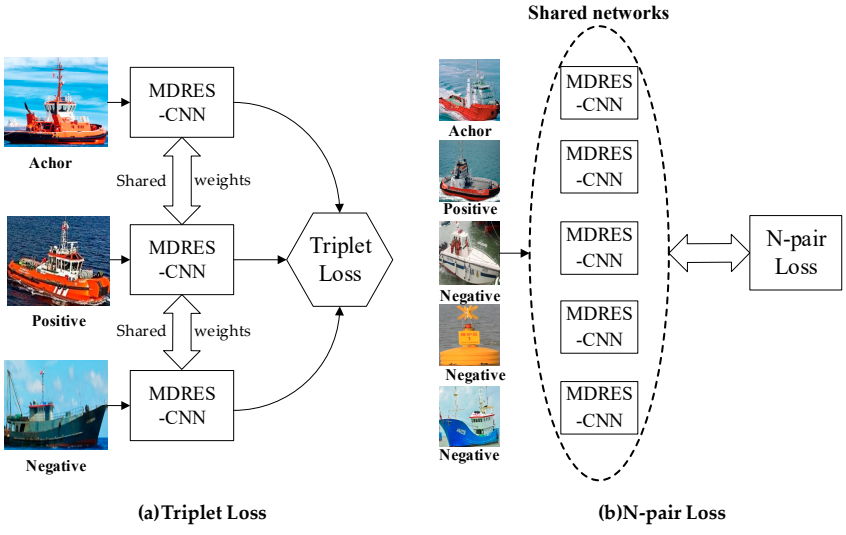


**Figure 7.** The diagram of different loss functions: (**a**) triplet loss; (**b**) N-pair loss.

If there are $N$ categories in the dataset, each positive sample pair $y_{ii}$ corresponds to $N-1$ negative sample pairs.

Set $x \in \chi$ as input data, and output label $y \in \{1, 2, \ldots, L\}$ for it. $x^+$ and $x^-$ are positive and negative samples of $x$ to indicate that $x$ and $x^+$ are from the same class, $x$ and $x^-$ are from different classes,

respectively. The kernel $f(\cdot;\theta) = \chi \to \mathbb{R}^K$ generates embedded vector $f(x)$. For simplicity, $x$ in $f(x)$ is usually omitted, and $f$ would inherit all subscripts and subscripts. (N+1)-tuplet loss is defined as:

$$L\left(\left\{x, x^+, \{x_i\}_{i=1}^{N-1}; f\right\}\right) = \log\left(1 + \sum_{i=1}^{N-1} \exp(f^T f_i - f^T f^+)\right) \tag{5}$$

Tuplet loss involves all categories of negative examples that are desirable, but it is impractical when the number of output categories $L$ is large. Even if we limit the number of negative examples for each category to one, performing standard optimization (such as random gradient descent (SGD)) is still too onerous. In order to avoid excessive computational burden, this test introduces an effective batch-processing structure. Let $\left\{\left(x_1, x_1^+\right), \ldots, \left(x_N, x_N^+\right)\right\}$ be instances from $N$ different categories, and construct $N$ tuples from $N$ pairs of samples, denoted by $\{S_i\}_{i=1}^N$, where $S_i = \left\{x_i, x_1^+, x_2^+, \ldots, x_N^+\right\}$. Here $x_i$ is a positive sample and $x_j$, $j \neq i$ is a negative sample. N-pair loss can be expressed as:

$$L_{N-pair-mc}\left(\left\{\left(x_i, x_i^+\right)\right\}_{i=1}^N; f\right) = \frac{1}{N}\sum_{i=1}^{N} \log\left(1 + \exp(f_i^T f_j^+ - f_i^T f_i^+)\right) \tag{6}$$

## 4. Experiments and Analysis

### *4.1. Parameters Setting and Data Enhancement*

In this study, the open source deep learning framework PyTorch is used as the platform, and NVIDIA GTX 1060T (NVIDIA, Santa Clara, CA, USA) is used to optimize through SGD. Owing to the small size of the self-built dataset, training directly on the residual network may not converge. Therefore, ResNet-50 is first pre-trained on the ImageNet for initialization, and then the model is transferred and trained again on self-built surface target dataset. When images are trained on ImageNet, networks iterate 80,000 steps with a learning rate of 0.001, 40,000 steps with a learning rate of 0.0001, and 20,000 steps with a learning rate of 0.0001. We also set the batch size to 128. Thereafter, the model parameters are transferred to the self-built dataset for training. At this time, we set the batch to 16, and networks iterate 20,000 steps with a learning rate of 0.01, 10,000 steps with a learning rate of 0.001, and 10,000 steps with a learning rate of 0.0001. The model uses SGD or Adam to train and optimize the network, and the learning rate is set as above. The momentum is set to 0.9, and the weight attenuation is set to $5 \times 10^{-4}$.

In order to improve the generalization ability and robustness of the model, data enhancement methods are used, including rotation, mirroring, contrast enhancement, brightness enhancement, random cropping, and other methods. The data enhancement methodology could be divided into online enhancement and offline enhancement. Online enhancement refers to the use of algorithms to enhance the original data randomly during the training process. This method does not need to change the local dataset and can directly use the GPU for calculation. Offline enhancement is used to save the enhanced data directly, which needs to enlarge the size of the dataset. The combination of the two methods can achieve more possibilities. The number of images after rotation (different angles), mirroring, contrast enhancement, and brightness enhancement is approximately six times that of the original training set, and about 1/6 of them are selected as the verification set to train the parameters of the model. After offline enhancement, the total number of our self-built dataset reached 3246, of which the number of frigates reaches 594, and the number of reefs is the least and there are 270. The number of other types of targets is between 350 and 420. Figure 8 shows an example of data enhancement for the training set (the image in the example comes from our self-built dataset).

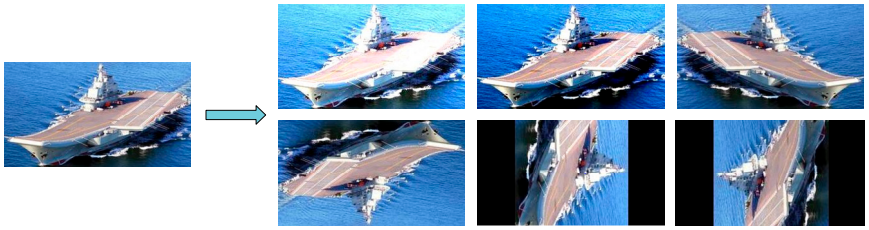

**Figure 8.** Examples of data enhancement.

In the testing phase, the classification accuracy is used as the result evaluation index. It is one of the most commonly used indexes in image classification and defined as the proportion of correctly classified images to the number of images in this category. The number of surface targets in each type of dataset is different, so the total accuracy cannot be used directly to evaluate the performance of the network. In order to ensure the balance of the test system and the accuracy of the final results, we randomly select images from each type of target to form a test set and ensure that their numbers are equal. The classification accuracy of proposed algorithm on the self-built dataset is the average of the classification accuracy of all categories.

$$\text{Accuracy}_i = \frac{\text{N}_i^{\text{c}}}{\text{N}_i} \tag{7}$$

In Equation (7), $\text{Accuracy}_i$ represents the correct rate of classification in category $i$, $\text{N}_i^{\text{c}}$ represents the number of the correctly classified images in the test set, and $\text{N}_\text{i}$ represents the number of images in category $i$.

$$\text{Accuracy} = \frac{\sum_{i=1}^{n} \text{Accuracy}_i}{n} \tag{8}$$

In Equation (8), Accuracy represents the classification accuracy of the proposed model on the self-built dataset, $n$ shows the number of categories. Judging whether the image is classified correctly is determined by judging whether the real label and the predicted label of the image are consistent.

### *4.2. Network Visualization and Display of Experiments*

For surface targets, residual blocks could extract many features, including contour, color, texture, and so on. However, a large number of unrelated areas in images would consume computing resources, and even affect the accuracy of recognition. Therefore, attention mechanism is applied in our proposed module to focus more on multi-dimension and spatial information, which reduces the inference of irrelevant background information. We use the proposed model on our self-built dataset, and save the generated contour map and attention map. Figure 9 illustrates these results.

From Figure 9, discriminative areas could be obtained through proposed attention model instead of unrelated regions. We also record the confusion matrixes of a selected dataset to evaluate the performance of proposed method. In order to ensure the effectiveness of the confusion matrixes, we randomly select 800 images in the test set for verification. The data capacity of 800 sheets is large enough to avoid contingency and the number of images of each type reaches 100 sheets. The larger data capacity is also to make the result of the confusion matrix more meaningful and representative. Our proposed model is compared with MAMC [27], which is a weakly supervised attention method and has the advantage of end-to-end training. It performs well in fine-grained classification and is used to compare with other methods. The results are demonstrated in Figure 10.

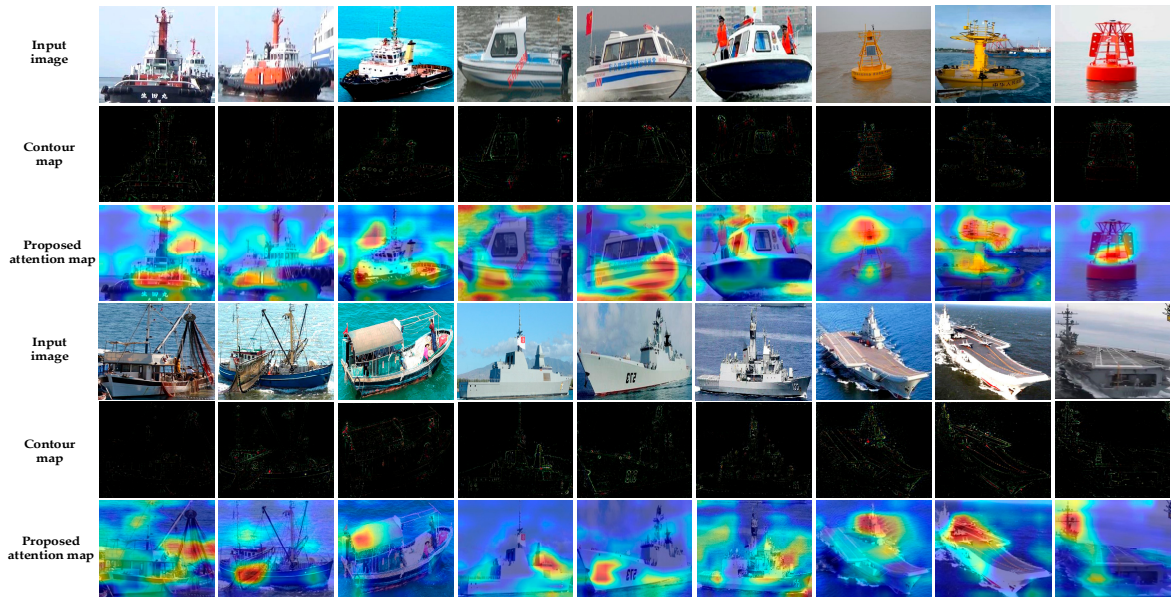


**Figure 9.** Examples of contour map and attention map.

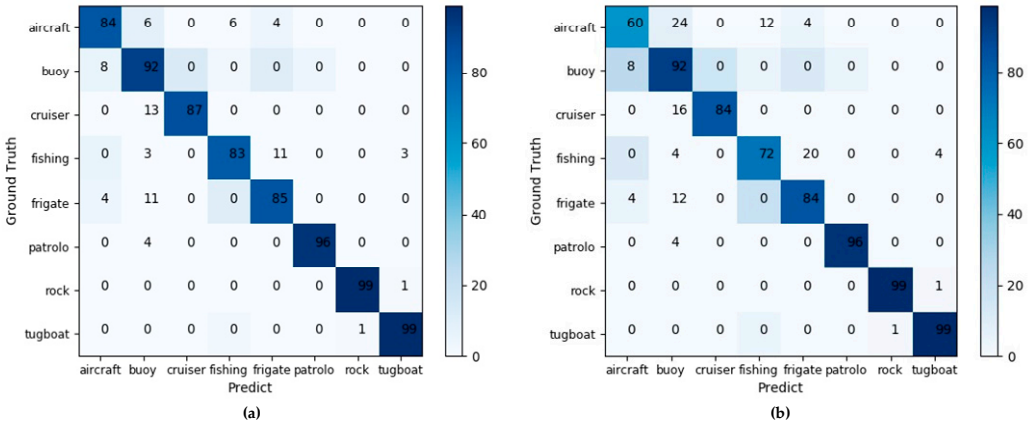


**Figure 10.** Confusion matrixes of different models: (**a**) our proposed model; (**b**) MAMC.

For those that are easier to identify, like rocks and tugboats, there is no difference between the above two models, which can be obtained in Figure 10. However, when identifying aircrafts and fishing boats, our proposed model performs much better than MAMC. Six out of 100 aircraft carriers are misidentified as buoys, and 6 aircrafts are misidentified as fishing boat in our model, while 24 out of 100 aircraft carriers are misidentified as buoys, and 12 aircrafts are misidentified as fishing boat in MAMC. The recognition error is due to the small number of trained samples of the aircraft carrier and the visual similarity between the aircraft carrier and the buoy. The experimental results show that our model is not only stable for general object recognition, but also better for objects with small visual differences than commonly used fine-grained classification algorithm.

### *4.3. Experiments and Results Analysis*

#### 4.3.1. Comparison Before and After Transfer Learning

Owing to the limited number of self-built target datasets, this study uses transfer learning to ensure that the network has sufficient feature expression capabilities. In order to confirm the effectiveness of transfer learning, the classification accuracy of using transfer learning and not using transfer learning are compared on the basis of adopting the model in this study. In addition, as mentioned

above, the target dataset is quite different from the source dataset and the number of the former is small. The parameters of networks should be transferred and fine-tuned. In order to determine the parameters of which layers can be fine-tuned to obtain a better classification effect, different ResNet-50 convolutional layers are frozen separately. ResNet-50 has a total of five convolutional blocks. The first block, the first two blocks, the first three blocks, and the first four blocks are sequentially frozen. Only the back convolutional blocks are trained. To ensure the stability and accuracy of the results, a single test is not enough. We introduce Monte Carlo cross validation, that is, each time the dataset is divided, the images are shuffled, and then randomly selected (not replaced after the extraction). Moreover, it is necessary to make the data distribution of the training set and the test set consistent. Since the number of each type of targets in the dataset is different, in the process of dividing the training set and the test set, we randomly select and divide in sub-categories according to the ratio of 8:2 instead of selecting from all images. This process is repeated three times. The classification accuracy rate is the average of these three results. Results are shown in Table 1.

**Table 1.** Experiment results of different use of transfer learning on self-built dataset.

| Using Methods | Accuracy/(%) |
|---|---|
| No transfer learning | 81.3 |
| Freeze the first block | 87.5 |
| Freeze the first two blocks | 89.0 |
| Freeze the first three blocks | **90.5** |
| Freeze the first four blocks | 89.3 |
| Freeze all convolutional blocks | 87.9 |

The bold numbers above indicate the method with the highest percentage of correct identifications. As shown in Table 1, the classification accuracy of freezing the first three blocks is the best among all transfer learning methods, reaching 90.5%, which is significantly higher than that of not using transfer learning. In addition, the results show that using different parameters of networks has an influence on the recognition accuracy. As the frozen blocks increases, so does the recognition accuracy until a peak at the first three blocks. After that, the recognition accuracy begins to decline by a small degree. When all convolutional blocks are frozen, there is only the classification layer that is trained. The accuracy reaches 87.9%. Some of the results in the experiment are slightly smaller than the value of a single test, but they are not much different. Most of them converge well to the values given in the table, which shows that our model is relatively stable, and the error fluctuation is relatively small in the case of randomly extracting data. It has been verified by experiments that the shallow layer of convolutional layers can usually extract low-level simple features, such as some edge features, while the deep layer of convolutional layers can extract more complex features, which is beneficial for the recognition of similar objects. This could explain why the accuracy of the module has a fluctuation with the increase of the frozen layers. As for the module that we propose, shallow layers could extract low-level features, like contour and color, while deep layers need to learn some specific features. Therefore, it could have a better effect when shallow layers are frozen and deep layers are trained.

In addition, after randomly dividing the training set and test set each time, we record the change of loss in training process and compare the size and the speed of convergence when using transfer learning and not using transfer learning in Figure 11a. The accuracy of freezing different blocks in three sub-tests is also depicted in Figure 11b.

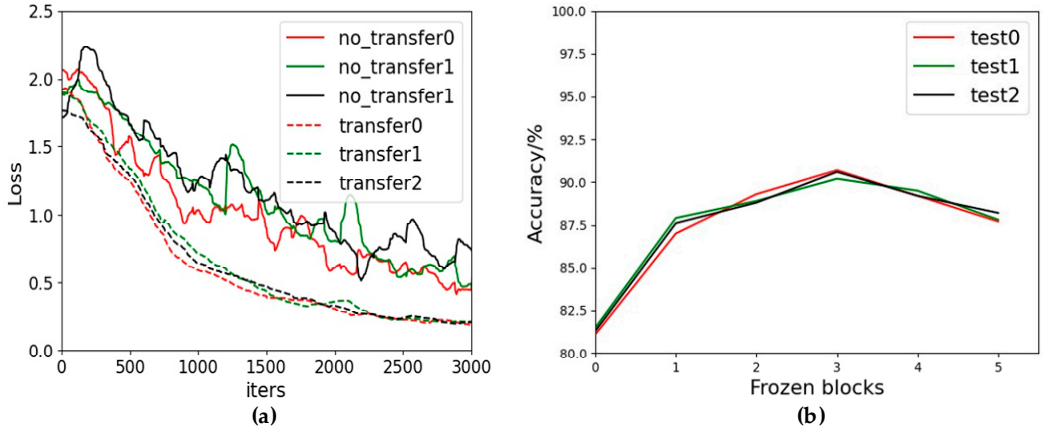


**Figure 11.** Changes of loss and accuracy in transfer learning: (**a**) Comparison of the loss of using transfer learning and no transfer learning in three sub-tests; (**b**) comparison of the accuracy of freezing different convolutional blocks in three sub-tests.

It could be drawn from the figures, compared with the method of not using transfer learning, the loss in model using transfer learning converges faster and smaller. Despite some fluctuations, the results of the three sub-tests of the same method are relatively similar, which means proposed model is stable and balanced. We calculate the standard deviation of losses during three sub-tests when using transfer learning and not using transfer learning. When using transfer learning, the standard deviation of the test is 0.022991, and the standard deviation is 0.103226 when not using transfer learning. This shows that the results of the sub-tests are very similar, and it also reflects the stability of the proposed model, especially after using transfer learning. After using the method of transfer learning, when the number of iterations reaches 3000, the loss is below 0.1. However, when proposed model does not use transfer learning, loss converges slow and has more volatilities, which means transfer learning has an important influence on the training process of the model. Moreover, the accuracy rate will not be consistent with the increase in the number of frozen layers. When the first three blocks are frozen, the accuracy rate reach its peak.

#### 4.3.2. Comparison of Different Sample Sizes

A stable system can continuously learn new knowledge from new samples and retain most of the knowledge already learned. In our dataset, the same type of surface targets follows the same distribution. As the scale of the dataset continues to expand, the learned features should become more abundant, and the accuracy rate will become more accurate. In order to prove the stability and robustness of the proposed model, we use 100, 200, 500, and 1000 images as the training set. The training set is randomly selected. To avoid chance, the process of selecting different training sets is repeated three times. Take the average of these three results under different sample scales, and record it.

From figures in the above, under different data scales, the loss and accuracy in the training process differ. In Figure 12a, the loss becomes smaller, and the convergence becomes faster with the increase of the data size. In the case of a training set of 1000, loss basically reaches stability when iterations are 1500, while the remaining three curves are still falling, fluctuating, and do not converge. When the training set size is 100 and 200, the difference in loss is not very large. In Figure 12b, as the size of the training set increases, the accuracy is higher, the number of iterations required for convergence is less, and there are fewer fluctuations. The above changes in loss and accuracy are the average of three sub-tests, and the experimental results are convincing. When the data size is 100 or 200, because the learned features are limited, the proposed model is not stable, which is manifested in the inability to converge and a lot of fluctuations. When the data size reaches 1000 and above, the trained model

converges much faster and can reach a higher accuracy. These also reflect the stability of the proposed model in another aspect.

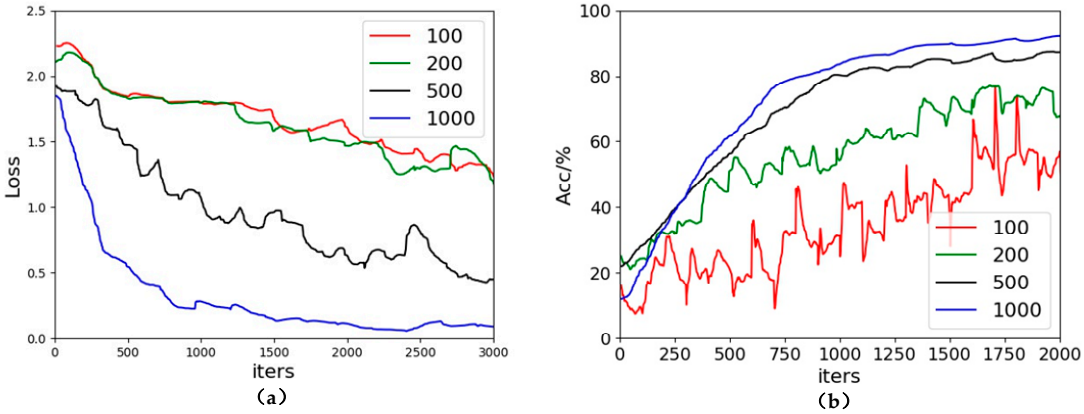


**Figure 12.** Average results of five tests under different data scales: (**a**) The loss during training; (**b**) the accuracy during training.

#### 4.3.3. Comparison with Other Weak Supervision Methods

In order to evaluate the method proposed in this paper, the existing popular fine-grained classification methods are used for comparison. The parameter setting and optimization methods of other algorithms are consistent with this model. To ensure the effectiveness and practicability of the algorithm, experiments are not only conducted on our self-built dataset, but also tested on public datasets(Stanford Cars [28], FGVC-aircraft [29]). The reason that we choose cars and aircrafts as comparative datasets is that the appearance of cars and aircrafts is more similar to that of water surface targets than other objects.

The multi-attention residual model used in this study belongs to the weakly supervised classification models. It does not require additional labelling information such as bounding boxes. Both strongly supervised classification algorithm and weakly supervised classification algorithm are conducted to compare with our model, including R-CNN, FCAN [30], B-CNN [31], RA-CNN, MA-CNN [32], MAMC, DFL-CNN [33], and TASN.

In addition, channel attention-ResNet (CA-RES) represents a residual network using only the channel attention module, and Spatial Attention-ResNet (SA-RES) represents a residual network using only the spatial attention module. To ensure the stability and validity of the experimental results, we adopt a cross-validation method and use data repeatedly. The accuracy rate obtained is the average of multiple test results. The name, basic framework and accuracy of each model are recorded. The experimental results are shown in the following table.

It can be seen from the results in Table 2 that the classification effect of the proposed method on self-built surface target dataset and aircraft dataset is better than other methods listed in this article, and the accuracy rates of 91.5% and 92.1% are achieved respectively. The classification accuracy of the proposed model on Stanford Cars is the same as the MAMC model, reaching 93%, which is superior to most fine-grained classification algorithms. The results show that the recognition accuracy of the space attention module is higher on surface targets and aircrafts. For car recognition, channel attention module has a better recognition effect.

**Table 2.** Comparative accuracy results of our model with other mainstream weakly supervised methods on different datasets (our self-built dataset, Stanford Cars, Aircraft).

| Approach | Backbone | Accuracy/(%) | | |
|---|---|---|---|---|
| | | Self-Built | Cars | Aircrafts |
| R-CNN | AlexNet | 81.8 | 88.4 | 86.6 |
| FCAN | VGG-16 | 82.5 | 89.1 | / |
| B-CNN | VGG-M+VGG-D | 83.3 | 91.3 | 84.1 |
| RA-CNN | VGG-19 × 3 | / | 92.5 | 88.2 |
| MA-CNN | VGG-19 × 3 | 86.1 | 92.8 | 89.9 |
| MAMC | ResNet-101 | 88.3 | 93.0 | / |
| DFL-CNN | ResNet-50 | / | 93.1 | 91.7 |
| TASN | ResNet-50 | 89.3 | **93.8** | / |
| CA-RES | ResNet50 | 87.6 | 92.5 | 90.6 |
| SA-RES | ResNet50 | 88.2 | 91.6 | 89.5 |
| Proposed model | ResNet50 × 2 | **90.5** | 93.0 | **92.1** |

The bold numbers above indicate the highest accuracy on the corresponding dataset when using the leftmost method. On one hand, for other fine-grained recognition methods, the classification accuracy on three datasets of Stanford Cars, FGVC-aircrafts and self-built surface target dataset is from high to low. On the other hand, for the multi-attention residual module that we proposed, the classification accuracy on three datasets of self-built surface targets, FGVC-aircrafts and Stanford Cars is from high to low, which is the opposite of the effect of the above fine-grained classification algorithms. This contrast might be caused by different scales of each dataset. Our proposed module is more suitable to small-scale dataset, since the method of transfer learning is used to improve the disability of limited data. It should be noticed that classification effect of methods adopting ResNet as a basis is usually superior to methods using other backbones, such as AlexNet and VGGNet.

## 5. Discussion

Based on the existing deep learning network and attention mechanism, this paper proposes a multi-attention residual module for the classification of surface targets. In addition, we add a channel attention module and a spatial attention module between the residual units to focus on characteristics and parts that are helpful for classification. This module could reduce the influence of changing the attitude and scale of surface targets. In addition, a dataset of surface targets has been established. Transfer learning and N-pair loss are used to increase the ability of the neural network to learn characteristics and enlarge differences between different categories. Finally, comparative experiments are conducted on the self-built dataset, and the results show that this method can effectively identify surface targets and is superior to most mainstream weakly supervised classification algorithms. Owing to the limited scale of the self-built dataset, it is necessary to continuously update and maintain the dataset to improve the quantity and quality of that. It should be pointed out that the feature dimension of this model is relatively large, which consumes a lot of computing resources. When this algorithm is integrated in terminals such as drones, the endurance capabilities of those are insufficient. Therefore, the features dimensionality and the redundant structure of the network should be reduced without loss of classification accuracy. Moreover, transfer learning can be further explored. A more suitable migration method could be chosen through a large number of experiments. This is the future scope.

**Author Contributions:** Conceptualization, R.G. and B.S.; methodology, R.G.; software, B.S.; validation, X.Q., Z.Z. and S.S.; formal analysis, P.W.; investigation, R.G.; resources, B.S.; data curation, X.Q.; writing—original draft preparation, R.G.; writing—review and editing, B.C.; visualization, R.G.; supervision, R.G.; project administration, B.C.; All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

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

## References


1. He, J.; Guo, Y.; Yuan, H. Ship Target Automatic Detection Based on Hypercomplex Flourier Transform Saliency Model in High Spatial Resolution Remote-Sensing Images. *Sensors* **2020**, *20*, 2536. [CrossRef] [PubMed]
2. Rajasekaran, S.; Raj, R.A. Image recognition using analog-ART1 architecture augmented with moment-based feature extractor. *Neurocomputing* **2004**, *56*, 61–77. [CrossRef]
3. Susaki, J. Knowledge-Based Modeling of Buildings in Dense Urban Areas by Combining Airborne LiDAR Data and Aerial Images. *Remote. Sens.* **2013**, *5*, 5944–5968. [CrossRef]
4. Chang, K.; Ghosh, J. Three-dimensional model-based object recognition and pose estimation using probabilistic principal surfaces. *Electron. Imaging* **2000**, *3962*, 192–204.
5. Khellal, A.; Ma, H.-B.; Fei, Q. Convolutional Neural Network Based on Extreme Learning Machine for Maritime Ships Recognition in Infrared Images. *Sensors* **2018**, *18*, 1490. [CrossRef] [PubMed]
6. Lin, C.-J.; Lin, C.-H.; Sun, C.-C.; Wang, S.-H. Evolutionary-Fuzzy-Integral-Based Convolutional Neural Networks for Facial Image Classification. *Electronics* **2019**, *8*, 997. [CrossRef]
7. Guo, W.; Xia, X.; Wang, X. A remote sensing ship recognition method of entropy-based hierarchical discriminant regression. *Optik* **2015**, *126*, 2300–2307. [CrossRef]
8. Alzubaidi, L.; Al-Shamma, O.; Fadhel, M.A.; Farhan, L.; Zhang, J.; Duan, Y. Optimizing the Performance of Breast Cancer Classification by Employing the Same Domain Transfer Learning from Hybrid Deep Convolutional Neural Network Model. *Electronics* **2020**, *9*, 445. [CrossRef]
9. Hua, Y.; Yang, Y.; Du, J. Deep Multi-Modal Metric Learning with Multi-Scale Correlation for Image-Text Retrieval. *Electronics* **2020**, *9*, 466. [CrossRef]
10. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
11. LeCun, Y.; Boser, B.; Denker, J.S.; Henderson, D.; Howard, R.E.; Hubbard, W.; Jackel, L.D. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Comput.* **1989**, *1*, 541–551. [CrossRef]
12. Krizhevsky, A.; Sutskever, I.; Hinton, G. ImageNet Classification with Deep Convolutional Neural Networks. *Neural Inf. Process. Syst.* **2012**, *25*. [CrossRef]
13. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. *arXiv* **2014**, arXiv:1409.4842.
14. 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 (CVPR 2016), Las Vegas, NV, USA, 26 June–1 July 2016.
15. Ge, Z.; McCool, C.; Sanderson, C.; Corke, P. Subset feature learning for fine-grained category classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR 2015), Boston, MA, USA, 8–10 June 2015.
16. Ge, Z.; Bewley, A.; McCool, C.; Corke, P.; Upcroft, B.; Sanderson, C. Fine-grained classification via mixture of deep convolutional neural networks. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV 2016), Lake Placid, NY, USA, 7–10 March 2016.
17. Lin, T.-Y.; Roychowdhury, A.; Maji, S. Bilinear CNN Models for Fine-Grained Visual Recognition. In Proceedings of the IEEE International Conference on Computer Vision (ICCV 2015), Santiago, Chile, 7–13 December 2015.
18. Zhang, N.; Donahue, J.; Girshick, R.; Darrell, T. Part-Based R-CNNs for Fine-Grained Category Detection. In *Lecture Notes in Computer Science, Proceedings of the European Conference on Computer Vision (ECCV 2014), Zurich, Switzerland, 6–12 September 2014*; Springer: Berlin, Germany, 2014.
19. Zeiler, M.D.; Fergus, R. Visualizing and Understanding Convolutional Networks. In Proceedings of European Conference on Computer Vision. In *Lecture Notes in Computer Science, Proceedings of the European Conference on Computer Vision (ECCV 2014), Zurich, Switzerland, 6–12 September 2014*; Springer: Berlin, Germany, 2014.
20. Branson, S.; Van Horn, G.; Perona, P.; Belongie, S. Improved Bird Species Recognition Using Pose Normalized Deep Convolutional Nets. In Proceedings of the British Machine Vision Conference (BMVC 2014), Nottingham, UK, 1–5 September 2014.

21. Bentaieb, A.; Hamarneh, G. Topology Aware Fully Convolutional Networks for Histology Gland Segmentation. In *Lecture Notes in Computer Science, Proceedings of the Medical Image Computing and Computer-Assisted Intervention (MICCAI 2016), Athens, Greece, 17–21 October 2016*; Springer: Berlin, Germany, 2016.
22. He, K.; Georgia, G.; Piotr, D.; Ross, G. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision (ICCV 2017), Venice, Italy, 22–29 October 2017.
23. Xiaohua, W.; Muzi, P.; Lijuan, P.; Hu, M.; Chunhua, J.; Fuji, R. Two-level attention with two-stage multi-task learning for facial emotion recognition. *J. Vis. Commun. Image Represent.* **2019**, *62*, 217–225. [CrossRef]
24. Fu, J.; Zheng, H.; Mei, T. Look Closer to See Better: Recurrent Attention Convolutional Neural Network for Fine-Grained Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017), Puerto Rico, PR, USA, 24–30 June 2017.
25. Chen, X.; Gupta, A. An Implementation of Faster RCNN with Study for Region Sampling. In *Lecture Notes in Computer Science, Proceedings of the Medical Image Computing and Computer-Assisted Intervention (MICCAI 2016), Athens, Greece, 17–21 October 2016*; Springer: Berlin, Germany, 2016.
26. Zheng, H.; Fu, J.; Zha, Z.-J.; Luo, J. Looking for the Devil in the Details: Learning Trilinear Attention Sampling Network for Fine-Grained Image Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2019), Long Beach, CA, USA, 16–20 June 2019.
27. Sun, M.; Yuan, Y.; Zhou, F.; Ding, E. Multi-Attention Multi-Class Constraint for Fine-grained Image Recognition. In Proceedings of the European Conference on Computer Vision (ECCV 2018), Munich, Germany, 8−14 September 2018.
28. Krause, J.; Stark, M.; Deng, J.; Fei-Fei, L. 3D Object Representations for Fine-Grained Categorization. In Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCV 2013), Sydney, Australia, 3−6 December 2013.
29. Maji, S.; Rahtu, E.; Kannala, J.; Blaschko, M.; Vedaldi, A. Fine-Grained Visual Classification of Aircraft. Computer Vision and Pattern Recognition. *arXiv* **2013**, arXiv:1306.5151.
30. Shelhamer, E.; Long, J.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. *IEEE Trans. Pattern Anal. Mach. Intel.* **2017**, *39*, 640–651. [CrossRef]
31. Lin, T.Y.; Roychowdhury, A.; Maji, S. Bilinear CNNs for Fine-grained Visual Recognition. *arXiv* **2015**, arXiv:1504.07889.
32. Zheng, H.; Fu, J.; Mei, T.; Luo, J. Learning Multi-attention Convolutional Neural Network for Fine-Grained Image Recognition. In Proceedings of the IEEE International Conference on Computer Vision (ICCV 2017), Venice, Italy, 22–29 October 2017.
33. Wang, Y.; Morariu, V.I.; Davis, L.S. Learning a Discriminative Filter Bank within a CNN for Fine-grained Recognition. *arXiv* **2016**, arXiv:1611.09932.