symmetry

Article

Research on Spider Recognition Technology Based on Transfer Learning and Attention Mechanism

Jianming Wang [1,2,*], Qiyu Chen [1] and Chenyang Shi [1]

[1] School of Mathematics and Computer Science, Dali University, Dali 671003, China; qiyu_chen@outlook.com (Q.C.); scy@stu.dali.edu.cn (C.S.)
[2] Yunnan Provincial Key Laboratory of Entomological Biopharmaceutical R&D, Dali University, Dali 671000, China
* Correspondence: wjm@dali.edu.cn

Abstract: Methods such as transfer learning and attention mechanisms play an important role in small-sample image classification tasks. However, the conventional transfer method retains too much prior knowledge of the source domain and cannot learn the feature information of the target domain well. At the same time, it is difficult for the neural network model to find discriminative features and locate key feature regions, and it is easily interfered with by information such as complex backgrounds. Spiders usually appear symmetrical, but they are not perfectly symmetrical. How to accurately classify spider images depends on how to make the model focus on the key features for recognizing spiders in these symmetrical and asymmetrical regions. In view of the above problems, in this paper, we propose ECSM-ResNet-50, a model for small-sample spider image classification. The model fuses channel and spatial information and pays attention to the correlation between different locations in the input data. The Efficient Channel Attention (ECA) mechanism and the spatial attention mechanism were added to the model, and the self-attention mechanism was added to the end of the model. ECSM-ResNet-50 was constructed and trained on a small-sample spider data set (SPIDER9-IMAGE) using a layer-by-layer fine-tuning transfer learning strategy. Compared with ResNet-50, ECSM-ResNet-50 improved the average accuracy of nine species of spider recognition by 1.57% to 90.25%. This study contributes to the field of small-sample image recognition.

Keywords: deep learning; spider identification; transfer learning; fine-tuning; attention mechanism

check for updates

Citation: Wang, J.; Chen, Q.; Shi, C. Research on Spider Recognition Technology Based on Transfer Learning and Attention Mechanism. *Symmetry* **2023**, *15*, 1727. https://doi.org/10.3390/sym15091727

Academic Editor: Jian-Qiang Wang

Received: 1 August 2023
Revised: 1 September 2023
Accepted: 6 September 2023
Published: 8 September 2023

1. Introduction

At present, deep learning technology is widely used in the field of image recognition and has achieved many results [1]. Compared with the traditional manual screening method, it can efficiently recognize and classify images, reduce the recognition error rate, and shorten the recognition and classification time. The effect of deep learning is heavily dependent on the quantity and quality of data [2]. However, in the case of a wide range of species distribution and a limited image collection environment, it is difficult for researchers to obtain a large number of samples, or the cost of obtaining samples is too high. Therefore, how to complete the image classification task with high quality in a small-sample data set has become a hot topic and a difficult point in research. For images with high feature similarity between species, it is difficult to classify even artificially, such as the pattern and shape of the abdomen of spiders, and feature information such as antennae and limb structure. It is difficult for the neural network model to find features that can be used to discriminate between different species with limited data. At the same time, complex background information will also interfere with the model discrimination of foreground and background information [3].

The emergence of transfer learning mitigates the problems of overfitting and low model recognition often associated with small-sample learning and aids in the training of new tasks by leveraging the powerful feature extraction capabilities of models pre-trained on large data sets. Focusing on the unsupervised domain adaptation problem, Wang [4] introduced a Gaussian-guided latent alignment method, which verified that the proposed method has good knowledge transferability. Kalvakolanu [5] applied transfer learning and data augmentation methods to ResNet-34 and ResNet-50 models, and the accuracy was significantly improved compared to the de novo training method. You [6] proposed a framework called Co-Tuning to learn the relationship between source and target categories from pre-trained models and calibrated predictions in two steps. At present, most transfer methods freeze all layers of the model and only train the fully connected layer. However, due to the different feature dimensions and categories extracted by different layers of the model, the deeper layers mainly contain classification information, and this information varies depending on the data set. Traditional transfer fine-tuning methods can easily make the model unable to fully learn the feature information of the target domain data set, so it is necessary to transfer the layers of specific tasks in a targeted manner.

When the convolutional neural network extracts features, it is easily disturbed by complex background information and cannot pay good attention to key feature information. The emergence of an attention mechanism enables the neural network model to focus on the local information of the image and improve its ability to discriminate key features. The attention mechanism can make the model pay more attention to the key features of the image, thus reducing the interference of irrelevant information and enhancing the utilization of features. For different species of spiders, the key features for their accurate classification may exist in the symmetrical or asymmetrical regions of the spider. For example, in the symmetrical region of spiders, the cephalothorax and abdomen of Thomisidae spiders are short and wide, and the eyes of jumping spiders are divided into three columns and vary in size. In asymmetrical regions, *Cyrtophora* species have abdomens with a distinct cloudy pattern. Models may pay more attention to these specific features.

At present, attention mechanisms such as SENet (Squeeze-and-Excitation Networks) [7], CBAM (Convolutional Block Attention Module) [8], and self-attention [9] have been widely introduced into deep models, but these attention additions introduce new parameters that are difficult to train adequately with small-sample data and even have negative effects. In the process of model training, because the data set is too small and the model parameters are too large, the model cannot learn enough information and features from the data, so it overfits. In summary, the research in this paper focuses on solving the problem of insufficient model feature extraction and easy overfitting in small sample image classification. In this paper, the model is fine-tuning using transfer learning to alleviate the overfitting problem by utilizing the generic features already learned from the pre-trained model. The attention mechanism is improved in order to use the information aggregation ability of the attention mechanism to enhance the feature extraction ability of the model. It is demonstrated experimentally that this approach can solve the above problem in small sample image classification.

The principal contributions of this paper are as follows: (1) The ECSM module was constructed by combining the Efficient Channel Attention (ECA) mechanism with the spatial attention mechanism and adding the self-attention mechanism behind it. This module fuses channel domain and spatial domain information to enhance local information aggregation and has global information aggregation capability. (2) The ECSM module was added to the end of the ResNet-50 model. The ECSM-ResNet-50 model was constructed. This module enhances the local and global information aggregation capability of the ResNet-50 model. (3) The model is fine-tuning using transfer learning methods. We found that the best fine-tuning strategy was to freeze the underlying residual blocks Block 1 and Block 2 of the ResNet-50 model and train only Block 3, Block 4, and the fully connected layer. (4) The performance of ECSM-ResNet-50 was successfully validated in the spider data set SPIDER9-IMAGE. The experimental results show that the proposed ECSM-ResNet-50 was more effective compared to ResNet-50.

2. Materials

2.1. Data Sources

The small-sample spider data set includes spiders with little feature variation among different species, and it is difficult to identify key features. Therefore, it is difficult to classify spiders at a fine granularity. Moreover, spiders have medical and ecological value and play an important role in drug research and development and ecological balance. Therefore, spiders were chosen as the research object of this experiment. This study is based on the live and specimen image data from the Yunnan Provincial Key Laboratory of Entomological Biopharmaceutical R&D, Dali University, combined with field collection, network collection, and other methods to supplement the data, and was manually classified, identified, and annotated by experts. The spider data set SPIDER9-IMAGE was constructed to identify nine species of spiders with a small sample size.

The division and settings of the spider data set SPIDER9-IMAGE are shown in Table 1. The data set includes nine species of spiders, including Thomisidae, Theraphosidae, *Lycosa*, *Araneidae*, *Missulena*, Araneidae, *Trichonephila clavata*, Philodromidae, *Plator*, and *Cyrtophora*, and divides the training set and test set according to the ratio of 7:3.

Table 1. Data Set.

Spider Species	Number of Training Sets	Number of Test Sets
Thomisidae	176	74
Theraphosidae	174	74
Lycosa	170	74
Miussulena	176	74
Araneidae	176	74
Trichonephila clavata	176	74
Philodromidae	180	76
Plator	226	96
Cyrtophora	164	70

2.2. Data Processing

The data was resized, shuffled, and normalized in the experiment. Resizing is done to adjust the image size to match different models. In this case, the image size is resized to the standard input size of the ResNet-50 model, which is 224 × 224 pixels. Shuffling is performed to randomly reorganize the training image data, thereby preventing the model from overfitting and enhancing its generalization ability. Traditional normalization is applied to control the pixel values of the image, which originally range from 0 to 255, to a normalized range of 0 to 1. This prevents issues such as gradient explosion or vanishing during training. When using pre-trained model parameters, the data should be preprocessed according to the data preprocessing method used for pre-training the model. The ImageNet data set is large and contains a considerable amount of animal image data, making it widely used for model pre-training. Since this experiment focuses on spiders as the research object, the ImageNet data set can be used as the pre-training data set. Furthermore, the target domain data set should be preprocessed by subtracting the mean and variance of the ImageNet data set, which follows the data preprocessing method used for the source domain.

3. Methods

This paper mainly uses the deep residual network Resnet-50, pre-trained on ImageNet, to train on the spider data set SPIDER9-IMAGE. We experimented with different fine-tuning methods and froze different modules of the model to find the best model training strategy suitable for the small-sample data in this experiment. Finally, in order to combine with the transfer learning method, an improved and efficient mixed domain attention mechanism module is introduced at the end of the model so that the model can pay better attention to the key feature information in the image, accurately classify the foreground from the background, reduce the interference of complex background information, and improve model accuracy.

3.1. Transfer Learning

Traditional machine learning tasks require massive amounts of data as support, and the quality and abundance of data determine the upper limit of a neural network model. However, in many cases, some specific image data are difficult to obtain, which brings great difficulties and challenges to the training of neural networks. The emergence of transfer learning [10] has greatly improved the accuracy of small-sample learning and has alleviated the overfitting problem caused by insufficient data volume. Transfer learning is the process of transferring knowledge learned from a task in the source domain to a task in the target domain to improve the predictive performance of the model in the target domain task. Domain and task [11] are two important concepts for transfer learning. The difference in the probability distribution between the source domain and the target domain is very important. It is necessary to select a situation where the target domain and the source domain have similar data distributions for transfer. This transfer learning method can achieve twice the result with half the effort. Otherwise, if the distribution difference between the source domain and the target domain is too large, it is likely to result in negative transfer [12]. ImageNet contains a large amount of animal, insect, and spider data, which are more suitable for the fine-grained classification of spider species in this study. Therefore, ImageNet was chosen as the source domain.

3.1.1. Pre-Training and Fine-Tuning

In transfer learning, there is a transfer based on shared parameters, the main method of which is pre-training and fine-tuning [13].

The common approach to fine-tuning is to remove the fully connected layer used by the source model for classification, rebuild a fully connected layer for a specific classification task, and then load the matched parameters from the source model into the new model. Generally, only the fully connected layer is trained, and this approach can achieve a high accuracy on the classification task with a new data set. However, since the parameters of the extracted feature layer of the model are fixed, relying only on the feature extraction ability of the source model leads to insufficient generalization ability of the model in the case of very high sample similarity in fine-grained recognition.

3.1.2. Layer-by-Layer Fine-Tuning of Transfer Learning Strategies

Traditional fine-tuning methods have limitations. In this paper, we design a transfer learning strategy for training the ResNet-50 model using three different fine-tuning methods for small-sample spider data. (1) Freeze all convolutional layer parameters to train only the fully connected layer. (2) Train from scratch using pre-trained parameters as initial parameters. (3) Freeze the bottom parameters of the model and train only the parameters of the top layer. The third fine-tuning method is subdivided into three parts: (3a) Freeze the input layer and residual block Block 1 and train only residual blocks Block 2, 3, 4, and the top layer of the model. (3b) Freeze the input layer and residual blocks Block 1 and 2, and train only residual blocks Block 3 and 4 and the top layer of the model. (3c) Freeze the input layer and Blocks 1, 2, and 3, and train only Block 4 and the model top layer. The model updates only the unfrozen parameters. Through comparative experiments, we find the most suitable transfer learning strategy for extracting feature information that can adequately fit the current data. The transfer learning strategy is shown in Table 2. The model fine-tuning and training process is shown in Figure 1.

Table 2. Transfer Learning Strategy.

Fine-Tuning Method Label	Pre-Trained Source Domain	Description of Fine-Tuning Method
1	ImageNet	Only train fully connected layers
2	ImageNet	Train from scratch using pre-trained parameters as initial parameters
3a	ImageNet	Only train the residual blocks Block 2, 3, 4
3b	ImageNet	Only train residual blocks Block 3, Block 4
3c	ImageNet	Only train residuals block Block 4

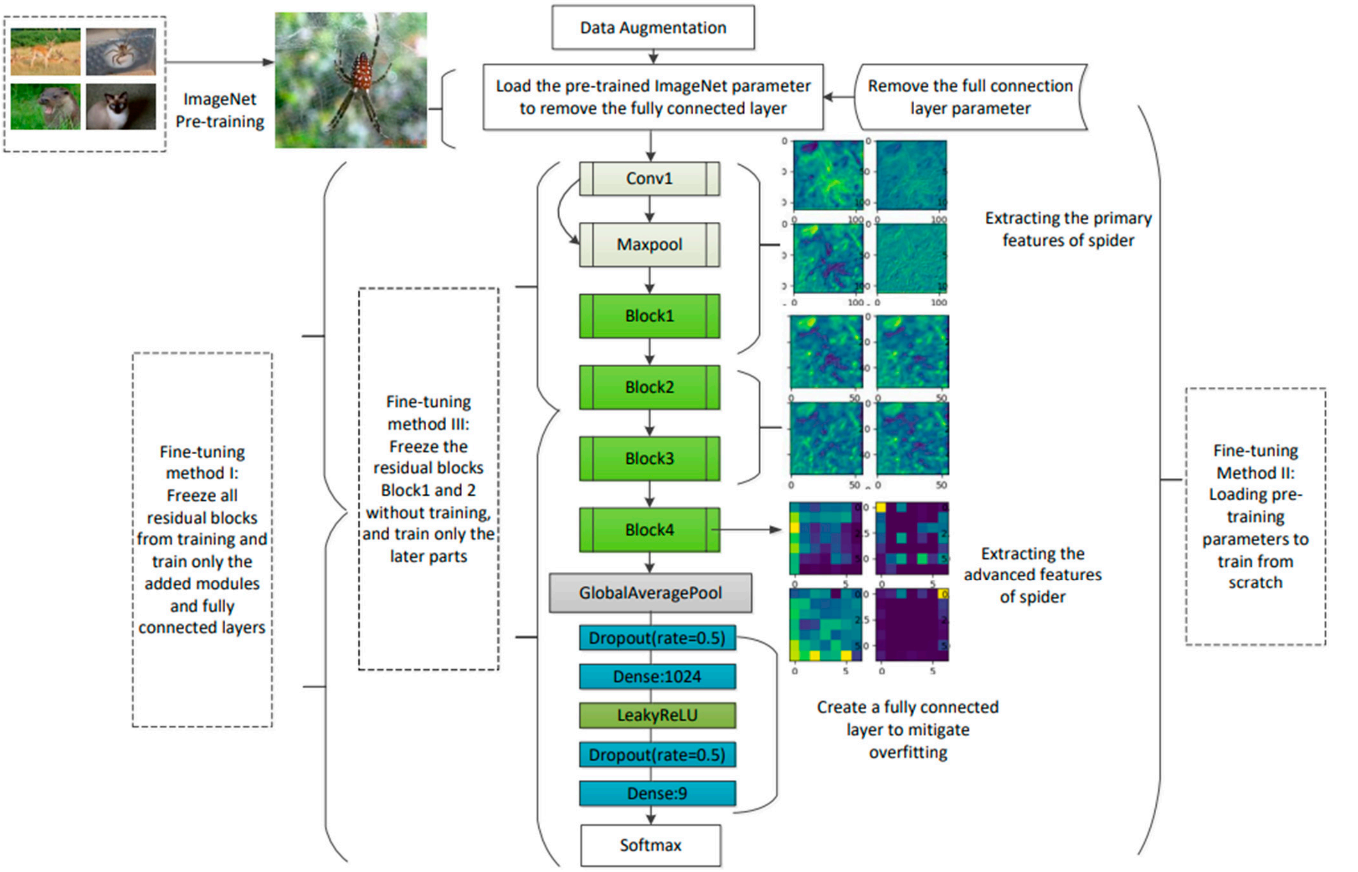

ImageNet Pre-training
Data Augmentation
Load the pre-trained ImageNet parameter to remove the fully connected layer
Remove the full connection layer parameter
Conv1
Maxpool
Block1
Block2
Block3
Block4
GlobalAveragePool
Dropout(rate=0.5)
Dense:1024
LeakyReLU
Dropout(rate=0.5)
Dense:9
Softmax
Extracting the primary features of spider
Extracting the advanced features of spider
Create a fully connected layer to mitigate overfitting
Fine-tuning method I: Freeze all residual blocks from training and train only the added modules and fully connected layers
Fine-tuning method III: Freeze the residual blocks Block1 and 2 without training, and train only the later parts
Fine-tuning Method II: Loading pre-training parameters to train from scratch

Figure 1. Layer-by-layer fine-tuning and training process of the ResNet-50 model.

In the case of an adequate data set, the layer-by-layer freezing tuning approach can be cumbersome, and it may be better to train the model from scratch. However, in the case of insufficient data sets, the transfer learning strategy with layer-by-layer fine-tuning can fully utilize the knowledge already learned by the pre-trained model in order to find the optimal solution that best fits the target domain.

3.2. Improved Efficient Mixed Domain Attention Mechanism Module

The attention mechanism [14] is a deep learning method similar to the human visual system. Its construction idea is to make the system pay more attention to the main information in the image, like a human would, and purposefully ignore the secondary information. For example, for a bird in the sky, the attention mechanism will pay more attention to the local information of the bird and ignore the background information of the blue sky. In computer vision, the attention mechanism calculates the weight distribution by scanning the image feature information and then superimposes the weight distribution on the original feature map to achieve the purpose of weighting local key information.

3.2.1. Channel Domain Attention Mechanism

Channel domain attention is a type of soft attention. In CNNs (convolutional neural networks), the image processing is initially RGB three channels, and then, under a series of convolution and pooling operations, the image feature information of each channel is extracted. The components of the image under different convolution operations can be used to find out the key information of the image and weigh the components on each channel. The higher the weight, the higher the correlation between this channel and the important information in the graph, and the more attention is paid to this channel.

Squeeze and Excitation is the main part of the channel domain attention module SE (Squeeze and Excitation) [15], shown in Figure 2. First, a transformation operation is performed on an input feature map X, which is mapped by a function to get the output O, which is actually a convolution operation shown in Equation (1):

$$O = G_{\text{ts}}(\text{X}) \tag{1}$$

where O represents the new feature map obtained by the transformation operation and function mapping of the feature map X, and X represents the input feature map.

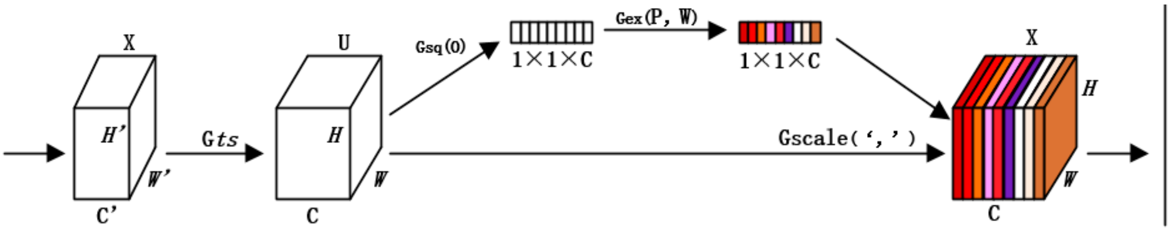


Figure 2. SE Module.

After that, the output P is obtained by pressing O through the squeeze operation Gts (Squeeze). Specifically, through global average pooling, the corresponding spatial information $H * W$ on each channel is squeezed into the corresponding channel and merged into one number; one pixel represents a channel and becomes a vector of dimension 1 $\times$ 1 $\times$ C. This operation squeezes the feature space to achieve a global low-dimensional embedding of O, that is, to have a global receptive field, as shown in Equation (2):

$$P = Gsq(O) = \frac{1}{H * W} \sum \sum_{j=1}^{w} O(i, j) \tag{2}$$

where P represents the result after performing global average pooling on feature O in spatial dimension $H * W$:

$$F = Gex(P, W) = \beta(g(p, fc)) = \beta(fc_2\alpha(fc_1 P)) \tag{3}$$

where F represents P through two fully connected layers, ReLu activation function and sigmoid activation function to get the final weight matrix, as shown in Equation (3).

The recalibration of the feature map is accomplished by applying the resulting weight matrix F to each channel of O, and the value at each position is multiplied by the corresponding channel weight.

ECANet [16] was presented at the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) in 2020. ECANet has made some optimizations to SENet. SENet, though with excellent performance, may cause a loss of spatial information by introducing global average pooling in the squeezing operation to embed spatial information in the channel. The introduction of two fully connected layers in the Excitation operation inevitably adds a large number of parameters. According to this drawback, ECANet proposes the concept of banded matrices, which use a one-dimensional convolution operation sliding top-down over the eigenvectors to achieve local interactions between channels. This operation significantly saves the number of parameters and improves performance.

The ECA attention module is shown in Figure 3. The relationship between the local cross-channel convolution kernel *K* and the number of characteristic channels *C*, is shown in Equations (4) and (5):

$$C = \phi(K) = 2^{(\gamma * k - b)} \tag{4}$$

$$K = \psi(C) = \left| \frac{\log_2(c)}{r} + \frac{b}{r} \right| \tag{5}$$

where *C* represents the number of feature channels and *K* represents the local cross-channel convolution kernel. In the ECANet paper, the authors used the mapping function Ø(*K*), that is, given a channel *C*, the corresponding *K* should be 2, with the default *r* = 2 and *b* = 1. This method realizes *K*-value adaptiveness.

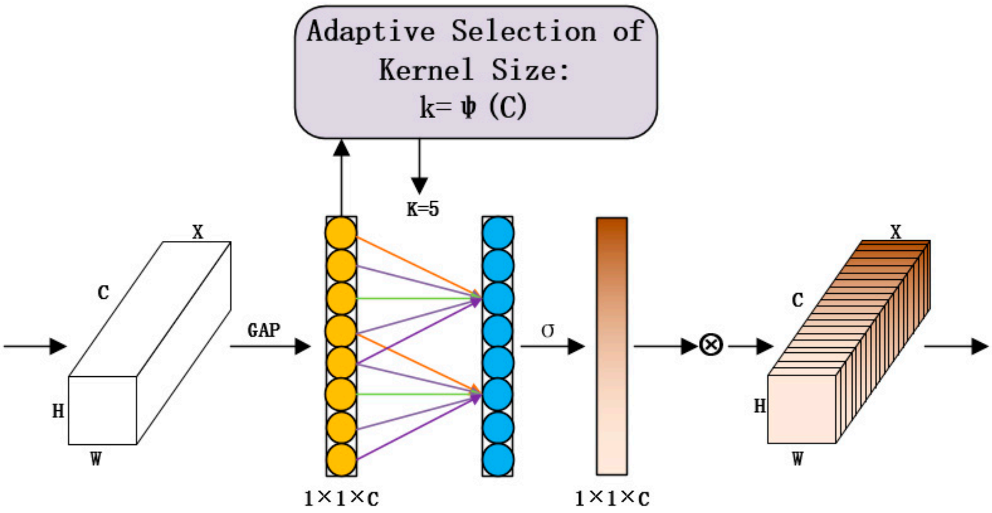


Figure 3. ECA Module.

3.2.2. Spatial Domain Attention Mechanism

The spatial domain attention mechanism is mainly used to suppress the channel domain feature information and highlight the spatial domain feature information by dimensionality reduction of the channel dimension information. Thus, the spatial transformation of the spatial domain information in the image is performed to extract the key information and map it to another space. Usually, a spatial mask of the same size as the feature map is first generated, and then each position is scored, thereby calculating the importance.

3.2.3. Convolutional Block Attention Mechanism (CBAM)

CBAM is a mixture of different features calibrated with multiple attentions. For channels, most of them are representations of feature abstractions, and for spatial, the positional information possessed is richer. For channel features, the input features are first compressed into the channel by global average pooling and global maximum pooling to compensate for the lack of spatial information in the embedding. After reducing the number of parameters by a perceptron, the feature vectors of the two channels are summed to fuse the information of the two channels. Then the activation function is used to obtain the weights, and the feature information calibrated by the channels is input into the spatial attention, and the spatial attention vector is obtained after a series of spatial changes, as shown in Figure 4.

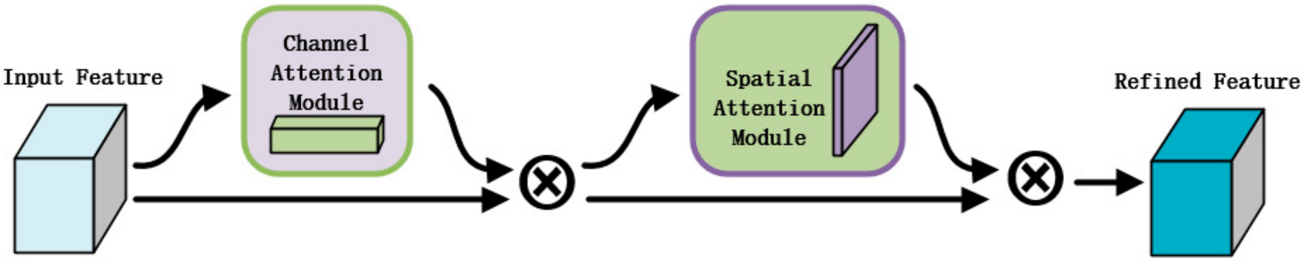


Figure 4. CBAM Module.

3.2.4. Self-Attention

Self-attention is an idea borrowed from Natural Language Processing(NLP), so names such as Query, Key, and Value are still retained. Figure 5 below is the basic structure of self-attention, and the input convolution feature maps are the basic trunk CNN (backbone) extracted feature map. The self-attention mechanism only involves the location attention module, but not the commonly used channel attention mechanism. The self-attention mechanism is one of the important components of the transformer and a variant of the attention mechanism. Compared with other attention mechanisms, the self-attention mechanism pays more attention to global feature information and the correlation between them. In the image domain, the self-attention mechanism learns the relationship between pixel points and pixel points at other locations, that is, capturing long-distance relationships.

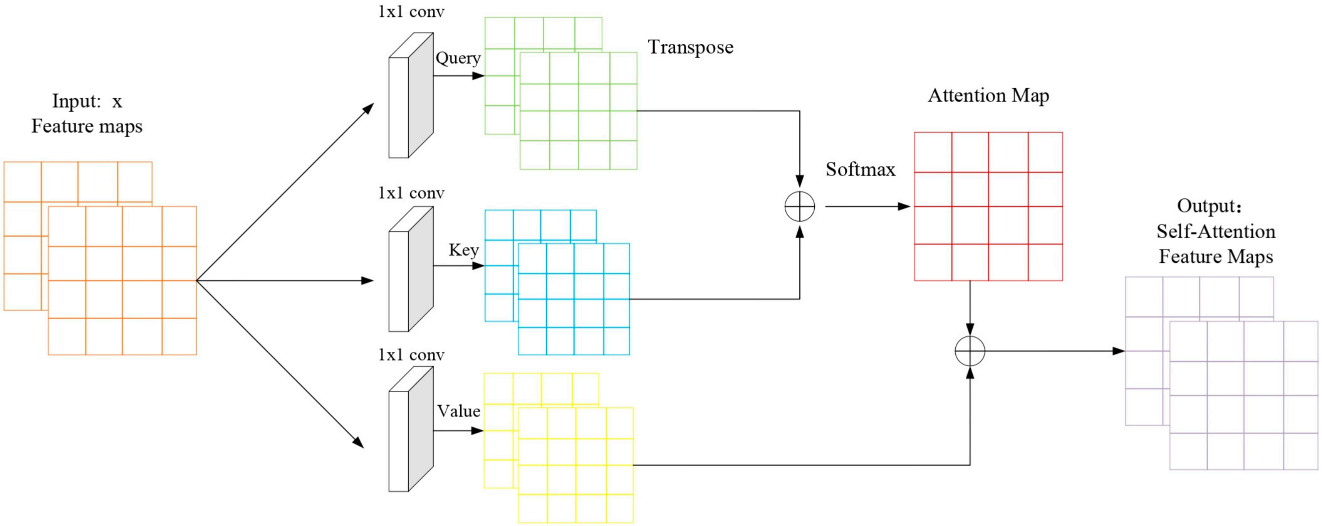


Figure 5. Self-Attention.

3.2.5. Efficient Channel Spatial Model (ECSM)

Based on the mixed domain attention mechanism, this study designed an improved Efficient Channel Spatial Model (ECSM) for the ResNet-50 model structure in order to efficiently extract the key feature information of the target under complex background interference. The attention module is to connect the ECA [16] mechanism and the spatial domain attention mechanism in a serial manner. However, although the efficient mixed-domain attention makes up for the lack of channel attention on spatial location information, it still only strengthens the local information in the image features. Therefore, self-attention mechanisms capable of capturing long-distance dependencies between information in an image are introduced in the hybrid domain. As mentioned above, the model calibrates the features three times to achieve the combination of the channel domain with the spatial domain, and the local information with the global information, which is the process of the improved Efficient Mixed-Domain Attention Mechanism ECSM (Efficient Channel Spatial Model) module, as shown in Figure 6.

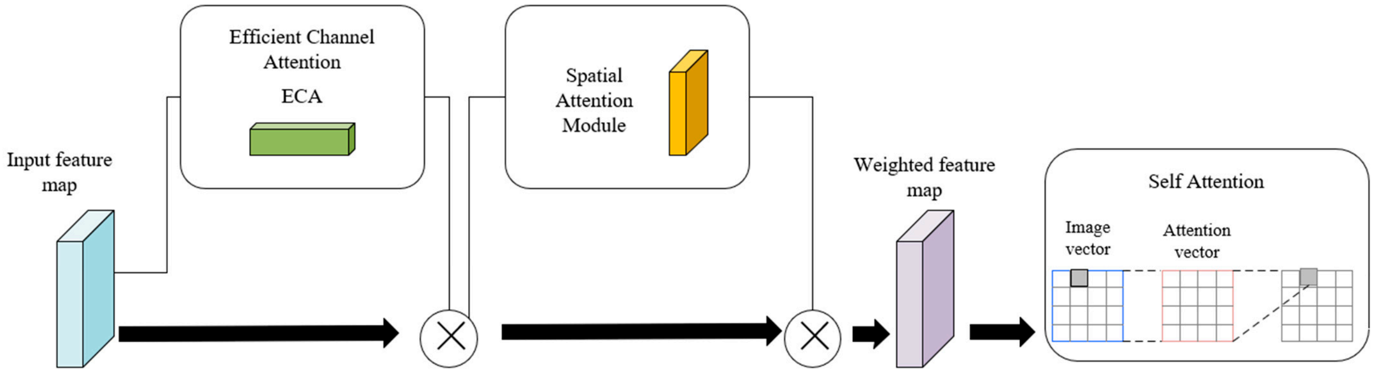


Figure 6. Efficient Channel Spatial Model.

3.3. *ECSM-ResNet-50*

The improved Efficient Mixed-Domain Attention Mechanism ECSM module is designed to combine the channel domain and spatial domain information to locally enhance the key information weights and then globally scan to capture the dependencies between the parts. In this work, the ResNet-50 model, which is widely used in the field of image recognition, is selected based on the server configuration conditions and the size of the data set. At the end of the residual network model ResNet-50, after the fourth Block, the ECSM attention module is introduced before the full connectivity layer to strongly extract the key features of the image. This change in model structure does not break the connection between the pre-trained parameters at the bottom of the model, and allows us to take advantage of the strong feature extraction capability of the pre-trained model even when the model structure changes due to the introduction of the attention mechanism.

3.3.1. ResNet-50

The ResNet [17] network is designed to address the problem of performance degradation as the network becomes deeper. The ResNet-50 residual network can be viewed as a stack of residual blocks. The difference between the residual network and the general network is that in the residual network, a portion of the data is first copied and accumulated before undergoing a nonlinear transformation. Ideally, as the depth of the network increases, the training error should gradually decrease. ResNet can effectively reduce the error caused by training in deep networks. The formula for learning features in the ResNet network learning features is shown in Equation (6):

$$x_{l+1} = x_l + F(x_l, W_l) \tag{6}$$

where x_{l+1} represents the features after residual block processing, x_l represents the direct mapping, and $F(x_l, W_l)$ represents the residual part, which generally consists of two to three convolution operations. In ResNet-50, this residual structure also becomes the bottleneck residual structure. This is because the feature dimension of the input image is compressed from 256 to 64 dimensions, and then it is boosted to 256 dimensions by a 1 × 1 convolutional kernel, forming a bottleneck structure with two thick sides and a thin middle. As shown in Figure 7.

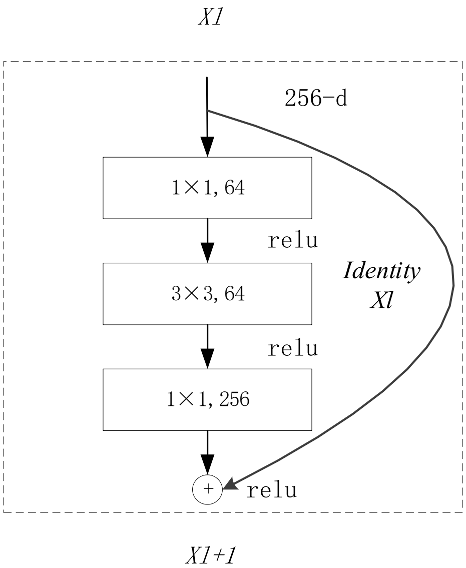


Figure 7. Residual block structure.

3.3.2. ECSM-ResNet-50 Network

Due to the complex structure of ResNet, there are numerous model parameters, and the small-sample data set cannot adequately train such a large network structure. Therefore, the transfer learning approach is applied to train the residual network model. This is achieved by loading the parameters of the ResNet-50 model, which has been pre-trained on ImageNet, as the initial parameters. The fully connected layer parameters are then removed, and the fully connected layer network is reconstructed. Additionally, dropout is added to mitigate the overfitting problem, since the low-level features of edges and contours do not appear to be so important in small-sample learning [18]. Therefore, we chose to load the parameters pre-trained on ImageNet and froze the two residual block parameters at the bottom of the model to not participate in the training, so as to take full advantage of the strong low-level features extraction ability of the pre-trained model on a large data set. Secondly, the ECSM attention mechanism module is introduced at the end of the model used to fuse spatial and channel information to improve the model's ability to discriminate key features.

The introduction of BN (Batch Normalization) [19] in the model was proposed to solve the problem of Internal Covariate Shift. Batch Normalization is an important part of neural networks along with convolution and pooling. The function of the BN layer is to normalize the input of each layer, and to control the input values of the entire neural network within a stable range that is easy to calculate. The BN layer is implemented as shown in Equations (7)–(10):

$$\mu_B = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{7}$$

$$\sigma_B^2 = \frac{1}{n}\sum_{i=1}^{n} (x_i - \mu_B)^2 \tag{8}$$

$$x_i' = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}} \tag{9}$$

$$y_i = \gamma_i x_i' + \beta_i \tag{10}$$

where x_i represents one sample from the input of a batch and n is the batch size. u_B represents the elemental mean of each batch input, σ_B^2 represents the variance of each mini-batch, x_i' represents the normalization of each element, and y_i represents the final output of the network after the normalization process. The structure of the ECSM-ResNet-50 model is shown in Figure 8. The model parameter settings are shown in Table 3.

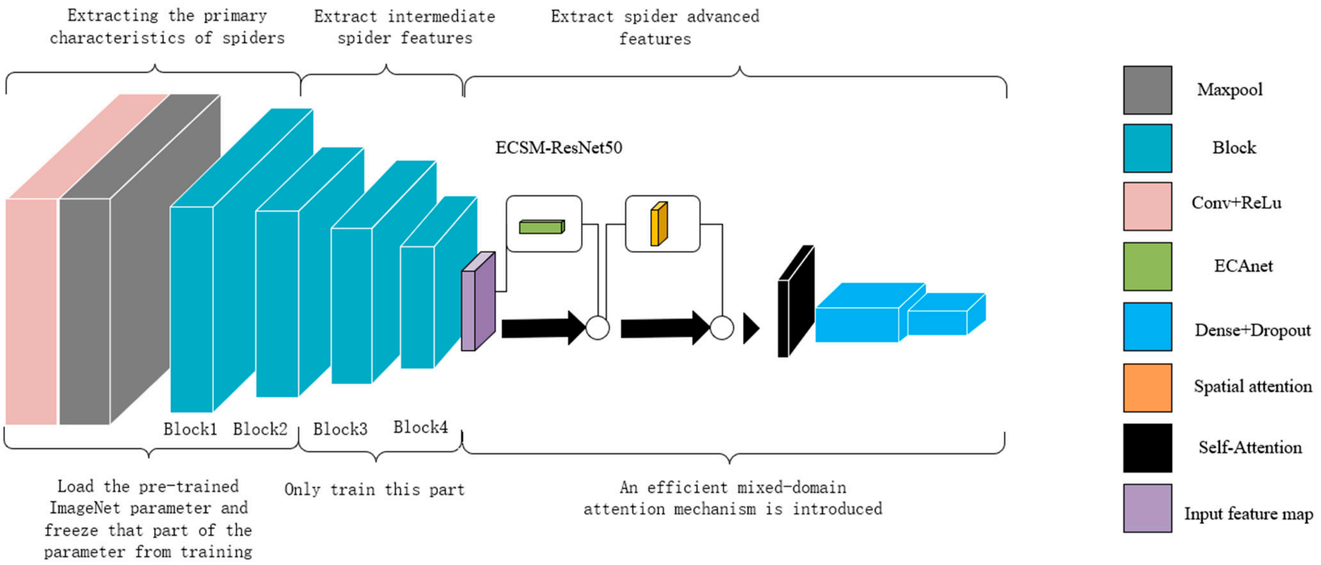

Extracting the primary characteristics of spiders
Extract intermediate spider features
Extract spider advanced features
ECSM-ResNet50
Block1
Block2
Block3
Block4
Load the pre-trained ImageNet parameter and freeze that part of the parameter from training
Only train this part
An efficient mixed-domain attention mechanism is introduced
Maxpool
Block
Conv+ReLu
ECAnet
Dense+Dropout
Spatial attention
Self-Attention
Input feature map

Figure 8. ECSM-ResNet-50 structure.

Table 3. ECSM-ResNet-50 Parameter Set.

Model Parameter Setting Layers	Parameters	Active	Padding	Output Shape	Trainable
input_1 (Layer)	0			(224, 224, 3)	False
conv1 (Conv2d)	9408	ReLu	Same	(112, 112, 64)	False
conv1/batchnorm	256			(112, 112, 64)	False
Relu	0			(112, 112, 64)	False
Block 1	218,624	ReLu	Same	(56, 56, 256)	False
Block 2	1,226,752	ReLu	Same	(28, 28, 512)	False
Block 3	7,118,848	ReLu	Same	(14, 14, 1024)	True
Block 4	14,987,264	ReLu	Same	(7, 7, 2048)	True
ECSM-Attention	4,154,123	ReLu	Same	(7, 7, 2048)	True
dropout	0			(None, 1024)	True
dense	1,049,600	ReLu		(None, 1024)	True
dropout_1	0			(None, 1024)	True
dense_1	5125			(None, 9)	True
softmax (Softmax)	0			(None, 9)	True

3.3.3. ECSM-ResNet-50 Differences from Existing Methods

Compared to ResNet-50, ECSM-ResNet-50 introduces the ECSM module after the last residual block of the model. This module enhances the local information aggregation capability of the model and increases the global information aggregation capability of the model. Among them, the self-attention mechanism introduces a global receptive field where the convolution in the residual block can efficiently learn abstract and low-resolution feature maps in large-scale images, while the added self-attention can process and summarize the information contained in the feature maps. The combination of the two adds global information aggregation capability to the model.

The ECA attention, spatial attention, and self-attention mechanisms extract features from different perspectives, and by combining them, the correlation between channels, the importance of spatial locations, and the dependency of different locations in a sequence can be considered together. This allows for a more comprehensive extraction of features and enhances the expressive power of the model.

Combining multiple attention mechanisms can improve the robustness of the model. By combining them, the potential limitations of a single attention mechanism can be reduced, allowing for better adaptation to complex visual scenes and task requirements.

The model has the disadvantage of a high computational burden. The inclusion of attention mechanisms leads to an increase in computational complexity and the number of parameters, necessitating larger computational resources and training data in order to achieve optimal performance. Training the model from scratch increases the training time.

In this study, we utilize a pre-trained model and employ a layer-by-layer fine-tuning strategy to leverage the powerful feature extraction capability of the pre-trained model and address the issue of model accuracy caused by insufficient training data. Dropout is used to reduce the number of training parameters in order to alleviate the problem of overfitting. By combining the three attention mechanisms, ECSM-ResNet-50 can better utilize the powerful feature extraction capability of the pre-trained model, resulting in a more comprehensive feature capability.

Therefore, utilizing the ECSM-ResNet-50 model with a transfer learning strategy to train the spider data set SPIDER9-IMAGE is advantageous.

3.3.4. Hyperparameter Settings

The experimental hardware configuration is RTX2080Ti GPU × 1, memory 32G × 2, processor Intel E5-2603 × 2. The model is implemented using python3.7 and pytorch1.9.1 deep learning framework under the Windows 10 operating system. The compiled software is PyCharm.

According to the size of the data set and the configuration of the server and other equipment, the number of training rounds is set to 60 Epochs, the Learning rate is 0.0002, and the Batchsize is 32. Use Adam optimization algorithm as optimizer. The model hyperparameter settings are shown in Table 4.

Table 4. Hyperparameter Settings.

Parameter Name	Parameter Value
Epoch	60
Learning rate	0.0002
Batchsize	32
Optimizer	Adam

4. Results and Discussion

4.1. Comparative Experiment of Different Transfer Learning Fine-Tuning Methods

Many models pre-trained on the ImageNet data set can perform well on new tasks. However, the parameters of different layers of the model have different effects on the accuracy of the model. Therefore, choosing appropriate pre-training parameters and fine-tuning layers is very important to obtain good model accuracy. In this experiment, the ResNet-50 model is chosen for its optimization, and the best fine-tuning method is selected by fine-tuning the transfer learning strategy layer by layer. The original fully connected layer of the model is deleted, the fully connected layer is reestablished, global average pooling is added, and the training parameters are reduced by randomly discarding the fully connected layer neurons using dropout. The powerful feature extraction capability of the pre-trained model is utilized to alleviate the problem of insufficient model accuracy caused by insufficient training data.

A comparison of the first convolutional kernel and batch normalization layer parameters using transfer learning with and without the transfer learning method can be found. The distribution of the pre-training parameters using transfer learning is more concentrated and the size difference between the parameters is larger. The distribution of parameters without using transfer learning is more even, as shown in Figure 9.

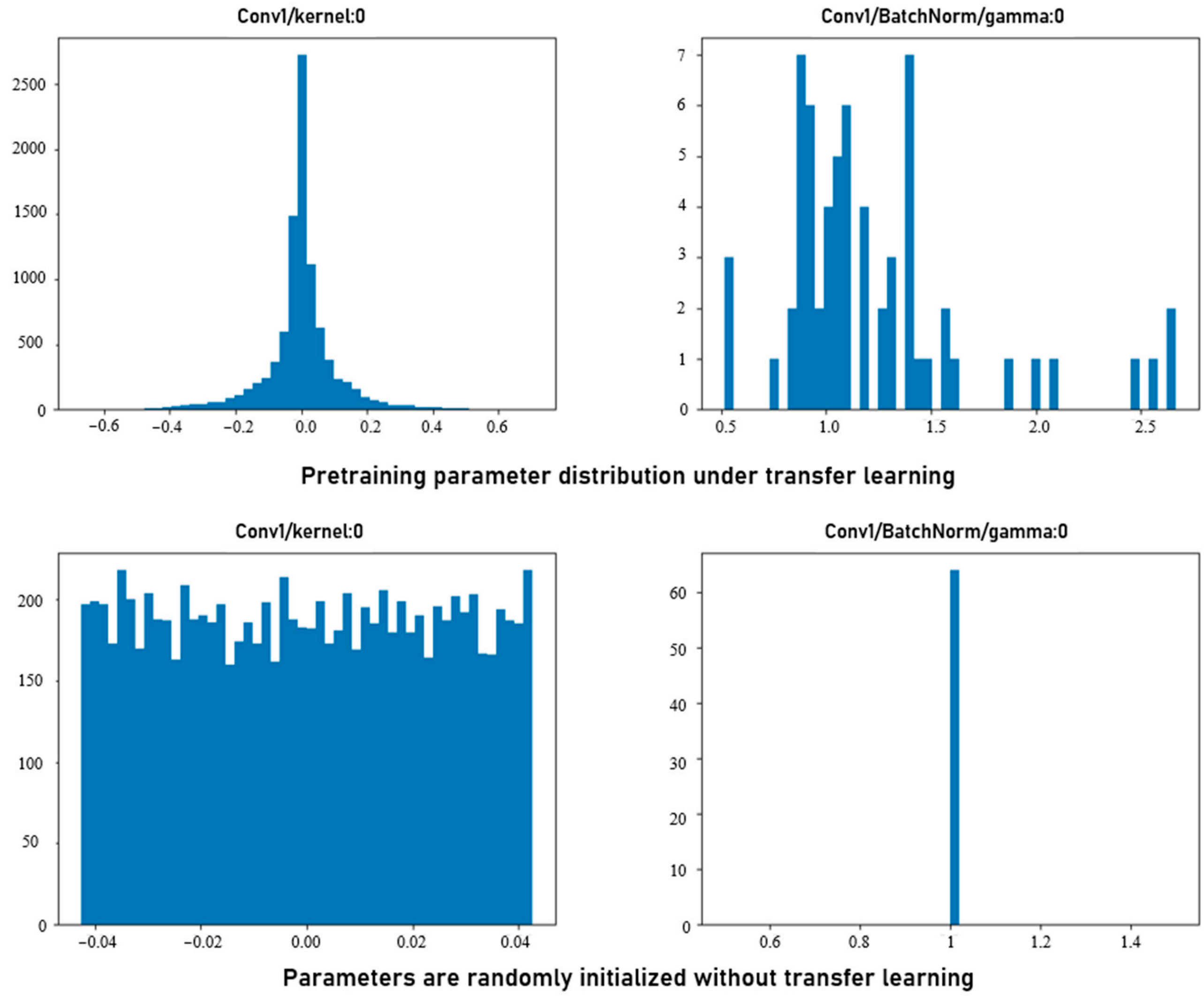


Figure 9. Pre-training parameter distribution.

The number of training rounds is set to 60 Epochs, the learning rate is 0.0002, the Batchsize is 32, and Adam [20] is used as the optimizer in the experiments. When trained with different layer-by-layer fine-tuning transfer learning strategies, ResNet-50 achieves an average test accuracy of 88.68% on the spider data set SPIDER9-IMAGE under the fine-tuning method (3b), which is 1.25% higher than the fine-tuning method (1) and 0.38% higher than the fine-tuning method (2). Among the three fine-tuning methods in method (3), the accuracy of Group b is 0.3% higher than Group a and 1.25% higher than Group c. Training time was significantly lower relative to training from scratch. The experimental results show that it is not better to train more parameters, and selectively and appropriately freezing and thus reducing the training parameters according to the function of each layer of the model can be beneficial to the training of the model. The experimental results of the model under five groups of different transfer learning strategies are shown in Table 5.

The feature matrix for extracting the spider image features according to the model training process is shown in Figure 10. For the spider data set SPIDER9-IMAGE, the regularity of the features extracted by the model during the training process is to extract the edge contour of the spider first, and then extract the detailed information on the spider. As the network level deepens, the more abstract the extracted features are, the fuzzier the corresponding feature matrix becomes. Therefore, freezing the bottom Block 1 and Block 2 parameters of the model can make full use of its strong low-level features extraction ability pre-trained on large networks. The spider data set SPIDER9-IMAGE is then utilized to train the middle and top layers of the model, Block 3 and Block 4, so that they can better extract features unique to spiders. This method learns more information about the features of the spider compared to the traditional freezing of all parameters except the fully connected layer. The pre-training plus fine-tuning method also improves the training speed of the model to a greater extent. Training with the transfer learning approach, the model stands at a higher starting point and can reach an initial accuracy of more than 70% on the test set. It can be seen that the source domain ImageNet is well adapted to the spider data set in the target domain, and fine-tuning the parameters of the pre-trained model provides a great help in the spider species recognition task.

Table 5. Layer-by-layer fine-tuning of the transfer strategy experimental results.

Model	Method	Test Acc/%	Training Parameters	Freeze Parameters	Training Time/s
ResNet-50	1	87.43	2,149,426	23,561,152	2550
	2	88.30	25,710,578	0	3260
	3a	88.30	25,432,114	278,464	2850
	3b	88.68	24,212,530	1,498,048	2690
	3c	87.74	17,114,162	8,596,416	2610

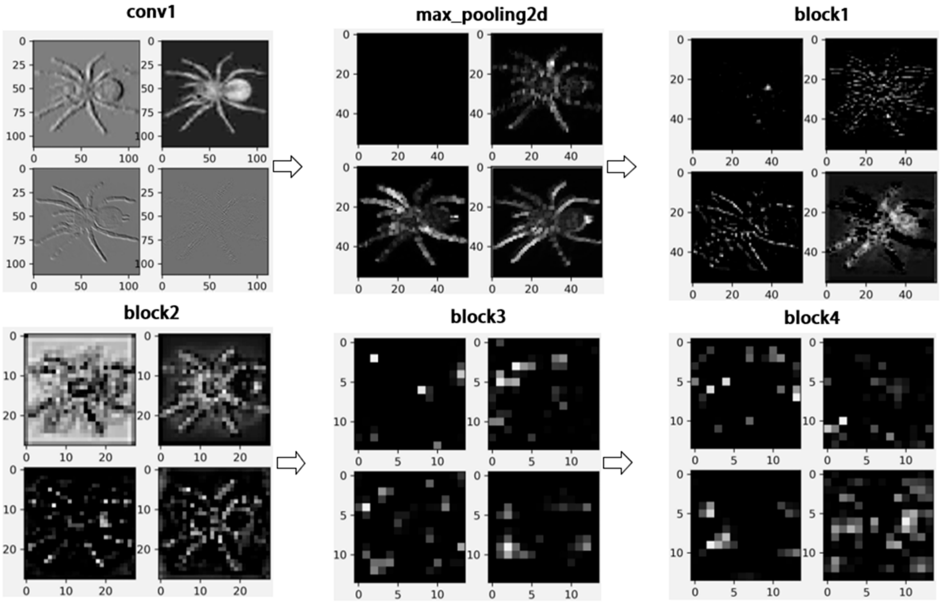


Figure 10. Feature matrix during model training.

4.2. Comparative Experiments of Different Attention Mechanisms

In order to compare the performance of the model's new ECSM attention module with other mainstream attention mechanisms on the spider data set SPIDER9-IMAGE, three attention mechanism models were selected for comparison tests.

The ResNet-50 model was used as the first set of experiments, and the other three attention mechanism modules were selected to be introduced into the ResNet-50 model for comparison tests. In this paper, the position of joining is different for different attention mechanisms. For the SE module in the second group, it is inserted after the four Blocks of the model. The CBAM module of the third group is added to the bottleneck of the model, which is the position after the third convolution operation and before the downsampling. The fourth group of self-attention mechanisms and the fifth group of ECSM attention mechanisms are loaded into the tail of the model.

Epoch was set to 60, learning rate to 0.0002, Batchsize to 32, and Adam was used as the optimizer in the comparison experiments of different attention mechanisms. Five experiments were conducted for each set of attention models, and each experiment constructed the data set by randomly dividing the samples according to the ratio of 70% for the training set and 30% for the testing set. Finally, the average of the five experiments was calculated as the final result.

The results in Tables 6 and 7 demonstrate that the introduction of the attention mechanism, without altering the overall structure of the model structure, will inevitably lead to an increase in the number of model parameters, but it will also enhance the performance of the model to a certain extent. When SE was introduced, the average accuracy of the model on the test set of five experiments reached 89.42%, which was 0.74% higher than the original ResNet-50 network. Introducing the attention mechanism model, the average accuracy of five test sets reached 89.36% compared to the original ResNet-50 network increase by 0.68%. The ECSM-ResNet-50 model, which incorporates an improved hybrid domain attention mechanism, was introduced at the end of the ResNet-50 model, specifically after the four Block layers, and the best fine-tuning method (3b) was selected based on the above layer-by-layer transfer learning experimental results for training. The results showed that the model achieved an average accuracy of 90.25% on the test set in five experiments under the fine-tuning method (3b), which outperformed other models. In the case of adding only a small number of model parameters, the average accuracy of the ECSM-ResNet-50 model improved by 1.57% compared to the original ResNet-50 model, which is higher than the 0.89% improvement achieved by using self-attention mechanisms alone. However, for the residual network with the introduction of CBAM, the accuracy of the model did not improve, but decreased by 2.42%. This shows that the type and size of the data set, the number of model parameters, and the placement of attention mechanisms can all impact the accuracy of the model. Under the same attention mechanism, if the data set is too small, the model may not be sufficiently trained and may be unable to extract enough feature information, which may even lower recognition accuracy.

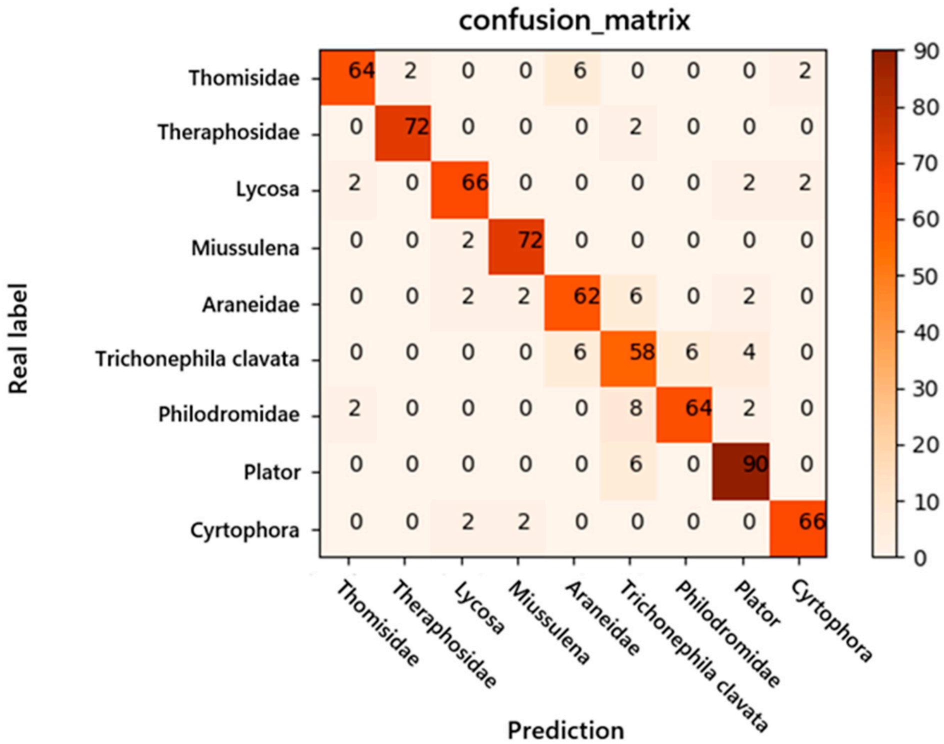
Table 6. Experimental results of different attention mechanisms.

Model	Method	Param/M	Avg Test Accuracy/%	Avg Training Time/s
ResNet-50	3b	25.5	88.68	2690
+SE	2	26.2	89.42	3580
+CBAM	2	28.5	86.26	3660
+Self-attention	3b	28.7	89.36	2840
+ECSM	3b	28.7	90.25	2890

Table 7. Results of five experiments.

ResNet-50	1	2	3	4	5	Avg Test Accuracy%
+SE	89.78	89.78	88.89	89.47	89.18	89.42
+CBAM	85.38	85.09	87.13	87.72	85.97	86.26
+Self-attention	90.35	88.89	89.47	88.89	89.18	89.36
+ECSM	90.95	89.78	90.06	90.64	89.47	90.25

The ECSM-ResNet-50 model, which achieved 90.25% accuracy on the test set, was chosen for testing. According to the confusion matrix in Figure 11 and the results in Table 8, the model recognized Theraphosidae, and it was more prone to errors in recognizing *Trichonephila clavata* and Araneidae; for all other spiders, a higher than 90% recognition rate is guaranteed. The model has high stability.



	Thomisidae	Theraphosidae	Lycosa	Miussulena	Araneidae	Trichonephila clavata	Philodromidae	Plator	Cyrtophora
Thomisidae	64	2	0	0	6	0	0	0	2
Theraphosidae	0	72	0	0	0	2	0	0	0
Lycosa	2	0	66	0	0	0	0	2	2
Miussulena	0	0	2	72	0	0	0	0	0
Araneidae	0	0	2	2	62	6	0	2	0
Trichonephila clavata	0	0	0	0	6	58	6	4	0
Philodromidae	2	0	0	0	0	8	64	2	0
Plator	0	0	0	0	0	6	0	90	0
Cyrtophora	0	0	2	2	0	0	0	0	66

Figure 11. Confusion matrix.

Table 8. Accuracy of various spiders.

Spider	Precision/%	Recall/%
Thomisidae	86.49	94.12
Theraphosidae	97.31	97.31
Lycosa	91.67	91.67
Miussulena	97.31	94.74
Araneidae	83.78	83.78
Trichonephila clavata	78.38	72.50
Philodromidae	84.21	91.43
Plator	93.75	90.10
Cyrtophora	94.28	94.28

4.3. Comparing Experiments with Different Models

To compare the performance of the ECSM-ResNet-50 model with other models on the spider data set SPIDER9-IMAGE, we selected models that were pre-trained on ImageNet, such as Vision-Transformer [21] and GoogleNet [22], for comparison, as shown in Table 9. The ResNet-50 model achieved an average accuracy of 88.68% on the test set, the GoogleNet model achieved 85.34%, the Vision-Transformer model achieved 83.31%, the VGG-16 [23] model achieved 79.06%, and the EfficientNet [24] model achieved 87.74%. The ECSM-ResNet-50 model had the highest test accuracy of 90.25% among the four types of models. Related studies have shown that the Vision-Transformer outperforms CNN on numerous large data sets. However, when it comes to the spider data set SPIDER9-IMAGE, the ViT model cannot be fully trained due to its small-sample size. As a result, its performance does not surpass that of ResNet-50.

Table 9. Comparison of experimental results with different models.

No	Model	Test Accuracy/%
1	VGG-16	79.06
2	GoogLeNet	85.34
3	ResNet-50	88.68
4	Vision-Transformer	83.31
5	EfficientNet	87.74
6	ECSM-ResNet-50	90.25

5. Conclusions

For small-sample data sets, classification tasks have little discriminative feature information, and it is easy to overfit, resulting in classification difficulties and other problems. In this paper, we optimize on the basis of the ResNet-50 model. On one hand, the transfer learning strategy involves utilizing the method of pre-training and fine-tuning to design the frozen layers in a layer-by-layer manner. Selecting only Block 3 and Block 4 for training while keeping Block 1 and Block 2 of the frozen model is the optimal approach. While effectively utilizing the powerful common feature extraction capability of the pre-trained network model, the model can fully extract the high-level features of the target domain. Traditional data augmentation methods are used to increase the amount of data and enhance the generalization performance of the model. On the other hand, the introduction of the attention mechanism enables the construction of the ECSM-ResNet-50 model, which is suitable for small-sample data sets and further enhances the model's performance. The experimental results lead to the following conclusions.

(1) In this paper, we employ a layer-by-layer fine-tuning transfer learning strategy. Through comparative experiments, we determined that freezing the underlying residual blocks Block 1 and Block 2 of the ResNet-50 model and training only Block 3, Block 4, and the fully-connected layer is more effective for the spider data set SPIDER9-IMAGE.

(2) The ECSM-ResNet-50 network model achieves an average accuracy of 90.25% on the test set of the spider data set SPIDER9-IMAGE, which is 1.57% higher than that of the ResNet-50 network. Compared to other mainstream attention mechanisms, it also demonstrates a slight improvement in average recognition accuracy.

(3) ECSM-ResNet-50 has the advantage of the ECSM module at the end of the model. It increases the global receptive field and enhances the model's ability to aggregate global information. The model's feature extraction capability has been improved. The combination of multiple attention mechanisms can improve the model's robustness. It can reduce the potential limitations of a single attention mechanism and better adapt to complex visual scenes and task requirements.

The method has the disadvantage of a large calculation burden. The use of the attention mechanism increases the computational complexity and the number of parameters.

(4) The ResNet model has the advantage of parameter sharing and focuses on aggregating local information, but it lacks the ability to aggregate global information. This paper aims to explore an effective approach to combining global and local information. Specifically, in this paper, the ECSM module is added to the ResNet model, in which the self-attention mechanism can increase the global sensory field of the model, which can make the model focus on aggregating global information. Experiments show that the method used in this paper is an effective exploration.

Author Contributions: Writing—original draft preparation, C.S.; writing—review and editing, J.W.; data processing and visualization, Q.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China (grant number 32001313), Yunnan Fundamental Research Projects (grant number 202201AT070006), Yunnan Postdoctoral Research Fund Projects (grant number ynbh20057) and Fundamental Research Joint Special Youth Project of Local Undergraduate Universities in Yunnan Province (grant number 2018FH001-106).

Data Availability Statement: Not available.

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

References

1. Ji, S.L.; Du, T.Y.; Deng, S.G. A Review of Deep Learning Model Robustness Research. *Chin. J. Comput.* **2022**, *45*, 190–206.
2. Zhao, L.K.; Jin, X.L.; Wang, Y.Z. A Review of Small Sample Learning Research. *J. Soft.* **2021**, *32*, 349–369.
3. Yu, Y.; Wang, L.W.; Zhang, Y.L. Data Enhancement Algorithm Based on Correlation of Feature Extraction Preferences with Background Color. *J. Comput. Appl.* **2019**, *39*, 3172–3177.
4. Wang, J.; Chen, J.; Lin, J. Discriminative Feature Alignment: Improving Transferability of Unsupervised Domain Adaptation by Gaussian guided Latent Alignment. *Pattern Recognit.* **2021**, *116*, 107943. [CrossRef]
5. Kalvakolanu, A.T.S. Plant Disease Detection from Images. *arXiv* **2020**, arXiv:2003.05379.
6. You, K.; Kou, Z.; Long, M. Co-Tuning for Transfer Learning. In Proceedings of the Neural Information Processing Systems (NeurIPS), Virtual, 6–12 December 2020.
7. Zhong, Z.; Lin, Z.Q.; Bidart, R. Squeeze and attention networks for semantic segmentation. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 13062–13071.
8. Woo, S.; Park, J.; Lee, J.Y. CBAM: Convolutional Block Attention Module. In *Lecture Notes in Computer Science*; Springer: Cham, Switzerland, 2018; Volume 11211, pp. 3–19.
9. Vaswani, A.; Shazeer, N.; Parmar, N. Attention Is All You Need. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017; pp. 1–15.
10. Long, M.S. Research on Transfer Learning Issues and Methods. Doctoral Dissertation, Tsing Hua or Qinghua University, Beijing, China, 2014.
11. Huang, Z.; Pan, Z.; Lei, B. Transfer Learning with Deep Convolutional Neural Network for SAR Target Classification with Limited Labeled Data. *Remote Sens.* **2017**, *9*, 907. [CrossRef]
12. Ardalan, Z.; Subbian, V. Transfer Learning Approaches for Neuroimaging Analysis: A Scoping Review. *Front. Artif. Intell.* **2022**, *5*, 780405. [CrossRef] [PubMed]
13. Li, J.; Liu, Y.; Li, Q. Generative adversarial network and transfer-learning-based fault detection for rotating machinery with imbalanced data condition. *Meas. Sci. Technol.* **2021**, *33*, 045103. [CrossRef]
14. Manning, C.D. Effective Approaches to Attention-based Neural Machine Translation. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015.
15. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. *IEEE Trans. Pattern. Anal. Mach. Intell.* **2020**, *42*, 2011–2023. [CrossRef] [PubMed]
16. Wang, Q.; Wu, B.; Zhu, P. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 11531–11539.
17. He, K.; Zhang, X.; Ren, S. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
18. Guan, Y. Flower Recognition System Based on Residual Network Transfer Learning. *Comput. Eng. Appl.* **2019**, *55*, 770–778.
19. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015.
20. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *Comput. Sci.* **2014**. [CrossRef]
21. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A. An Image is Worth 16 × 16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929.
22. Szegedy, C.; Liu, W.; Jia, Y. Going Deeper with Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1–9.
23. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *Comput. Sci.* **2014**. [CrossRef]
24. Tan, M.; Le, Q.V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *arXiv* **2019**, arXiv:1905.11946.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.