Research on Fine-Grained Classification of Small Sample Marine Organism Images

Huibin Luo(✉) and Zixin Lin

Beijing Institute of Technology, Zhuhai 519000, China
zhbitluo@163.com

Abstract. Marine organisms exhibit high species diversity and minimal intra-species differences, often requiring the observation of local variations to accurately identify their categories. However, due to the complexity of underwater imaging conditions, marine organism image datasets are scarce, making accurate classification challenging. To address the problem of limited marine organism image datasets and the difficulty of fine-grained classification, we propose an image classification method based on transfer learning and multi-model ensemble. Firstly, a series of image augmentations are applied to the small sample marine organism images. Next, transfer learning is employed by leveraging the ImageNet-1000 pre-trained weights to aid rapid model learning. ResneSt and Efficientnet deep learning networks are selected for model training. Subsequently, a multi-model weighted ensemble method is used for marine organism image classification. Finally, model predictions are further optimized through binary classification. Experimental results demonstrate that the accuracies of the four models, ResneSt269, Efficientnet-b5, Efficientnet-b6, and "Efficientnet-b5 + Efficientnet-b6," reach 97.5%, 97.68%, 97.86%, and 98.04%, respectively. After model optimization, the ensemble model "Efficientnet-b5 + Efficientnet-b6" achieves an accuracy of 99.11%. Therefore, the proposed method based on transfer learning and multi-model ensemble shows promise for fine-grained classification of small sample marine organism images.

Keywords: Marine organisms · Fine-grained classification · Transfer learning · Multi-model weighted ensemble · Image augmentation

1 Introduction

With the advancement of modern technology, human exploration of the oceans has deepened. Effective methods and monitoring techniques are crucial for determining the quantity and distribution of marine resources, enabling better utilization and conservation. For instance, research on marine organism image classification, through real-time monitoring of marine biological information using underwater video technology, can facilitate the development of marine biological conservation strategies for ecosystem balance. However, capturing marine organism images requires coordination between

V. C. M. Leung et al. (Eds.): Qshine 2023, LNICST 574, pp. 229–243, 2024.
https://doi.org/10.1007/978-3-031-65123-6_17

marine researchers and divers, making image classification more challenging than conventional tasks. Marine organism image classification falls under fine-grained image classification, as it requires experts to analyze subtle features such as color, shape, and texture of marine organisms for accurate classification, demanding significant human resources and time.

In recent years, the rapid development of deep learning technology, especially convolutional neural networks, has fueled progress in image classification. Researchers have utilized large image datasets [1, 2] to extract semantic features using deep learning models like EfficientNet [3] and ResNeSt [4], achieving high accuracy in image classification tasks[5]. In the context of marine organism image classification, some studies have explored deep learning models, such as Qin et al. [6], tamou et al. [7], and Sun et al. [8], based on extensive image datasets (e.g., Fish4Knowledge dataset comprising 23 fish species and 27,370 images), demonstrating high classification accuracy. However, due to limitations in underwater imaging conditions, marine organism image datasets are often scarce, leading to limited research on fine-grained classification of small sample marine organism images. Hence, this research aims to investigate marine organism image classification under limited datasets, starting with data analysis to balance different classes and then augmenting the dataset through various strategies, followed by transfer learning for fine-tuning networks and loading pre-trained weights. The study utilizes different deep learning models like ResNeSt and EfficientNet for training and testing. To enhance fine-grained classification accuracy, a multi-model ensemble approach is employed. Finally, model optimization is performed through binary classification.

2 Preprocessing of Marine Organism Image Data

2.1 Analysis of Marine Organism Image Dataset

2.1.1 Category and Quantity Analysis

Table 1. Species composition and number of image training set

Category	Species name	Number of training sets
1	Eretmochelys imbricata	84
2	Phycodurus eques	84
3	Carcharias taurus	84
4	Triaenodon obesus	80
5	Chelonia mydas	79
6	Zanclus cornutus	76
7	Paraluteres prionurus	76

(*continued*)

Table 1. (*continued*)

Category	Species name	Number of training sets
8	Canthigaster coronata	72
9	Solenostomus paradoxus	70
10	Rhincodon typus	68
11	Caranx sexfasciatus	63
12	Arothron nigropunctatus	62
13	Galeocerdo cuvier	56
14	Plectorhinchus chaetodonoides	56
15	Amphiprion frenatus	56
16	Abudefduf saxatilis	55
17	Stegostoma fasciatum	55
18	Arothron hispidus	49
19	Canthigaster valentini	49
20	Carcharodon carcharias	49

The experimental image dataset used in this study is provided by the University AI Challenge [9], comprising 1881 marine organism images classified into 20 categories, with 1323 images in the training set and 558 in the test set. Table 1 shows the composition and quantity of each species in the training set, with corresponding sample images displayed in Fig. 1.

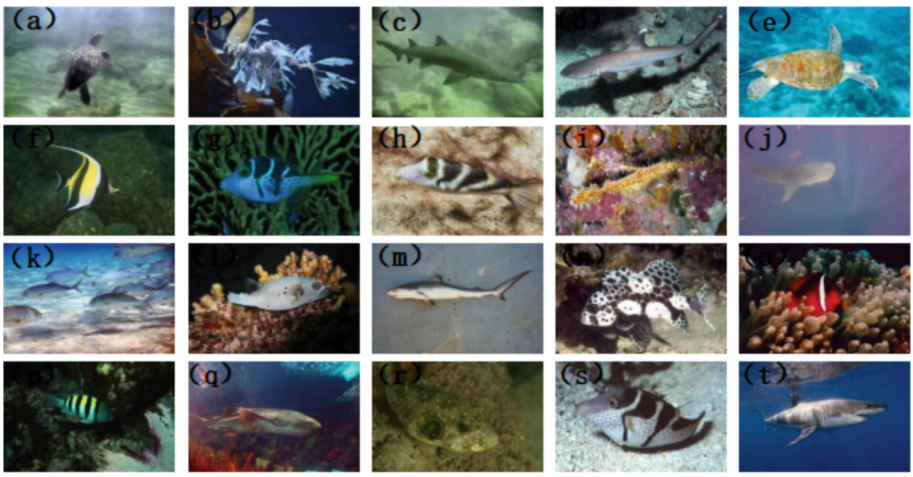

Fig. 1. Sample images of species in the dataset

The Table 1 indicates an imbalance in the training set, with some categories having fewer samples than others. To address this issue, data augmentation is performed to increase the training weight of classes with fewer samples. For example, if class 1 has 50 samples and class 2 has 30 samples, 20 samples are randomly selected from class 2 and duplicated until the class 2 data reaches 50 samples.

2.1.2 Image Size Analysis

As shown in Fig. 2, there are relatively large and relatively small parts of the image size, but most of them will be concentrated near the 500 * 500 area. Through clustering analysis of the width and height of marine biological images with a class of centroid numbers, the clustering results were [415.330, 555.628]. Therefore, in the study, image sizes of [456, 456] and [500, 500] can be selected as input sizes for comparison.

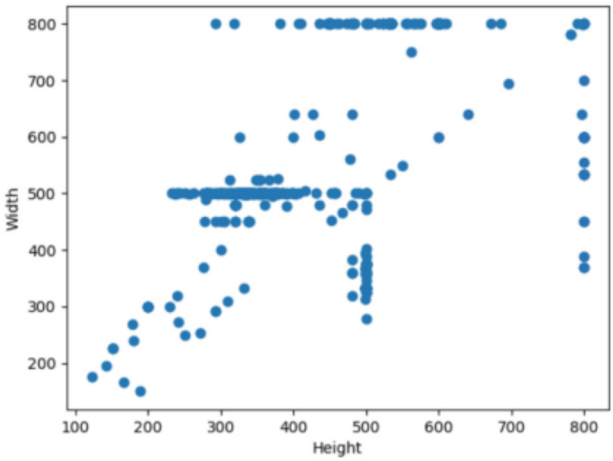

Fig. 2. The height and width of the image in the dataset

2.2 Data Augmentation of Marine Organism Image Data

As marine organism images are challenging to acquire due to underwater conditions, there is a limited number of samples available, and the images may suffer from poor clarity, caused by factors like lighting and visibility. With only 1323 images and 20 classes, there is a risk of overfitting and insufficient generalization during model training. To address these issues, data augmentation is employed to enhance the model's generalization and robustness without consuming excessive resources. However, not all data augmentation techniques are equally effective for fine-grained classification of marine organism images. As shown in Fig. 3, due to the small inter-class differences in fine-grained classification, applying color transformations to some species may reduce their distinctiveness, leading to classification errors.

Fig. 3. (L) Paraluteres prionurus. (R) Canthigaster coronata

Fig. 4. (L) Original image. (R) Enhanced image

Therefore, data augmentation strategies should be tailored to the characteristics of marine organisms (Fig. 4).

Common data augmentation techniques include flipping (horizontally flipping marine organism images around the Y-axis), blurring (applying random-sized kernel blurs to marine organism images), random rotation by 90°, constrained adaptive histogram equalization for contrast enhancement, grid distortion (random compression), and random affine transformations (random translation, scaling, and rotation). Figure 5 demonstrates the effects of these six commonly used data augmentation methods on marine organism images.

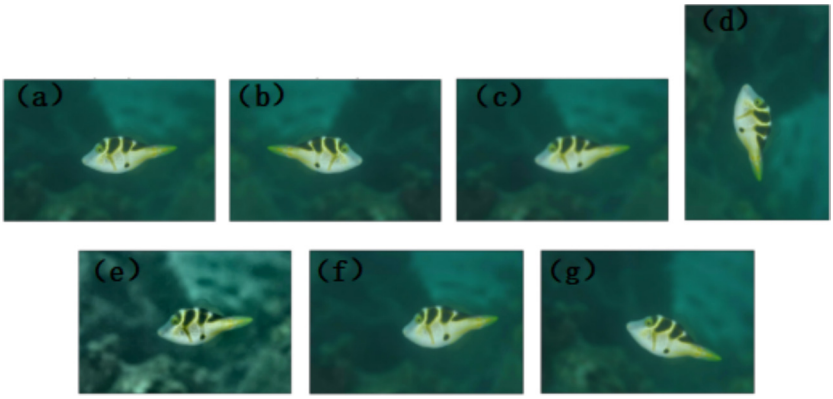

Fig. 5. Image enhancement processing effect in different ways

Additionally, CutMix data augmentation can be employed on top of the six aforementioned methods. CutMix involves randomly generating a cropping box, cropping a

region from one image, and pasting it into another image, resulting in a new data sample. Figure 6 illustrates the CutMix augmentation process, which introduces two class components when calculating the loss for the combined data samples.

$$\begin{aligned} \tilde{x} &= M \odot x_A + (1 - M) \odot x_B \\ \tilde{y} &= \lambda y_A + (1 - \lambda) y_B \end{aligned} \tag{1}$$

Among them, x_A represents Image A, and x_B represents Image B; y_A represents the label of Image A, and y_B represents the label of Image B; M is a mask matrix that is equal to the pixels in Image A or Image B, with cropped regions having values of 0 and the remaining reserved regions having values of 1; λ Represents the weighting coefficient [10].

Fig. 6. (L) image A (M) image B (R) Cutmix enhancement image

3 Deep Learning-Based Image Classification Models

In the field of deep learning, there are many models suitable for image classification. Among them, the ResNeSt and EfficientNet series have gained considerable attention from researchers due to their strong performance in attention mechanisms and neural architecture search.

3.1 ResNeSt Model

In the ResNet series, the ResNeSt-related blocks evolved from the ResNet bottleneck [11], as shown in Fig. 7. The Inception block, proposed by Szegedy et al. [12], is based on ResNet and divides the convolutional kernel (conv1x1 + conv3x3) into different sub-modules according to different receptive fields. In 2017, Xie et al. [13] introduced ResNeXt, which balanced ordinary convolutional kernels and depth-wise separable convolution through group convolution strategy. In 2020, Zhang et al. [4] combined ResNeXt and Inception blocks to create the ResNeSt Block, where multiple Cardinals are stacked after conv1x1 and conv3x3 sub-modules. A conv1x1 layer adjusts the number of channels, followed by an Add operation with the input. The structure is illustrated in Fig. 8. In summary, ResNeSt is based on ResNet, with additional features like split grouping and attention mechanisms, significantly improving the model's accuracy with only a small increase in parameter count.

In terms of attention mechanisms, the Split Attention combines attention and the split block in ResNeXt. The basic attention module includes SE-block (Squeeze-and-Excitation block) or CBAM (Convolutional block attention module), where SE-block

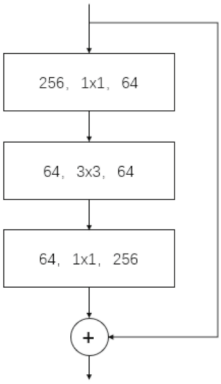

Fig. 7. Resnet bottleneck

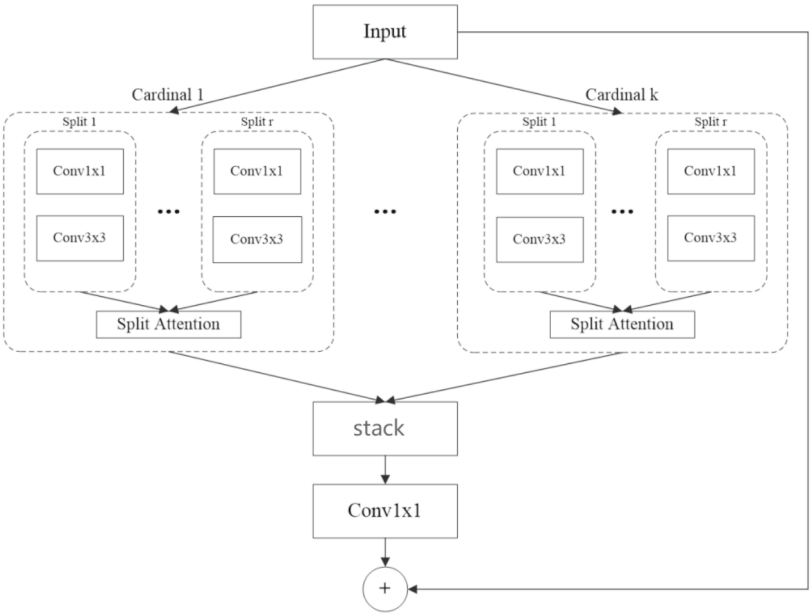

Fig. 8. ResneSt Block

balances the weights of different channel features, and CBAM combines spatial and channel attention mechanisms, achieving further improvements over SE-block. In fine-grained classification of marine organism images, where inter-class differences are subtle, the SE-block [14] and CBAM [15] attention mechanisms help the model focus on important parts of marine organism images (e.g., mouth color, size, or shape) to enhance feature representation. Thus, using ResNeSt with attention mechanisms is suitable for fine-grained classification of marine organism images.

3.2 EfficientNet Series Models

EfficientNet achieves optimal combinations of parameters through neural architecture search to represent the best capabilities of the model. Typically, higher accuracy in models requires increasing their size, such as the depth, width, and resolution of deep learning networks. However, as these parameters increase, the search space becomes vast, consuming significant resources. To improve resource utilization, researchers fixed the EfficientNet's network structure and imposed certain constraints on the three dimensions (depth, width, and resolution) parameters. By multiplying the baseline network (EfficientNet-b0) by a specified factor for these three dimensions, they achieved a better search. Changes in any of these dimension parameters may yield accuracy improvements to some extent, but as the parameter values increase, the improvement diminishes, often reaching an upper limit accuracy. To address the issue of an upper limit in dimension parameter accuracy, Tan et al. [3] proposed a Mixed Dimension Scaling method.

$$\begin{aligned} depth{:}d &= \alpha^{\phi} \\ width{:}w &= \beta^{\phi} \\ resolution{:}r &= \gamma^{\phi} \\ s.t.\ \alpha \cdot \beta^2 \cdot \gamma^2 &\approx 2 \\ \alpha \geq 1, \beta \geq 1, \gamma &\geq 1 \end{aligned} \tag{2}$$

This method balances the three dimension parameters (α, β, γ) through a mixing coefficient Φ (customized based on the situation). The constraint is that d, w^2, and r^2 are positively correlated and their product is approximately equal to 2. In short, EfficientNet optimizes and adjusts α, β, and γ parameters to minimize resource consumption while improving the model's representational capacity to the fullest extent possible.

4 Experimental and Results Analysis

4.1 Experimental Setup

4.2 Development Environment

The experiments in this study were conducted on a hardware setup consisting of an I7-6700 CPU and a GeForce RTX2080 GPU. PyTorch was utilized for image data analysis and preprocessing. The ImageNet-1000 pre-trained weights were loaded into the respective network models. The experiments were performed with different optimization algorithms, learning rate decay strategies, input image resolutions, and batch sizes for both single-model and multi-model ensemble training. For single-model marine image classification, ResNeSt269, Efficientnet-b5, and Efficientnet-b6 were employed. For multi-model weighted ensemble classification, combinations such as "Efficientnet-b5 + ResNeSt269," "Efficientnet-b6 + ResNeSt269," "Efficientnet-b5 + Efficientnet-b6," and "Efficientnet-b5 + Efficientnet-b6 + ResNeSt269" were used.

4.2.1 Transfer Learning

In deep learning, model training is a learning process that requires a significant amount of data for the model to achieve good fitting capabilities and avoid the influence of large differences in specific data on the model's judgment ability. However, in small sample image datasets, even with data augmentation, the dataset remains relatively small. The marine organism image dataset used in this study contains only slightly over 1300 images, with some categories having only 49 images. Training models with small sample data can lead to overfitting or weak generalization capabilities. To address this issue and improve model accuracy with limited data, transfer learning can be applied. Transfer learning involves fine-tuning the network to enable the model to rapidly acquire high-performance processing capabilities. In image classification, transfer learning can be achieved by loading pre-trained weights from a larger dataset like ImageNet-1K to assist the model in fast learning. In this study, the pre-trained weights for the image classification models ResNeSt269, Efficientnet-b5, and Efficientnet-b6 correspond to "Resnest269-0cc87c48.pth," "efficientnet-b5-b6417697.pth," and "efficientnet-b6-c76e70fd.pth," respectively. The process of incorporating transfer learning [16] into the marine organism image classification is illustrated in Fig. 9. Firstly, the parameters of the feature extraction layer in the network need to be transferred. Next, the classification layer of the network is modified to aid in training. Then, the parameters of the feature extraction layer are frozen, and the network training is activated to allow the transferred feature extraction parameters to aid in training the classification layer. Finally, the frozen feature extraction layer parameters are unfrozen, and training is performed for all parameters.

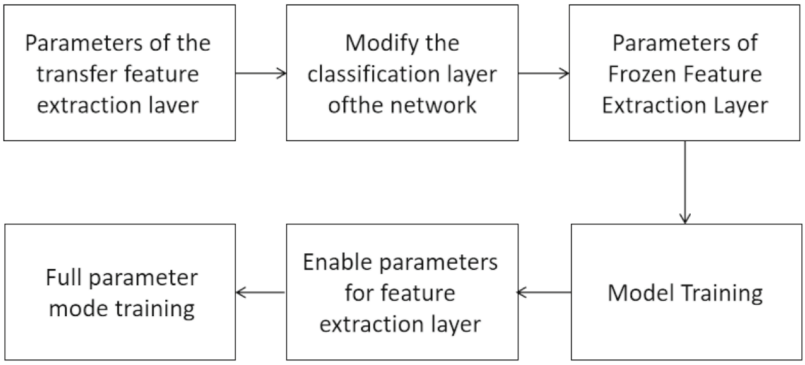

Fig. 9. Block diagram of model training process based on transfer learning

4.3 Single-Model Marine Organism Image Classification

4.3.1 Evaluation of ResNeSt269 Network Model

In this study, additional non-linear connecting layers such as Dropout and Relu layers were added to the classification layer of the ResNeSt269 network. The experiments

compared different image resolutions, optimizers, learning rate decay strategies, batch sizes, and data augmentation methods. The results of the experiments are shown in Table 2, with the highest accuracy achieved by the ResNeSt269 model reaching 97.50%. Based on the results in Table 2, further analysis reveals the following: (1) Compared to fixed step learning rate decay strategy, using the cosine annealing learning rate decay strategy allows for more comprehensive model training, enabling the model to converge better. (2) Regarding image resolution, the accuracy is higher when using input image size of 500*500 compared to input image size of 456*456 (3) For batch size, a larger batch size allows the model to fit the data distribution of a batch more effectively, resulting in better generalization capabilities. If the batch size is too small, it may lead to unusual data distribution in some images, affecting model fitting. (4) Regarding data augmentation, adding the Cut-Mix data augmentation can also improve the accuracy of the model classification to some extent.

Table 2. ResneSt model experimental results

Model	Resolution	Optimizer/Learning rate decay strategy	Batch Size	Acc (%)	Data augmentation methods
ResneSt269	456*456	SGD (0.001)/Fixed step learning rate attenuation	4	95.18	Six types of data augmentation
	500*500	SGD(0.001)/Cosine annealing learning rate decay	4	96.43	Six types of data augmentation
		SGD(0.001)/Cosine annealing learning rate decay	8	96.96	Six types of data augmentation
		SGD(0.001)/Cosine annealing learning rate decay	4	97.14	Six types of data augmentation + CutMix
		SGD(0.001)/Cosine annealing learning rate decay	8	**97.50**	Six types of data augmentation + CutMix

4.4 Evaluation of Efficientnet Network Model

Based on the analysis results from the previous section, for the Efficientnet model, the authors chose an image resolution of 500*500 and a batch size of 8. They evaluated the image classification performance with different optimizers and augmentation methods. The experimental results are shown in Table 3, indicating that the highest accuracy achieved by the model is also 97.50%, and different data augmentation methods can

influence the model's accuracy. Unlike the ResneSt model, where using 6 data augmentations helps improve generalization, the Efficientnet model achieves better accuracy with 5 data augmentations. Therefore, different models may require trying different data augmentation strategies, and for Efficientnet, incorporating CutMix data augmentation during data preprocessing also contributes to improved accuracy.

Table 3. Efficientnet model test results analysis table

Model	Optimizer/Learning rate decay strategy	Acc (%)	Data augmentation methods
Efficientnet-b5	Adam(0.0001)/Cosine annealing learning rate decay	96.96	Six types of data augmentation + CutMix
	Adam(0.0001)/Cosine annealing learning rate decay	**97.32**	Five types of data augmentation + CutMix
Efficientnet-b6	Adam(0.0001)/Cosine annealing learning rate decay	96.96	Six types of data augmentation + CutMix
	Adam(0.0001)/Cosine annealing learning rate decay	**97.50**	Five types of data augmentation + CutMix

4.5 Multi-model Weighted Ensemble for Marine Organism Image Classification

For multi-model weighted ensemble, the marine organism images are fed into each individual model, and each model outputs a result. Then, the results are combined using the softmax function to determine the allocation weights for each result. Each result is then multiplied by its corresponding weight and summed up to obtain the ensemble result [17]. The multi-model ensemble process is illustrated in Fig. 10, where each model's output is distributed based on the weight ratio. In the case of a two-model ensemble, for example, if model A assigns a confidence score of 0.8 to class 1, and model B assigns a confidence score of 0.5 to class 1, then the weight of model A's result in the final ensemble output would be $0.8/(0.8 + 0.5) = 0.615$, and the weight of model B's result would be $0.5/(0.8 + 0.5) = 0.385$. By combining the results of multiple models, the final test results have better representation capabilities.

The experimental results of the multi-model weighted ensemble are shown in Table 4, indicating that in general, multi-model ensemble tends to improve the model's performance compared to single-model classification. However, the number of models and their accuracy may not always be directly proportional. For example, the three-model ensemble method "Efficientnet-b5 + Efficientnet-b6 + ResneSt269" performed slightly worse than the two-model ensemble method "Efficientnet-b5 + Efficientnet-b6".

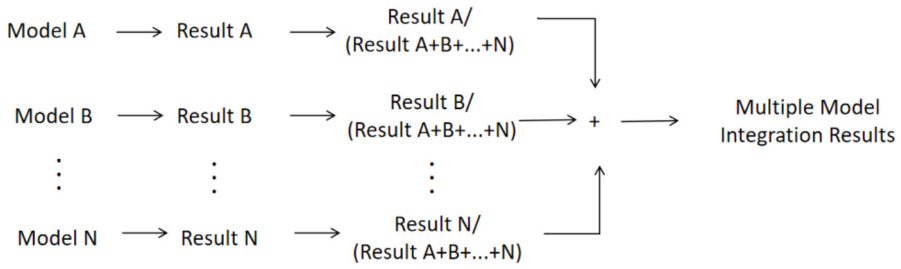

Fig. 10. Multi-model integration block diagram

Table 4. Multi-model integrated results

Model integration method	Acc (%)	Model improvement effect (%)
Efficientnet-b5 + ResneSt269	97.50	0.18
Efficientnet-b6 + ResneSt269	97.68	0.18
Efficientnet-b5 + Efficientnet-b6	**98.04**	**0.54**
Efficientnet-b5 + Efficientnet-b6 + ResneSt269	97.86	0.38

4.6 Model Optimization

In the single-model testing, the authors analyzed the classification accuracy of different models for each category and presented the results in Fig. 11. Observing Fig. 11, it can be seen that in the three image classification models (ResneSt269, Efficientnet-b5, and Efficientnet-b6), the accuracy for category 1 is relatively low, and among them, the Efficientnet-b6 model has the lowest accuracy for category 1. To further analyze the reasons for misclassifications in this model, the authors examined the individual class predictions made by the Efficientnet-b6 model, and the results are shown in Fig. 12. It was found that the Efficientnet-b6 model misclassifies a small portion of hawksbill turtles (category 1) as sea turtles (category 5). These two classes belong to the same broader category of turtles, and their sub-classification differs only slightly, leading to difficulties in accurate differentiation by all three models among the 20 classes. To address this, the researchers optimized the model by improving the classification of similar classes from multi-class to binary classification. The specific steps of the optimization process are as follows: Input the test set for model testing to obtain the model accuracy for the 20-class classification. Remove the classification results where the model predicted category 1 or category 5 and the true label is category 1 or category 5, and calculate the accuracy for the remaining 18-class classification. Set category 1 as the first category and category 5 as the second category in the binary classification. Test the model using binary classification and obtain the binary classification accuracy. Combine the results and calculate the final model's Top-1 accuracy.

Through model optimization, the authors retrained the marine organism images, enabling the model to focus on fitting the data distribution of these two classes and

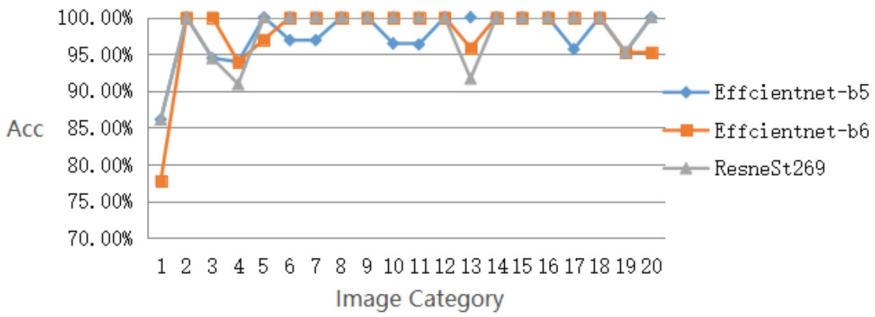

Fig. 11. Image Category accuracy under different models

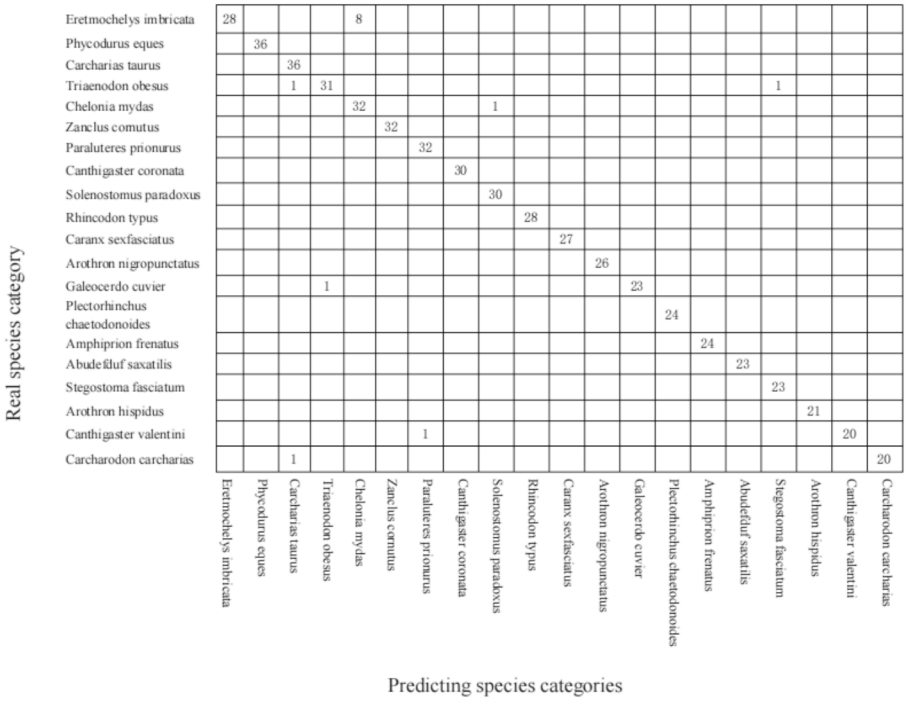

Fig. 12. Confusion matrix of prediction results of Efficientnet-b6 model

distinguishing the differences between them for better classification. The experimental results after optimization are shown in Table 5, where the optimized accuracy is obtained by summing the correctly classified images in binary classification and the correctly classified images in 18-class classification, then dividing by the total number of test samples. From Table 5, it can be observed that the optimized accuracy is higher than the 20-class classification for all models, with improvements of 0.36%, 0.72%, 1.07%, and 1.07% for ResneSt269, Efficientnet-b5, Efficientnet-b6, and the ensemble model "Efficientnet-b5 + Efficientnet-b6," respectively. The experimental results demonstrate

that the binary classification model optimization method can effectively enhance the fine-grained classification of similar species in marine organism images to a certain extent.

Table 5. Model optimization test result analysis

Model	20 Classification Acc (%)	18 Classification Acc (%)	2 Classification Acc (%)	Optimized Acc (%)
ResneSt269	97.50	98.17	95.65	97.86
Efficientnet-b5	97.32	98.16	97.14	98.04
Efficientnet-b6	97.50	98.78	97.06	98.57
Efficienet-b5 + Efficienet-b6	98.04	99.18	98.57	**99.11**

5 Conclusion

In this study, the authors addressed the challenge of fine-grained classification in small-sample marine organism image datasets. They combined transfer learning and multi-model weighted ensemble methods to train deep neural network models (ResneSt269, Efficientnet-b5, and Efficientnet-b6) for marine organism image classification. During the research process, they eliminated class imbalance issues through data analysis, applied a series of image augmentations to enhance model generalization and prevent overfitting in the context of a small-sample dataset and fine-grained classification. Furthermore, they chose deep neural network models to better extract semantic information from marine organism images, enhancing the models' representation capabilities. Additionally, they applied transfer learning to optimize model training parameters and improve learning efficiency. The multi-model weighted ensemble method was utilized to increase model accuracy. Finally, they employed a binary classification model optimization approach to improve the accuracy of fine-grained marine organism image classification. Overall, this study achieved relatively high accuracy for fine-grained marine organism image classification, considering the small-sample and low inter-class differences in marine organism images. The proposed combination of transfer learning and multi-model weighted ensemble methods shows promising application potential and provides technical support for marine organism classification and regional distribution research.

References

1. Krizhevsky, A., Sutskever, I., Hinton, G.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, vol. 25 (2012)
2. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv, arXiv:1409.1556 (2014)

3. Tan, M., Le, Q.V.: EfficientNet: rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning, pp. 6105–6114. PMLR (2019)
4. Zhang, H., Wu, C., Zhang, Z., et al.: ResNeSt: Split-Attention Networks. arXiv preprint arXiv: 2004.08955 (2020)
5. Zhou, F.Y., Jin, L.P., Dong, J.: Review of convolutional neural network. Chin. J. Comput. (2017). https://doi.org/10.11897/SP.J.1016.2017.01229
6. Qin, H.W., Li, X., Liang, J., et al.: DeepFish: accurate underwater live fish recognition with a deep architecture. Neurocomputing **187**, 49–58 (2016)
7. Tamou, A.B., Benzinou, A., Nasreddine, K., Ballihi, L.: Underwater live fish recognition by deep learning. In: Mansouri, A., El Moataz, A., Nouboud, F., Mammass, D. (eds.) ICISP 2018. LNCS, vol. 10884, pp. 275–283. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-94211-7_30
8. Sun, D.Y., Zhang, J.H., Sun, L.Y., et al.: Classification of reef biological images of marine ranch based on deep convolution neural network. Oceanologia Et Limnologia Sinica **52**(05), 1160–1169 (2021)
9. Ai futurelab 2020. Image dataset (2020). https://ai.futurelab.tv/contest_detail/18#data. Accessed 16 May 2022
10. Yun, S., Han, D., Chun, S., et al.: CutMix: regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6023–6032 (2019)
11. He, K.M., Zhang, X.Y., Ren, S.Q., et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, pp. 770–778. IEEE (2016)
12. Szegedy, C., Vanhoucke, V., Ioffe, S., et al.: Rethinking the Inception Architecture for Computer Vision, pp. 2818–2826. IEEE (2016)
13. Xie, S., Girshick, R., Dollár, P., et al.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1492–1500 (2018)
14. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141 (2018)
15. Woo, S., Park, J., Lee, J.Y., et al.: CBAM: convolutional block attention module. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 3–19 (2018)
16. Yosinski, J., Clune, J., Bengio, Y., et al.: How transferable are features in deep neural networks?. In: Advances in Neural Information Processing Systems, vol. 24 (2014)
17. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. Comput. Sci. **14**(7), 38–39 (2015)