Hindawi
Aquaculture Research
Volume 2023, Article ID 3224064, 14 pages
https://doi.org/10.1155/2023/3224064

Research Article

Individual Fish Recognition Method with Coarse and Fine-Grained Feature Linkage Learning for Precision Aquaculture

Jianhao Yin,[1,2] Junfeng Wu ,[1,2,3] Chunqi Gao,[1,2] Hong Yu,[1,2] Liang Liu,[1,2] Zhongai Jiang,[1,2] and Shihao Guo[1,2]

[1]*College of Information Engineering, Dalian Ocean University, Dalian 116023, China*
[2]*Dalian Key Laboratory of Smart Fisheries, Dalian Ocean University, Dalian 116023, China*
[3]*Key Laboratory of Environment Controlled Aquaculture, Ministry of Education, Dalian Ocean University, Dalian 116023, China*

Correspondence should be addressed to Junfeng Wu; wujunfeng@dlou.edu.cn

Received 11 July 2023; Revised 13 September 2023; Accepted 9 October 2023; Published 26 October 2023

Academic Editor: Houguo Xu

With the increasing level of precision and intelligence in the aquaculture, real-time mastery of the growth status of aquaculture individuals has become an important means to improve aquaculture efficiency and save resources and the environment. Therefore, accurate individual recognition of underwater fish has become one of the key technologies for precision aquaculture. In order to cope with the impact of the complex underwater environment on the recognition accuracy, this paper proposes a coarse and fine-grained features learning method for individual fish recognition. The method consists of a coarse-grained feature learning network and two fine-grained feature learning networks. The trunk of the network is responsible for learning coarse-grained features of the fish, the first branch learns fine-grained features of fish from head, body, and tail, and the second branch learns fine-grained features of fish from upper and lower fins. we supplemented different levels of noise and attack to the training set of fine-grained features and enriched the grayscale variation to cope with the complexity and variability of the underwater environment. The simulation experimental results show that the method achieves more than 96.7% in key indicators such as Rank-1 and Rank-5, and also performs well in other fish recognition tasks with certain generalization.

1. Introduction

Aquaculture has moved toward a rapid development path and aquatic product output has increased significantly. The development of aquaculture not only meets the demand for aquatic products and expands the export of aquatic products, but also makes an important contribution to increasing the income of fishermen. In recent years, new aquaculture technologies have been continuously introduced. Factory aquaculture, intelligent new aquaculture models have developed rapidly. Traditional aquaculture mode relying on experience, artificial, and weather has become more and more unsuitable for the needs of modern agricultural production and management. Because of the continuous expansion of aquaculture scale and categories, it is of great significance to effectively obtain and analyze some important information generated in the production process. The information is very important for reducing the risk of aquaculture, improving the economic benefits of enterprises, and reducing the labor intensity of employees. Accurate and real-time mastery of fish health, population density, and behavior could provide important data support for making production management decisions. It also provides analytical data for disease prevention and control, bait feeding, and feed formulation management. However, these data are obtained on the basis of fish individual recognition, which means that on the basis of the identified fish species, it is necessary to further determine the fish individual.

With the continuous development of artificial intelligence technology and computer vision technology, deep convolutional neural networks have been well applied in target detection, face recognition, and medical fields, which makes it possible to apply them to the field of fish individual recognition. However, as our research progressed, we found that the existing established techniques did not correspond to the characteristics of underwater fish activity. As shown in

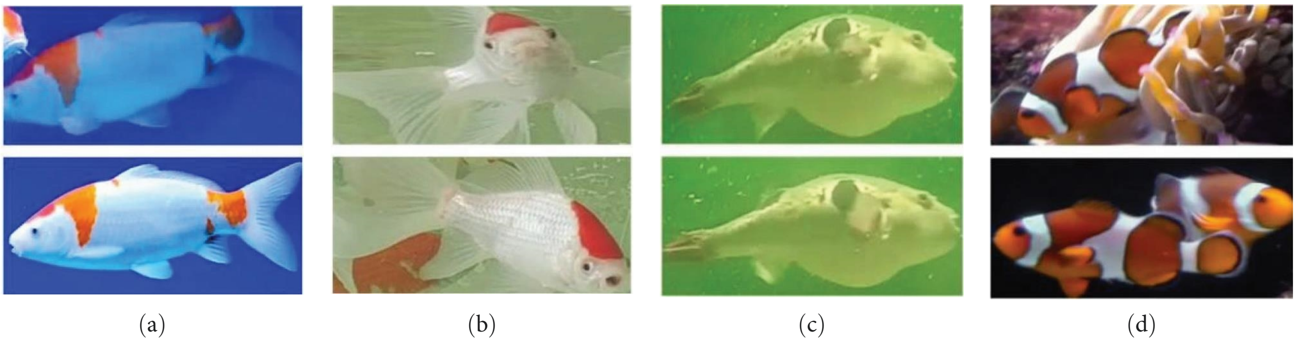


Figure 1: Underwater fish individual sample case (a) color variation, (b) posture variation, (c) similar background environment, and (d) feature occlusion.

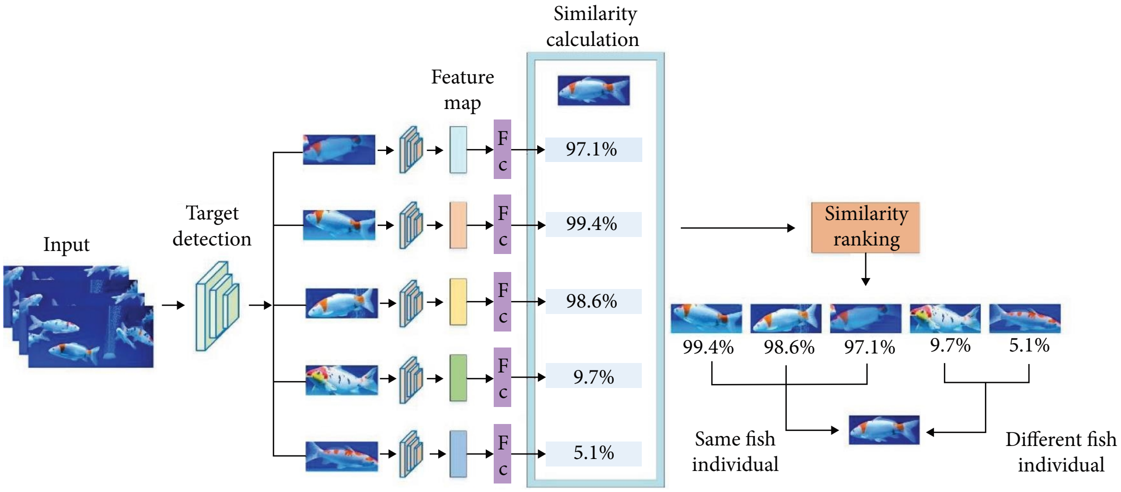


Figure 2: Schematic diagram of fish individual recognition process.

Figure 1. First, the same fish individual will create significant color variations under different lighting conditions, which makes it more challenging to recognize different fish individuals by macroscopic features. Second, the complex and variable underwater environment may lead to severe mutual occlusion of fish individuals, which makes it difficult to extract effective features for the different individuals with similar appearances. Third, it is challenging to get high-quality images of fish individuals due to their diverse variety of swimming trajectories and movements. Finally, it is difficult for researchers to determine the movement of fish using reference objects due to the variations in water quality and the lack of fixed reference objects in the underwater environment.

In order to effectively overcome the difficulties and further improve the accuracy and robustness of fish individual recognition methods. This paper proposes a fish individual recognition method based on coarse and fine-grained feature-linked learning, and the main contributions of this paper are:

(1) In order to solve the problem of difficult extraction of fish visual features due to the mutual occlusion of underwater fish, this paper proposes the extraction of fish visual features by using coarse and fine-grained feature-linked learning methods.

(2) To address the problem of low accuracy of fish individual recognition due to uneven illumination and blurred underwater images, we proposed a method to assign nonuniform weight to the backbone network to improve the extraction effect of fish features. The WeightConv block is formed by adding the SE (Squeeze and Excitation) attention mechanism module to the residual side section of the ResNet 50. The network can compute different weights of different channels, reduce the network's attention to the background environment, and enhance the ability to extract fish features. Thereby further improving the accuracy of the network to recognize fish individuals when the background environment is blurred or similar.

(3) To solve the problem of poor training effect of deep convolutional neural networks due to small underwater fish individual datasets and low-image quality, we proposed a new method applicable to underwater real-time fish individual recognition. As shown in Figure 2. This method does not require the advanced storage of fish characteristic information for retrieval, in contrast to traditional biometric techniques. The network can autonomously extract and store the visual attributes of unknown fish individuals and

assign them a unique identification number when performing the individual recognition task. This method provides real-time feedback on the results of the similarity calculation and its numbering information when fish individual features are extracted and queried again.

The structure of this paper is as follows: the second part of the paper mainly introduces the related work of individual recognition, and the third part mainly introduces our proposed coarse and fine-grained feature linkage learning method. The fourth part mainly introduces the results of the simulation experiments, and the fifth part summarizes the proposed work and prospects for the future work.

2. Related Work

With continuous research in the field of deep learning, biometric technology has been developed significantly, and face recognition technology has been very widely used. As research progresses, people gradually realize that biometric technology also has important significance in the fields of ecological environmental protection, precise breeding of animal husbandry, and protection of endangered species.

In order to accomplish individual identification of wild elephants, Körschens et al. [1] used curvature integration and two different matching algorithms to accomplish identification and data acquisition of wild elephants. Li et al. [2] in 2019 made public a dataset of 92 northeastern tigers with over 8,000 video clips of northeastern tigers and filled the gap in northeastern tiger dataset by including key points, identity information, and other annotations. Boom et al. [3] used 10 underwater cameras for 3 years to extract fish data from live video and obtained a dataset of 27,370 different fish of 23 species. Recognizing the need for datasets that track fish individuals over time, Yin et al. [4] created the DlouFish dataset with labeled images of 384 fish, each with different individual numbers, totaling 6,950 images.

In the field of underwater species recognition, the noise of underwater images such as water quality and light intensity are important factors that affect the accuracy of the model. Therefore, Jian et al. [5] proposed an underwater image correction method based on photoactive imaging in image preprocessing to correct and reconstruct distorted images. Pramunendar et al. [6] used Contrast Adaptive Color Correction technique (NCACC) for image enhancement and improved the species identification accuracy to 93.73%. Chuang et al. [7] proposed a framework for underwater fish recognition consisting of a fully unsupervised feature learning technique and an error resilient classifier that uses information from fuzzy images to assign coarse labels by optimizing the benefit of decisions made by the classifier, introducing the concept of partial classification.

When performing underwater fish individual recognition tasks, we usually need two stages. First, target detection is performed to separate the fish from the background environment, and then individual features are learned to complete the final recognition. Convolutional neural network (CNN) has also shown high performance in underwater visual enhancement [8, 9] and the fish detection [10–12]. Zhao et al. [13] designed a new composite backbone network (CBResNet) to learn scene change information and improve the accuracy of fish detection by improving ResNet. Villon et al. [14] used GoogLeNet to extract fish body features and Softmax classification method to detect reef fish. Hong Khai et al. [15] proposed an improved Mask R-CNN by classifying image data into three categories of low, medium, and high density, resulting in an enhanced Mask R-CNN model with an accuracy of 97.48%. Knausgård et al. [16] proposed a two-step deep learning method for the detection and classification of temperate fish. The fish detection accuracy, on the pretrained model, reached 99.27%.

In the fish recognition phase, transfer learning [17] has been commonly used to retrain pretrained networks because of the limited data. There are many topologies of networks that have emerged, such as VGG [18], GoogleNet [19], ResNet [20], and ResNeXt [21]. After training these pretrained networks on large datasets, such as ImageNet [22], they are then trained using fish datasets. Xue and Ju [23] added a flexible attention layer to AlexNet and used transfer learning for classification training, which greatly improved the classification effect of fish. Shafait et al. [24] proposed an image set classification paradigm for improving the recognition rate of multiple fish species. Qin et al. [25] first used a deep architecture to extract features from foreground fish images and then used a linear support vector machine (SVM) classifier for classification. An accuracy of 98.64% was obtained on the real-world fish recognition dataset. Tamou et al. [26] uses the pretrained AlexNet network to extract features from the foreground fish images of available underwater dataset, and then uses SVM classifier to classify. The CNN AlexNet is combined with transfer learning to realize the automatic classification of fish species. The accuracy of 99.45% was obtained in the fish recognition ground truth dataset. Rathi et al. [27] used deep learning and image processing to obtain higher discrimination accuracy with an accuracy of 96.29%. Pang et al. [28] uses the processed fish image and the raw fish image to generate the distance matrix, respectively, and reduces the impact of interference on fish classification by reducing the difference between the two distance matrices and extracting the interference information at the feature level. Some other works have been to propose new structures with fewer convolutional layers [29–31].

However, a lot of the researches on underwater fish recognition have focused on classifying fish and little on recognizing fish individual. It also does not address the problem of low-recognition accuracy due to feature occlusion. Therefore, this paper proposes a fish individual recognition method based on coarse and fine-grained feature-linked learning.

3. The Proposed Work

Comparing underwater fish individual recognition to the other biometric methods reveals significant differences. First of all, due to the characteristics of underwater fish movement, fish are more severely obscured from each other, which poses a significant obstacle for the efficient extraction

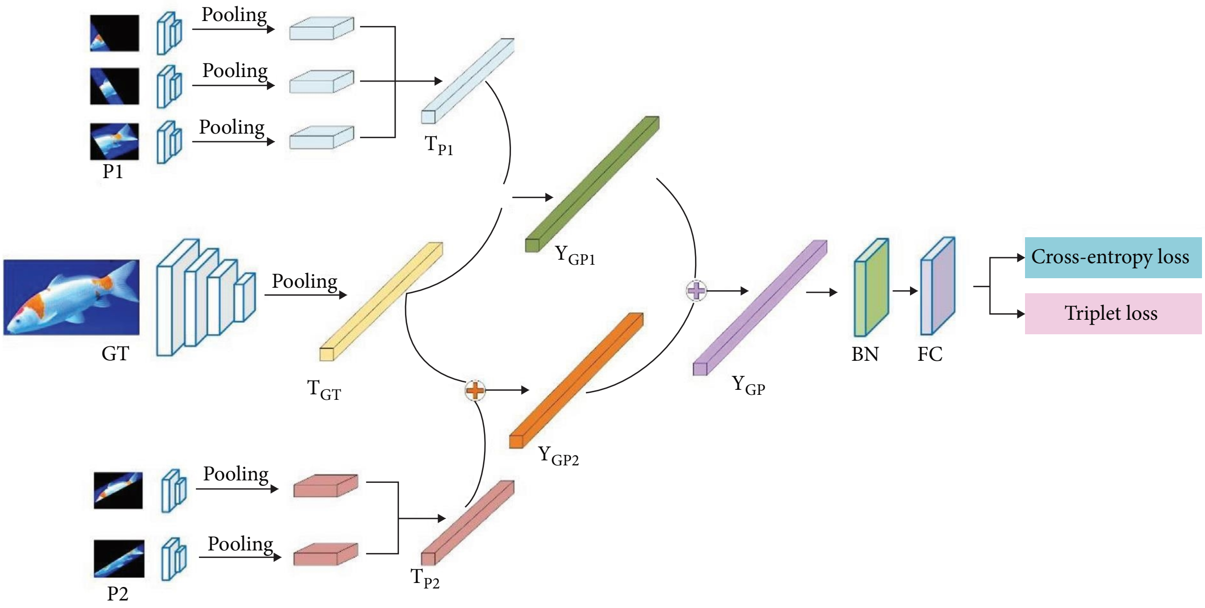


Figure 3: Network structure diagram.

of fish features. Second, in order to accurately recognize various fish, feature extraction methods must be able to detect minute differences between individual animals due to the significant visual similarity of features between individuals of the same species of fish. Finally, the complex and variable underwater environment with uneven illumination makes the acquired images and video data blurred, which makes the effective extraction of fish features more difficult.

To address the above characteristics of fish individual recognition, we proposed a method based on coarse and fine-grained feature-linked learning to further improve the accuracy and robustness of fish individual recognition. The method consists of three parts (Figure 3): a backbone (GT) that learns coarse-grained features and two branches (P1, P2) that learn fine-grained features. The input image of GT is a single fish image obtained by YOLO-V4 [32] target detection, with a size of 256 × 512. The input image of P1 is a head, body and tail part image divided by key points, each with a size of 64 × 64. The input image of P2 is the upper and lower fins, each with the same size of 64 × 64.

In the training process, two local vectors YGP1 and YGP2 with global features are obtained by fuzing the two local feature vectors with the global feature vector, respectively. Then YGP1 and YGP2 are fuzed to obtain a vector YGP with global features and different local features. Finally, the triplet loss and cross-entropy loss are calculated by the normalization layer and the fully connected layer, respectively, so as to achieve the purpose of coarse and fine-grained feature-linked learning.

The feature fusion method is shown in Equations (1–3). The TGT represents the global feature vector, TP1 represents the local feature vector of the head, body, and tail parts, and TP2 represents the local feature vector of the upper and lower fins.

$$\mathrm{YGP1} = \mathrm{TGT} + \mathrm{TP1}, \tag{1}$$

$$\mathrm{YGP2} = \mathrm{TGT} + \mathrm{TP2}, \tag{2}$$

$$\mathrm{YGP} = \mathrm{YGP1} + \mathrm{YGP2}. \tag{3}$$

In the testing process, only the coarse and fine-grained fuzed YGP is used for recognition to reduce the number of parameters while improving the prediction efficiency.

3.1. Coarse and Fine-Grained Feature-Linked Learning.

Coarse-grained features are low-level intuitive features such as color, contour, texture, and overall structure of the image as a whole. They have good invariance and easy computational characteristics, but high dimensionality and large computational efforts are its fatal drawbacks. Compared with coarse-grained features, fine-grained features can better extract the detailed parts of the image, such as curves, edges, corner points, and other special regions. They are abundant and independent in the image and will not affect the learning and extraction of local features due to the occlusion or absence of the other regions. As the number of chunked regions increases, the local information learned from each region becomes more detailed. The design of coarse and fine grained feature-linked learning in this method is described below.

In the process of coarse-grained feature learning, the input image with a size of 256 × 512 extracts the macroscopic features by the backbone network for computing channel weights. After the maximum pooling layer, a feature vector is obtained and is denoted as TGT.

In the process of fine-grained feature learning, we first preprocess the coarse-grained images and divide the image into local images by different regions according to the key

Table 1: Comparison of accuracy using different teacher models.

Layer name	Parameters	Output size
Conv1	7×7,64, stride 2	32×32
Conv2_x	3×3, Max pool, stride 2 $\begin{bmatrix} 1\times 1,\ 128 \\ 3\times 3,\ 128,\ C=32 \\ 1\times 1,\ 256 \end{bmatrix}\times 3$	16×16
Conv3_x	$\begin{bmatrix} 1\times 1,\ 256 \\ 3\times 3,\ 256,\ C=32 \\ 1\times 1,\ 512 \end{bmatrix}\times 4$	8×8
Conv4_x	$\begin{bmatrix} 1\times 1,\ 512 \\ 3\times 3,\ 512,\ C=32 \\ 1\times 1,\ 1024 \end{bmatrix}\times 23$	4×4
Conv5_x	$\begin{bmatrix} 1\times 1,\ 1024 \\ 3\times 3,\ 1024,\ C=32 \\ 1\times 1,\ 2048 \end{bmatrix}\times 3$	2×2
Average pool	2×2	1×1

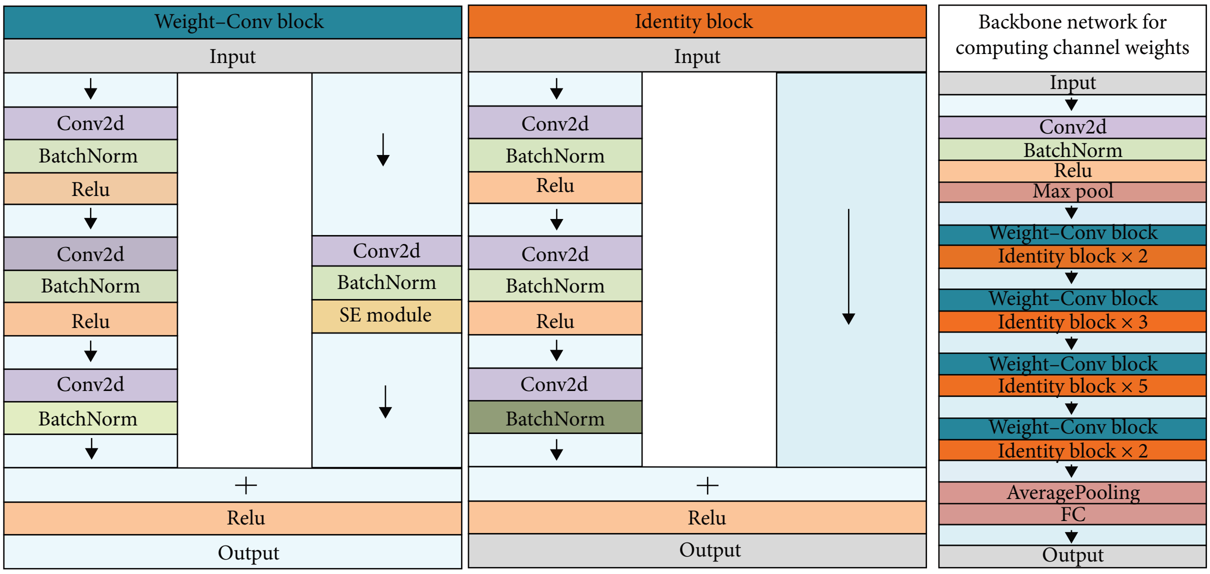


Figure 4: SE-ResNet network structure diagram.

points. After the input image of P1 passes through the maximum pooling layer, three feature vectors are generated, and then the three vectors are concatenated along the channel into a feature vector TP1. After the maximum pooling layer, the input image of P2 obtains two vectors, and a feature vector TP2 is obtained after concatenating.

Unlike the 50-layer network used in the coarse-grained feature learning part, the two fine-grained feature learning branches use a 101-layer network (Table 1), and the network structure is the same for both fine-grained features. The advantage of using different networks with different numbers of layers to learn different regions is that the level of feature abstraction is higher with the appropriate increase in the number of layers. The network with more layers in the spatial dimension can learn more detailed and finer-grained fish features.

3.2. Backbone Network for Computing Channel Weights. To address the problem of low accuracy of fish individual recognition due to uneven illumination and blurred underwater images, in this paper, the SE attention mechanism module [33] is added to the residual edge part of the ResNet-50 network to form the Weight–Conv block. The backbone network structure for computing channel weights is shown in Figure 4. In underwater fish recognition, there are phenomena such as turbid water, overlapping targets, and complex background environments. A backbone network that computes channel weights enables the model to focus more

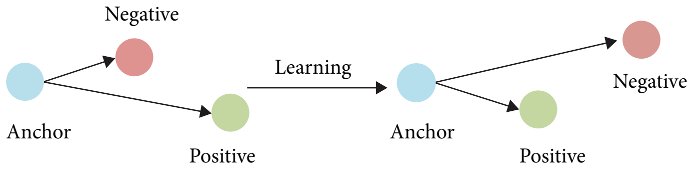


Figure 5: Schematic diagram of triple loss.

on the fish and less on underwater non-fish species such as algae and corals, so as to learn more about the texture features of the fish. For example, when a person sees a picture, he will first focus on the outline of the object in the picture as well as the content, and appropriately ignore the background information. By assigning different weights to each part region of the input image, the SE attention mechanism extracts more crucial and important information, improving resource utilization and model accuracy.

The backbone network for computing channel weights consists of two basic modules, Weight–Conv block and Identity block. Weight–Conv blocks are used to change the network dimension and identity blocks to deepen the network. Weight–Conv block is divided into two parts, the trunk and the residual side, and the trunk is composed of two convolutions, normalization, activation function, and one convolution and standardization; the residual edge part consists of one convolution, normalization, and SE attention mechanism modules. The Weight–Conv block can change the width, height, and number of channels of the output feature layer because of the presence of convolution in the residual edge part.

The identity block is also divided into two parts: trunk and residual edge. The structure of the backbone part is the same as that of the Weight–Conv block part; the difference between the two modules lies in the residual edge part. The residual edge of the identity block has no convolution and is directly connected to the output, so the input feature layer of the identity block has the same size as the output feature layer.

3.3. Loss Function. The loss function measures the accuracy of the model in prediction and affects the convergence of the algorithm, and in the field of target recognition, triplet loss and cross-entropy loss are widely used.

The triplet loss consists of three basic elements, Anchor, Negative, and Positive [34] as shown in Figure 5.

Anchor is a randomly selected sample from the training set, Positive is a positive sample of the same class as Anchor, and Negative is a negative sample of a different class from Anchor. Initially, the distance between Anchor and Positive is much larger than the distance between Anchor and Negative, and the distance between Anchor and Positive is closer to the same class of samples by learning. The triplet loss function is shown in Equation (4).

$$\mathrm{Loss} = \sum_{i}^{N} \left[\left\| f\left(x_i^a\right) - f\left(x_i^p\right) \right\|_2^2 - \left\| f\left(x_i^a\right) - f\left(x_i^n\right) \right\|_2^2 + \alpha \right], \tag{4}$$

x^a, x^p, and x^n denote the Anchor sample, the Positive sample, and the Negative sample. $\left\| f(x_i^a) - f\left(x_i^p\right) \right\|_2^2 \| f(x_i^a) - f\left(x_i^p\right) \|_2^2$ denotes the Euclidean distance measure between Anchor and Positive. $\| f(x_i^a) - f(x_i^n) \|_2^2$ denotes the Euclidean distance measure between Anchor and Negative. The distance between x^a and x^p distance and x^a and x^n distance is denoted by the α. When the distance between x^a and $x^n < x^a$ and x^p spacing sum α, the expression is greater than 0, at which point a loss occurs. When the distance between x^a and $x^n >= x^a$ and x^p spacing sum α, the loss is zero. When the value of α is small, the loss decreases rapidly and the trained results cannot discriminate images with similar features well. When α is large, the loss value will remain in a relatively large range during the training process and it is difficult to converge to zero. Therefore, it is important to choose a suitable α value for the model. In this paper, α is set to 1.0 in the choice of value.

Apart from triplet loss, the cross-entropy loss function is widely used in deep learning for classification tasks to evaluate the prediction accuracy of the classification models. Cross-entropy loss measures the disparity between the predicted probability distribution of a model and the actual labels in classification problems. For a problem with multiple categories, the actual labels of the samples are one-hot encoded vectors, i.e., $p = (0, 0, \ldots, 1, \ldots, 0)$, where the position of the 1 indicates the true class of the sample. Assuming the predicted probability distribution of the model for the sample is $q = (q1, q2, \ldots, qn)$, the cross-entropy loss function is given in Equation (5).A smaller cross-entropy loss value indicates that the model's prediction result is closer to the true label, resulting in a higher classification accuracy of the model.

$$\mathrm{Loss} = -\sum_{i=1}^{n} p(x_i) \log(q(x_i)). \tag{5}$$

In this paper, both these loss functions are used to optimize the network concurrently. The cross-entropy loss enables the model to better match the distribution of the data, while the triplet loss helps the model to better understand the similarity between different individuals within the same specie. As the fish individual recognition task is influenced by the unique underwater noise nature, which may result in noise or incorrect classification labels, utilizing only the cross-entropy loss function may lead to overfitting of the model. Thus, incorporating a triplet loss function can improve the model's ability to capture the variability between

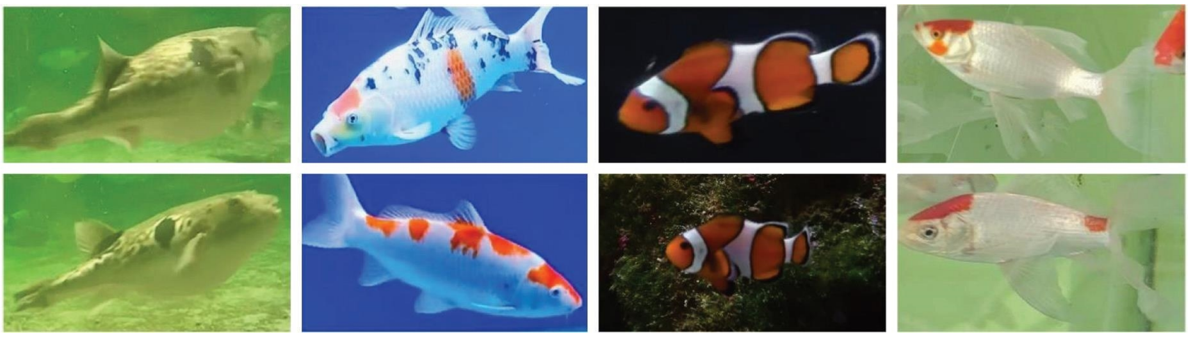


Figure 6: Coarse-grained dataset example diagram.

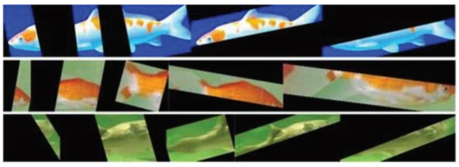


Figure 7: Fine-grained dataset example diagram.

data, thereby enhancing the model's generalization performance. Furthermore, the fish dataset used in this paper includes images of various fish species, necessitating the model to precisely differentiate between fish species while also accounting for the similarity differences among individuals of the same species to species fish individual. Consequently, both loss functions are employed during training to achieve better results in underwater fish individual recognition, as demonstrated in Equation (6). After fusion, the vector YGP with coarse-fine granularity features is calculated as triplet loss and cross-entropy loss, respectively, which are noted as *L*tripletGP and *L*crossGP, and the value of the β in this paper is 0.6.

$$\text{Loss} = \beta\,\text{Ltriple t}_{\text{GP}} + (1 - \beta)\text{Lcross}_{\text{GP}}. \qquad (6)$$

4. Simulation Experiments

4.1. Coarse-Grained Dataset. The coarse-grained dataset of fish used for the experiments in this paper is shown in Figure 6, which includes 1,800 colorful koi, 1,550 red and white koi, 1,850 Takifugu Rubripes, and 1,800 Amphiprioninaes, totaling 7,000 labeled single fish images. The images are randomly disturbed, and divided into a training set and a test set in an 8 : 2 ratio with slightly varying background conditions, lighting, and sharpness. The purpose of randomly placing the images affected by different external environments into the training and test sets is to improve the learning ability of the model during training and to verify the generalization of the model during testing.

4.2. Fine-Grained Dataset. The fine-grained dataset of fish used for the experiments in this paper is shown in Figure 7.

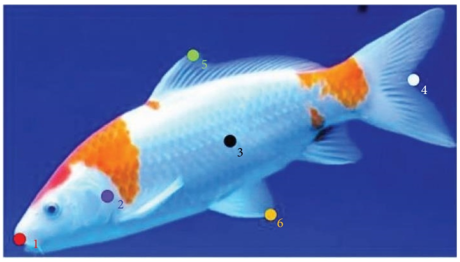


Figure 8: Fish individual key points.

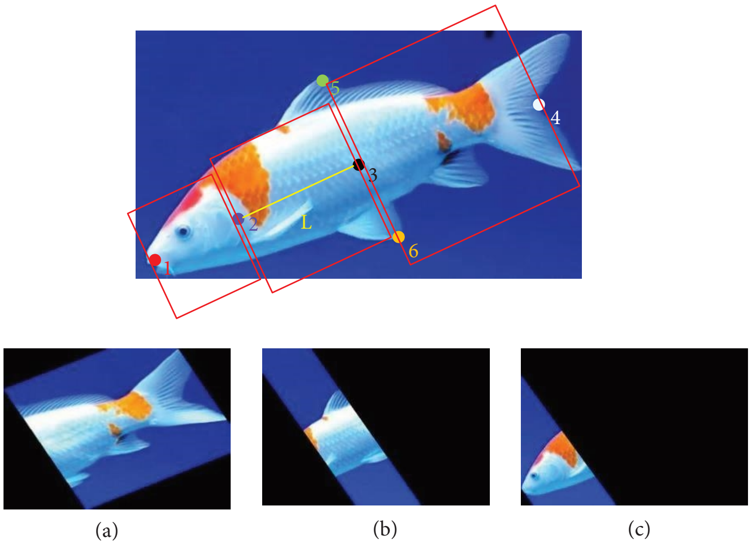
We manually labeled the fish images in the coarse-grained dataset by six key points: mouth, gills, belly, tail, dorsal fin, and ventral fin, as shown in Figure 8.

The fish was chunked according to the key points, and the chunking method we were inspired by Li. In the head–body–tail part, we take the distance between the key points as the length of the rectangle, denoted as *L*, and 1.2*L* as the width of the rectangle, resulting in three fine-grained blocks. As shown in Figure 9, the three images 9(a)–9(c) are used as the input of P1.

In the upper and lower fin sections, we use the distance between the key points as the height of the rectangle, denoted as *H*, and 5*H* as the length of the rectangle, resulting in two fine-grained blocks. The two images as shown in Figures 10(a) and 10(b) are used as the input of P2.

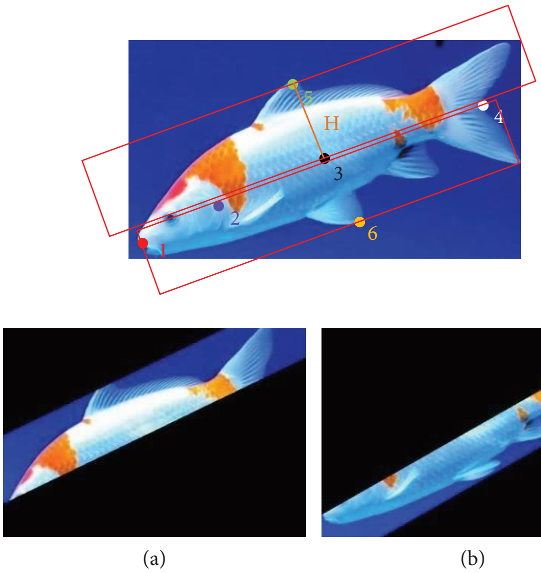
It is worth noting that the key points mentioned in this paper only need to be labeled during the training process, and do not need to be labeled again when new training images are added. The model will automatically chunk the fish according to the optimal weights obtained by training the already labeled key point positions.

4.3. Experimental Setup. All the experiments in this paper were done in Pytorch framework under Ubuntu 20.04 environment, and the training GPU was configured as GeForce RTX 3090. The loss function used was triplet loss and cross-entropy loss, the optimizer used was Adam, and the Batch-size was 32. We also used a warm-up strategy to bootstrap the network to get better performance. We increase the learning rate linearly from 2.5×10^{-4} to 2.5×10^{-3} using 20 epochs. After 20 epochs, the learning rate decays by 0.5 times every 80 epochs, and 500 epochs are trained, with algorithm evaluation metrics of Rank-1 and Rank-5.



(a) (b) (c)

Figure 9: Fine-grained images of head, body and tail after being partitioned by key points: (a) head, (b) body, and (c) tail.



(a) (b)

Figure 10: Fine-grained images of upper and lower fins after being partitioned by key points: (a) upper fin and (b) lower fin.

4.4. Data Preprocessing

4.4.1. Fish Individual Numbering. After video frame extraction, we manually identify and number the identical fish to determine which of the single fish are the same fish. The numbering rule is fish identity number and image number, for example, 000101 would represent the first fish image with identity number 1, and 001010 would represent the 10th fish image with identity number 10. Based on the coherence of the video, we artificially confirm the fish identity information. When getting the prediction results, we compare them by the identity number to record the accuracy of the prediction. This solves the problem that underwater species recognition cannot judge the target information based on the background environment.

4.4.2. Data Augmentation. To simulate the actual conditions, such as the fish's irregular swimming posture, various underwater illumination conditions, and blurred water quality, we add different degrees of random rotation and grayscale transformation and added Gaussian noise in the range of 0.01–0.2 to the fish images in the coarse-grained dataset. Considering that different sides of the fish have different texture features, the data augmentation method of flipping the images horizontally is not used in this paper. The coarse-grained dataset after data augmentation is shown in Figure 11.

4.5. Comparison of the Impact of Fine-Grained Feature Learning on Network Performance. The fish individual recognition method proposed in this paper includes one coarse-

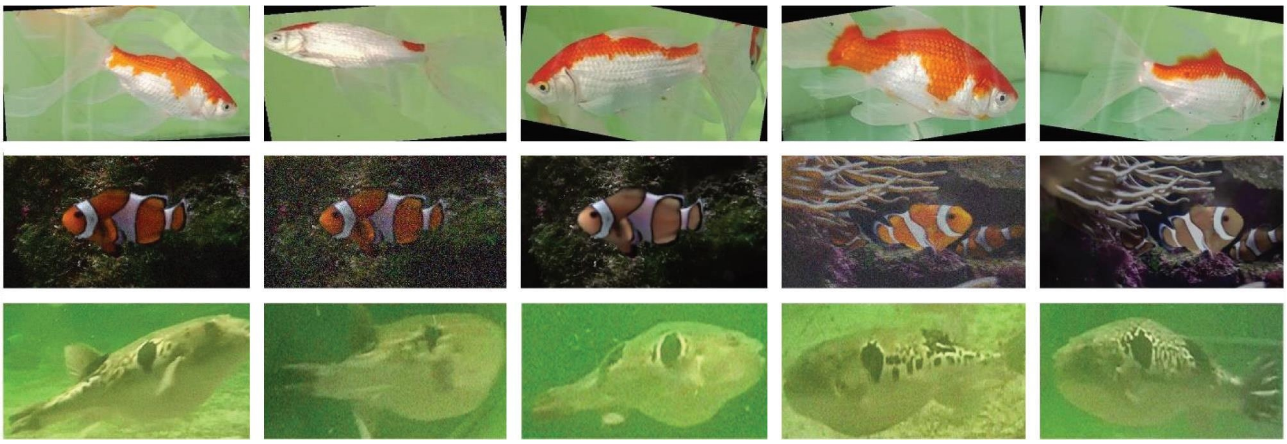


Figure 11: Coarse-grained dataset after data augmentation.

Table 2: Comparison of the impact of different fine-grained features on network accuracy in the colorful koi dataset.

Coarse-grained feature	Head, body, and tail fine-grained features	Upper and lower fins fine-grained features	Rank-1
√			88.3%
√		√	92.1%
√	√		93.5%
√	√	√	**97.4%**

Bold font indicates the method achieves the best results with the highest accuracy.

Table 3: Comparison of training and experimental data under the same dataset.

Dataset	Colorful koi	Red and white koi	Amphiprioninae
Rank-1	97.4%	96.7%	96.9%
Rank-5	98.6%	97.8%	97.5%

grained feature learning and two fine-grained feature learning, a total of three parts. Fine-grained feature learning was performed in the head, body, tail, and upper and lower fins.

In this section, we eliminate the coarse-grained feature fusion for the head, body, and tail parts and the coarse-grained feature fusion for the upper and lower fin parts. The effect of different fine-grained feature learning on the accuracy of the network in recognizing fish individuals was tested and the results are shown in Table 2.

Taking the colorful koi dataset as an example, the accuracy of Rank-1 was improved by 3.8% and 5.2% by fuzing the fine-grained features of two different parts with the coarse-grained features, respectively. The accuracy of Rank-1 was improved by 9.1% by fuzing the two parts with the trunk features at the same time, and the accuracy of Rank-1 reached 97.4%.

4.6. Model Generalization Capability Analysis. In this section, we analyze the experimental results when the training dataset and the test dataset belong to the same and different fish species, respectively, to verify the performance of the model in recognizing other species of fish.

4.6.1. Analysis of Experimental Results under the Same Fish Dataset. Three datasets, colorful koi, red, and white koi and Amphiprioninaes, were used as examples to analyze the model performance when the training dataset and the test dataset belonged to the same species of fish. As shown in Table 3, and the results are shown in Figure 12.

The fish individual numbers obtained by the experiment are as follows, where query_id represents the fish individual number to be queried; ans_ids represents the fish individual numbers recognized by the network after training as the same fish as the queried fish. The first of the three digits represents the individual number, and the last two represent the picture number. Similarly, the first two of the four digits represent the individual number, and the last two represent the picture number.

All fish numbers provided by the network when recognizing the fish individual with the query number 814 are different images of the same fish with ID 8; all fish numbers provided by the network when recognizing the fish individual with query number 904 are different images of the same fish with ID 9; all fish numbers provided by the network when recognizing the fish individual with query number 304 are different images of the same fish with ID 3.

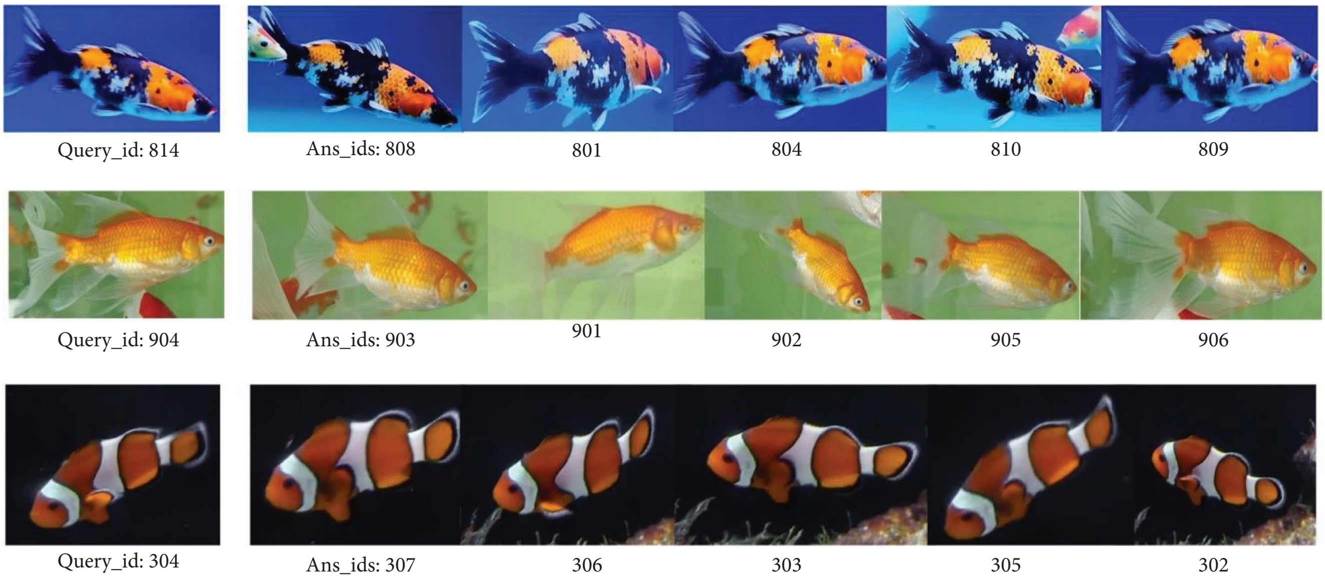


Figure 12: Experimental effects when the training and test data sets belong to the same species of fish.

Table 4: Comparison of training and experimental data under the different datasets.

Dataset	Colorful koi	Red and white koi	Amphiprioninae
Rank-1	97.4%	96.7%	96.9%
Rank-5	98.6%	97.8%	97.5%

The experimental results show that the model performs over 96.5% on both Rank-1 and Rank-5 indicators when the training dataset and the test dataset belong to the same species of fish.

4.6.2. Analysis of Experimental Results under the Different Fish Dataset. We used the model trained on the colorful koi dataset to recognize red and white koi, Takifugu Rubripes, and Amphiprioninaes. The red and white koi are the same species as the colorful koi, but with large differences in stripe characteristics. Takifugu Rubripes and Amphiprioninaes are different species from colorful koi species. The accuracy is shown in Table 4, and the experimental effect is shown in Figure 13.

All fish numbers provided by the network when recognizing the fish individual with the query number 1,705 are different images of the same fish with ID 17; the first four fish numbers provided by the network when recognizing the fish individual with the query number 208 are all different images of the same fish with ID 2, while the fifth one gives an image of the fish with ID 6; the first three fish numbers provided by the network when recognizing the fish individual with the query number 304 are all different images of the same fish with ID 3, while the fourth and fifth positions are given to images of the fish with ID 11.

The experimental results show that the model still performs well under different datasets, with accuracy above 90%. The effect graph shows that the first five fish on the red and white koi dataset are all hits, the first four hits on the Takifugu Rubripes dataset are correct, and the first three hits on the Amphiprioninaes dataset are precise. It is clear that the model does better at recognizing fish with distinct textural features and also does well at recognizing fish across species with high generalization.

4.7. Analysis of Fish Recognition Results in the Complex Underwater Environments. In order to verify the accuracy and stability of the model in the complex underwater environment, we used individual images of different species of fish with low definition, irregular fish swimming and feature obscuration as the test set. The experimental results are shown in Tables 5–6. The partial diagram of the test set used is shown in Figure 14.

In this section, Takifugu Rubripes, colorful koi with irregular swimming postures, and red and white koi with obscured features are used as test sets for fish individual recognition. The experimental results show that the model can still guarantee more than 95% accuracy under different levels of external environmental interference. Among them, Rank-1 achieves 97.3% in the case of random fish individual swimming postures.

The level of image clarity has a larger impact on the model's recognition accuracy than swimming position or feature occlusion, yet Rank-5 accuracy is only decreased by 0.6%. In the experiments simulating the complex underwater environment, the proposed method can still efficiently and accurately accomplish the task of individual recognition of target fish even if there are low clarity, large differences in fish swimming posture and fish feature occlusion.

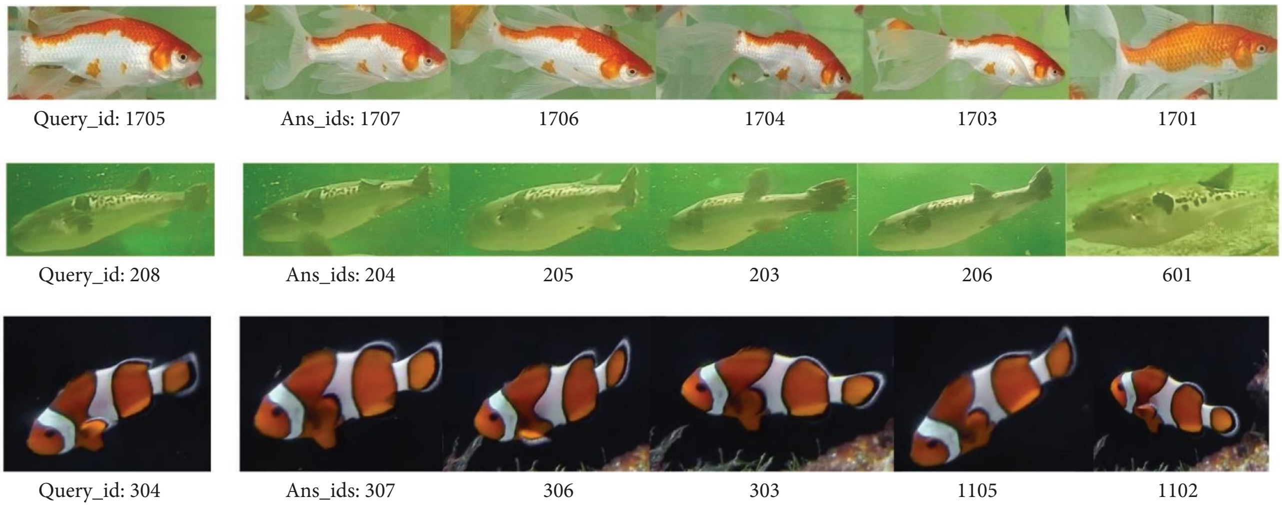


Figure 13: Experimental effect picture under the different datasets.

Table 5: Comparison of accuracy under low-definition Takifugu Rubripes test set.

Dataset	High definition	Low definition	Accuracy variation
Rank-1	**96.3%**	95.5%	0.8%
Rank-5	**97.8%**	97.2%	0.6%

Bold font indicates the method achieves the best results with the highest accuracy.

Table 6: Comparison of accuracy under the test set of red and white koi with a small amount of occlusion.

Dataset	Features complete	Feature occlusion	Accuracy variation
Rank-1	**96.7%**	96.4%	0.3%
Rank-5	**97.8%**	97.5%	0.3%

Bold font indicates the method achieves the best results with the highest accuracy.

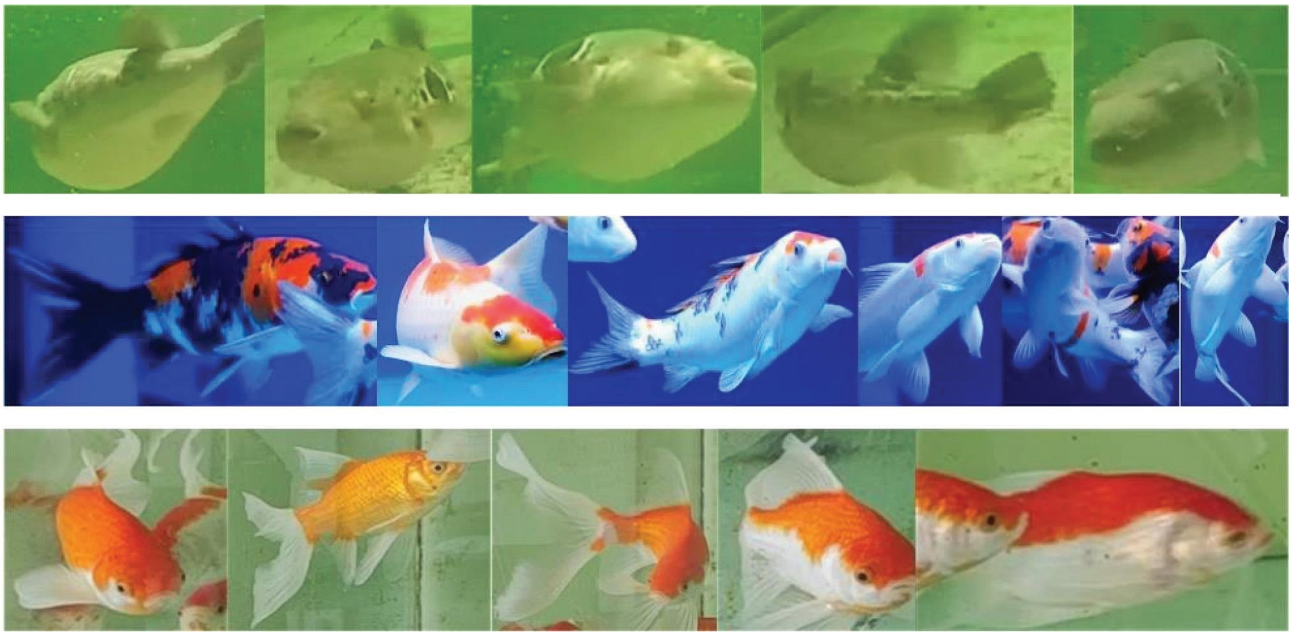


Figure 14: Lower quality fish individual test set.

Table 7: Comparison of training experimental data using computing channel weighting network.

Network	ResNet-50	Computing channel weighting network
Parameters	**25.6 M**	25.7 M
Rank-1	94.7%	**97.4%**
Rank-5	95.3%	**98.6%**

Bold font indicates the method achieves the best results with the highest accuracy.

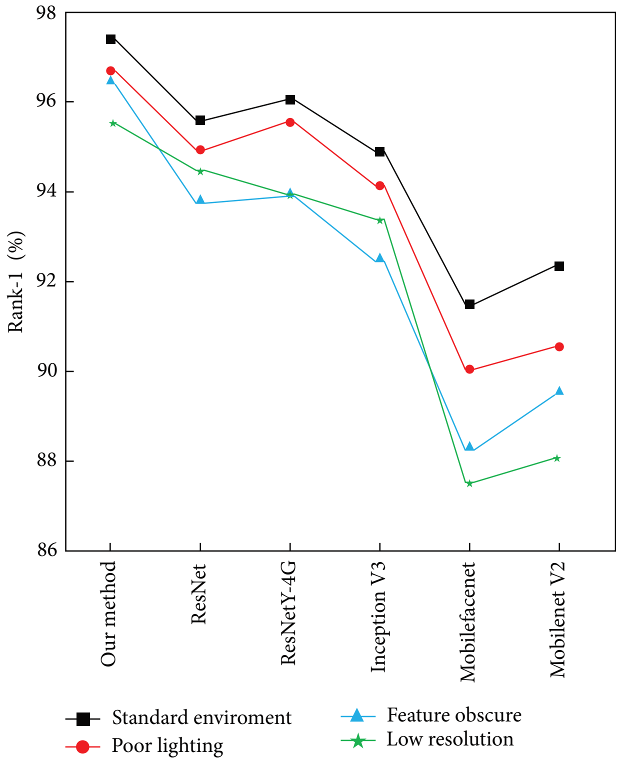


Figure 15: Comparison of different network experimental results in the same environment.

4.8. Experimental Results and Analysis under Different Networks

4.8.1. Experimental Results and Analysis of Backbone Network for Computing Channel Weights. We take the network as variables, the number of network layers and the dataset as constants, compare the performance of ResNet 50 and the backbone network for computing channel weights proposed in this paper under the colorful koi dataset from the aspects of parameter quantity, Rank-1 accuracy and Rank-5 accuracy. The experimental results are shown in Table 7. The results show that the backbone network for computing channel weights is able to significantly increase the model accuracy at the cost of a small increase in the number of parameters, with 2.7% increase in Rank-1 accuracy and 3.3% increase in Rank-5 accuracy compared to the ResNet-50 network.

4.8.2. Comparison of the Method Proposed in This Paper with Other Networks. We compare the proposed method with other networks to analyze the performance of the model in different underwater environments under the same dataset. The experimental results are shown in Figure 15, where the standard environment is fully lit, the fish feature is unobstructed, and the image resolution is clear.

The experiments are evaluated using Rank-1 as the evaluation criterion, and it is clear from the results that our proposed method has the highest accuracy in four different external environmental conditions. The accuracy of the model is affected when there is a change in the underwater environment. However, our proposed method is less affected by the environmental changes and still guarantees more than 95.5% accuracy under insufficient lighting conditions, obscured fish features, and low-image resolution. This is due to our coarse and fine-grained linked learning approach, which provides higher accuracy and greater stability than other networks.

5. Conclusions

In this paper, we proposed a fish individual recognition method based on coarse and fine-grained feature-linked learning. By chunking the fish by different positions and training the part and the whole image simultaneously, the learned fine-grained features are fuzed with the coarse-grained features to achieve the purpose of linked learning of coarse and fine-grained features. Additionally, we improve the network's capacity to extract fish features by computing channel weights, which increases the network's accuracy in recognizing fish individuals even when the background environment is blurry or similar. The method performs remarkably well in various datasets. It is capable of performing cross-species fish recognition tasks with great generalizability after data augmentation for underwater environment specific and fish features. The method can be combined with intelligent devices such as underwater robots and detectors to accomplish target capture, tracking and observation, and recording fish information. This has scientific significance for the development of aquaculture and environmental protection industries in the current environment.

Data Availability

The data presented in this study are available on request from the corresponding author. The data are not publicly available due to part of the data are provided by the cooperative enterprise.

Conflicts of Interest

The authors declares that there is no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was funded by National Natural Science Foundation of China (31972846), Key Laboratory of Environment Controlled Aquaculture (Dalian Ocean University) Ministry of Education (202205, 202315), and National Key Research and Development Program of China (2021YFB2600200).

References

[1] M. Körschens, B. Barz, and J. Denzler, "Towards automatic identification of elephants in the wild," arXiv preprint arXiv: 1812.04418, 2018.

[2] S. Li, J. Li, H. Tang, R. Qian, and W. Lin, "ATRW: a benchmark for Amur tiger re-identification in the wild," in *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 2590–2598, Association for Computing Machinery, New York, NY, USA, October 2020.

[3] B. Boom, P. X. Huang, C. Beyan et al., "Long-term underwater camera surveillance for monitoring and analysis of fish populations," in *2012 21st international conference on pattern recognition (ICPR 2012). Red Hook: Curran Associates, Inc.*, January 2012.

[4] J. Yin, J. Wu, C. Gao, and Z. Jiang, "LIFRNet:a novel lightweight individual fish recognition method based on deformable convolution and edge feature learning," *Agriculture*, vol. 12, no. 12, Article ID 1972, 2022.

[5] B. Jian, Y. Ling, X. Zhang, and J. Ou, "Computer image recognition and recovery method for distorted underwater images by structural light," *Journal of Physics: Conference Series*, vol. 2083, Article ID 042019, 2021.

[6] R. A. Pramunendar, S. Wibirama, P. I. Santosa, P. N. Andono, and M. A. Soeleman, "A robust image enhancement techniques for underwater fish classification in marine environment," *International Journal of Intelligent Engineering and Systems*, vol. 12, no. 5, pp. 116–239, 2019.

[7] M.-C. Chuang, J.-N. Hwang, and K. Williams, "A feature learning and object recognition framework for underwater fish images," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1862–1872, 2016.

[8] K. Hu, C. Weng, Y. Zhang, J. Jin, and Q. Xia, "An overview of underwater vision enhancement: from traditional methods to recent deep learning," *Journal of Marine Science and Engineering*, vol. 10, no. 2, Article ID 241, 2022.

[9] C. Edge, M. J. Islam, C. Morse, and J. Sattar, "A generative approach for detection-driven underwater image enhancement," arXiv preprint arXiv: 2012.05990, 2020.

[10] X. Li, M. Shang, H. Qin, and L. Chen, "Fast accurate fish detection and recognition of underwater images with fast R-CNN," in *OCEANS 2015—MTS/IEEE Washington*, pp. 1–5, IEEE, Washington, DC, October 2015.

[11] A. Jalal, A. Salman, A. Mian, M. Shortis, and F. Shafait, "Fish detection and species classification in underwater environments using deep learning with temporal information," *Ecological Informatics*, vol. 57, Article ID 101088, 2020.

[12] D. Zhang, N. E. O'Conner, A. J. Simpson, C. Cao, S. Little, and B. Wu, "Coastal fisheries resource monitoring through a deep learning-based underwater video analysis," *Estuarine, Coastal and Shelf Science*, vol. 269, Article ID 107815, 2022.

[13] Z. Zhao, Y. Liu, X. Sun, J. Liu, X. Yang, and C. Zhou, "Composited FishNet: fish detection and species recognition from low-quality underwater videos," *IEEE Transactions on Image Processing*, vol. 30, pp. 4719–4734, 2021.

[14] S. Villon, D. Mouillot, M. Chaumont et al., "A deep learning method for accurate and fast identification of coral reef fishes in underwater images," *Ecological informatics*, vol. 48, pp. 238–244, 2018.

[15] T. Hong Khai, S. N. H. S. Abdullah, M. K. Hasan, and A. Tarmizi, "Underwater fish detection and counting using mask regional convolutional neural network," *Water*, vol. 14, no. 2, Article ID 222, 2022.

[16] K. M. Knausgård, A. Wiklund, T. K. Sørdalen et al., "Temperate fish detection and classification: a deep learning based approach," *Applied Intelligence*, vol. 52, pp. 6988–7001, 2022.

[17] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv: 1409.1556, 2014.

[19] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, IEEE, Boston, MA, USA, June 2015.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, IEEE, Las Vegas, NV, USA, June 2016.

[21] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5987–5995, IEEE, Honolulu, HI, USA, July 2017.

[22] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and F. F. Li, "ImageNet: a large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, IEEE, Miami, FL, USA, June 2009.

[23] Y. Xue and Z. Ju, "Fish recognition algorithm based on improved AlexNet," *Electronic Science and Technology*, vol. 34, no. 4, pp. 12–17, 2021.

[24] F. Shafait, A. Mian, M. Shortis et al., "Fish identification from videos captured in uncontrolled underwater environments," *ICES Journal of Marine Science*, vol. 73, no. 10, pp. 2737–2746, 2016.

[25] H. Qin, X. Li, J. Liang, Y. Peng, and C. Zhang, "DeepFish: accurate underwater live fish recognition with a deep architecture," *Neurocomputing*, vol. 187, pp. 49–58, 2016.

[26] A. B. Tamou, A. Benzinou, K. Nasreddine, and L. Ballihi, "Underwater live fish recognition by deep learning," in *International Conference on Image and Signal Processing*, vol. 10884, Springer, Cham, June 2018.

[27] D. Rathi, S. Jain, and S. Indu, "Underwater fish species classification using convolutional neural network and deep learning," in *2017 Ninth international conference on advances in pattern recognition (ICAPR)*, pp. 1–6, IEEE, Bangalore, India, December 2017.

[28] J. Pang, W. Liu, B. Liu, D. Tao, K. Zhang, and X. Lu, "Interference distillation for underwater fish recognition," in *Asian Conference on Pattern Recognition*, C. Wallraven, Q. Liu, and H. Nagahara, Eds., vol. 13188, Springer, Cham, 2022.

[29] H. Qin, X. Li, Z. Yang, and M. Shang, "When underwater imagery analysis meets deep learning: a solution at the age of big visual data," in *OCEANS 2015—MTS/IEEE Washington*, pp. 1–5, IEEE, Washington, DC, USA, October 2015.

[30] A. Salman, A. Jalal, F. Shafait et al., "Fish species classification in unconstrained underwater environments based on deep learning," *Limnology and Oceanography Methods*, vol. 14, no. 9, pp. 570–585, 2016.

[31] M. Paraschiv, R. Padrino, P. Casari et al., “Classification of underwater fish images and videos via very small convolutional neural networks,” *Journal of Marine Science and Engineering*, vol. 10, no. 6, Article ID 736, 2022.

[32] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “Yolov4: optimal speed and accuracy of object detection,” arXiv preprint arXiv: 2004.10934, 2020.

[33] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141, IEEE, Salt Lake City, UT, USA, 2018.

[34] F. Schroff, D. Kalenichenko, and J. Philbin, “FaceNet: a unified embedding for face recognition and clustering,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815–823, IEEE, Boston, MA, USA, 2015.