Journal of
*Marine Science
and Engineering*

MDPI

Article

Accurate Fish Detection under Marine Background Noise Based on the Retinex Enhancement Algorithm and CNN

Yanhu Chen *, Yucheng Ling and Luning Zhang

The State Key Laboratory of Fluid Power & Mechatronic Systems, Zhejiang University, Hangzhou 310027, China; yuchengling@zju.edu.cn (Y.L.); 22160055@zju.edu.cn (L.Z.)

* Correspondence: yanhuchen@zju.edu.cn; Tel.: +86-13777879382

Abstract: Underwater detection equipment with fish detection technology has broad application prospects in marine fishery resources exploration and conservation. In this paper, we establish a multi-scale retinex enhancement algorithm and a multi-scale feature-based fish detection model to improve underwater detection accuracy and ensure real-time performance. During image preprocessing, the enhancement algorithm combines the bionic structure of the fish retina and classical retinex theory to filter out underwater environmental noise. The detection model focuses on improving the detection performance on small-size targets using a deep learning method based on a convolutional neural network. We compare our method to current mainstream detection models (Faster R-CNN, RetinaNet, YOLO, SSDetc.), and the proposed model achieves better performance, with a mean Average Precision (mAP) of 78.31% and a mean Miss Rate (mMR) of 54.11% in the open fish image data set. The test results for the data from the field experiment prove the feasibility and stability of our model.

Keywords: fish detection; deep learning; image preprocessing; convolutional neural network

check for updates

Citation: Chen, Y.; Ling, Y.; Zhang, L. Accurate Fish Detection under Marine Background Noise Based on the Retinex Enhancement Algorithm and CNN. *J. Mar. Sci. Eng.* **2022**, *10*, 878. https://doi.org/10.3390/jmse10070878

Academic Editors: Giuseppa Buscaino, Piermattei Viviana, Francesco Rende, Marina Penna and Simone Bonamano

Received: 15 April 2022
Accepted: 21 June 2022
Published: 27 June 2022

1. Introduction

The ocean is the birthplace of life on earth, and it contains abundant resources. With the increasing shortage of land resources, it is more and more urgent to explore marine resources [1]. Recently, underwater detection technologies have drawn remarkable attention for use in resource exploration. Optical sensing is a critical information acquisition source of underwater detection equipment due to its rich and intuitive perception information [2]. Object detection based on optical images is one of the key technologies that make underwater detection equipment intelligent. It facilitates the development of marine fishery resource detection, marine mineral resource detection, and submarine communication cable laying. Object detection is a very important research direction in the fields of computer vision, machine learning, and pattern recognition. Currently, target detection technology is mainly divided into the two-step target detection method, which is based on the region proposal, and the proposal-free method.

Region-based convolutional neural networks (R-CNN), which were originally proposed by Girshick et al. [3], add a region proposal method for object detection based on convolutional neural networks. It first performs a selective search on the input image to extract candidate regions that contain targets in the embodiment. It then conducts convolution operations in each candidate region through the CNN to extract a fixed-length feature vector. Then, the feature vector of each candidate region is input into a Support Vector Machine [4] (SVM) to make a binary classification decision. Finally, bounding box regression is adopted to improve the detection results. However, this method has the following problems: one is that scaling the candidate regions to a fixed size directly causes the aspect ratio of the detection target to become unbalanced, which may cause the loss of local details on the detection target. The other is that there may be repeated overlaps among the candidate regions, which causes the feature to be extracted repeatedly and seriously reduces the overall computational efficiency. He Kaiming et al. [5] proposed the Spatial Pyramid Pooling (SPP) algorithm to solve the low computational efficiency problem in R-CNN and the fixed size of candidate regions. The SPP algorithm performs a convolution calculation on the original image of the input image to obtain the feature map of the entire image, and then finds the corresponding mapping of each candidate box in the total feature map to improve the overall computing efficiency. Additionally, multi-scale pooling is used to replace the original single pooling to solve the problems that arise from a loss of detailed information. However, the algorithm process is too complicated and requires a lot of storage space.

Based on R-CNN and SPP, Girshick et al. [6] proposed Fast R-CNN with an ROI Pooling layer, which allows the model to obtain a feature map of the complete image with only one convolution calculation and output two vectors after the fully connected layer is processed. One of the vectors is used for Softmax classification, and the other vector is used for border regression. However, Fast R-CNN still has the problem of a low detection speed.

Ren et al. [7] proposed a region proposal network (RPN) to shorten the computing time. This method transfers the task of finding the target candidate regions to the RPN, which significantly improves the target detection speed. However, due to the deep extraction of the target candidate regions, one of the problems with this algorithm is the loss of target detailed features, resulting in poor positioning performance and the poor detection of small-sized targets.

Because target detection algorithms based on proposed regions need to construct the target candidate regions in advance, the calculation speed of this type of detection algorithm cannot meet the real-time detection requirements, the proposal-free algorithm was created. Redmon et al. [8] proposed a proposal-free target detection algorithm, YOLO, that does not require a manual design to extract features. It uses a separate convolutional network that is able to predict the position of the target box and the category of the target in the global features of the image. YOLO transforms the target detection problem into a regression problem and dramatically improves the calculation efficiency. However, due to the method of predicting the target box, which involves dividing the grid area, the algorithm has poor detection effects with adjacent small-sized targets and a poor generalization ability for new or abnormal targets.

To improve YOLO, Redmon et al. [9] proposed YOLOv2, which uses multi-scale training and enhances the resolution of the classifier to increase the detection accuracy. It uses a new joint training algorithm to strengthen the robustness of overall target detection. Compared to proposal-free methods, methods that are dependent on region proposals are more accurate but also have a lower calculation rate. Thus, this paper proposes a detection method that can identify fish features to improve detection accuracy and ensure real-time performance.

However, progress in marine object detection research is far behind land object detection. Zhang et al. [10] conducted ship detection via the segmentation of SAR images, which was effective in nearly all weather conditions as well as during both day and night. Yasin et al. [11] proposed an improved signal denoising method and applied sound waves for the target location. The above techniques have lower precision than the optical positioning systems. However, for the optical positioning system, there are also unfavorable factors, such as light scattering, refraction and absorption effects, and the existence of underwater floating objects that interfere with the image quality in the underwater environment, which causes problems such as background noise, color distortion and low contrast in underwater images. Moreover, due to the massive differences in the optical environment, high requirements are proposed to ensure the robustness of target detection for underwater images and tracking algorithms. Many object detection methods that have been successfully applied on land are not necessarily suitable for underwater environments. To facilitate the development of marine fishery resource detection methods, research for underwater target detection, and tracking technology is of significant importance.

Wang et al. [12] proposed a method for the real-time detection and tracking of normally behaving porphyry seabream. Li et al. [13] applied Fast R-CNN for the detection and recognition of fish species from underwater images with an emulation experiment. Cai et al. [14] combined YOLOv3 with MobileNet for fish detection on a real breeding farm. Kottursamy [15] proposed a solution for underwater image detection techniques in which features are deeply extracted by multi-scale CNN to attain higher accuracy when detecting fish features from input images with the help of the segmentation process. The above methods have good accuracy, but their real-time performance is not satisfactory. Sung et al. [16] proposed convolutional neural network-based techniques based on the YOLO. However, these methods only work if the condition of the target, water quality, and light and background changes are met when there are marine fish with different shapes and when the marine environment is extremely complex. D. Levy et al. [17] demonstrated a method using RetinaNet for detection and the Simple Online Realtime Tracker algorithm for tracking which worked well on their datasets (above and under water). In addition, most of the existing work either deals with a small dataset of a small number of species [15–20] or has low accuracy, robustness, or poor real-time capability [13,21–23].

In this paper, we have designed a fish detection system with an improved preprocessing module and a multi-scale fish detection module that is especially efficient for fish detection and recognition under conditions with marine background noise.

The rest of this paper is structured as follows: Section 2.1 explains the research methodology of the multi-scale retinex enhancement algorithm. Section 2.2 explains the fish detection ability of the system. Section 3 introduces the experimental set up and provides experimental results. In the end, Section 4 discusses the conclusions and future directions.

2. Materials and Methods

2.1. The Multi-Scale Retinex Enhancement Algorithm

Our multi-scale fish detection system, which is designed for fish detection and recognition under conditions with marine background noise, comprises two modules. The first module is the image preprocessing module, which filters out the underwater background noise. The second module is the fish detection module, which is suitable for multi-scale feature-based fish detection. The detection system performs image enhancement and other preprocessing operations and then loads the convolutional neural network model designed in this paper to achieve the fish recognition function. In the first module (Section 2.1), we develop algorithms to filter out the underwater background noise by combining the bionic structure of the fish retina and classical retinex theory. In the section module (Section 2.2), we establish a multi-scale feature-based fish detection model to solve the poor ability of the model to detect fish and other small objects.

The classic retinex theory is based on the bionics theory, which was inspired by the color constancy theory, which is based on the perception and response behavior of the human retinal visual system to the color and brightness of external objects [24,25]. Because the retinal vision system of fish has unique advantages at underwater optical spatial resolutions as well as advantages in terms of contrast sensitivity and color discrimination sensitivity [26], we combined the bionic structure of the fish retina and the classic retinex theory to propose a multi-scale retinex enhancement algorithm to achieve clearer visibility and a higher dynamic range. The algorithm flow chart is shown in Figure 1.

According to the above principle, the multi-scale retinex enhancement algorithm includes a photoreceptor cell simulation algorithm, a horizontal cell simulation algorithm, a bipolar cell simulation algorithm, a ganglion cell simulation algorithm, a color gamut adaptive adjustment algorithm, and a bright and dark channel fusion algorithm.

Photoreceptor cells are responsible for converting the light signals that are received into corresponding neural signals. According to the sensitivity to the spectral wavelength of the received optical signal, the photoreceptor cells can be further divided into long-(R) type, medium-(G), and short-(B) type photoreceptor cells [27]. These three cells correspond to the three-color channels of the color domain (R, G, B). As much, we have $f_R(x,y)$, $f_G(x,y)$, and $f_B(x,y)$, and the signal $L(x,y)$ is the average summation result of the channels.

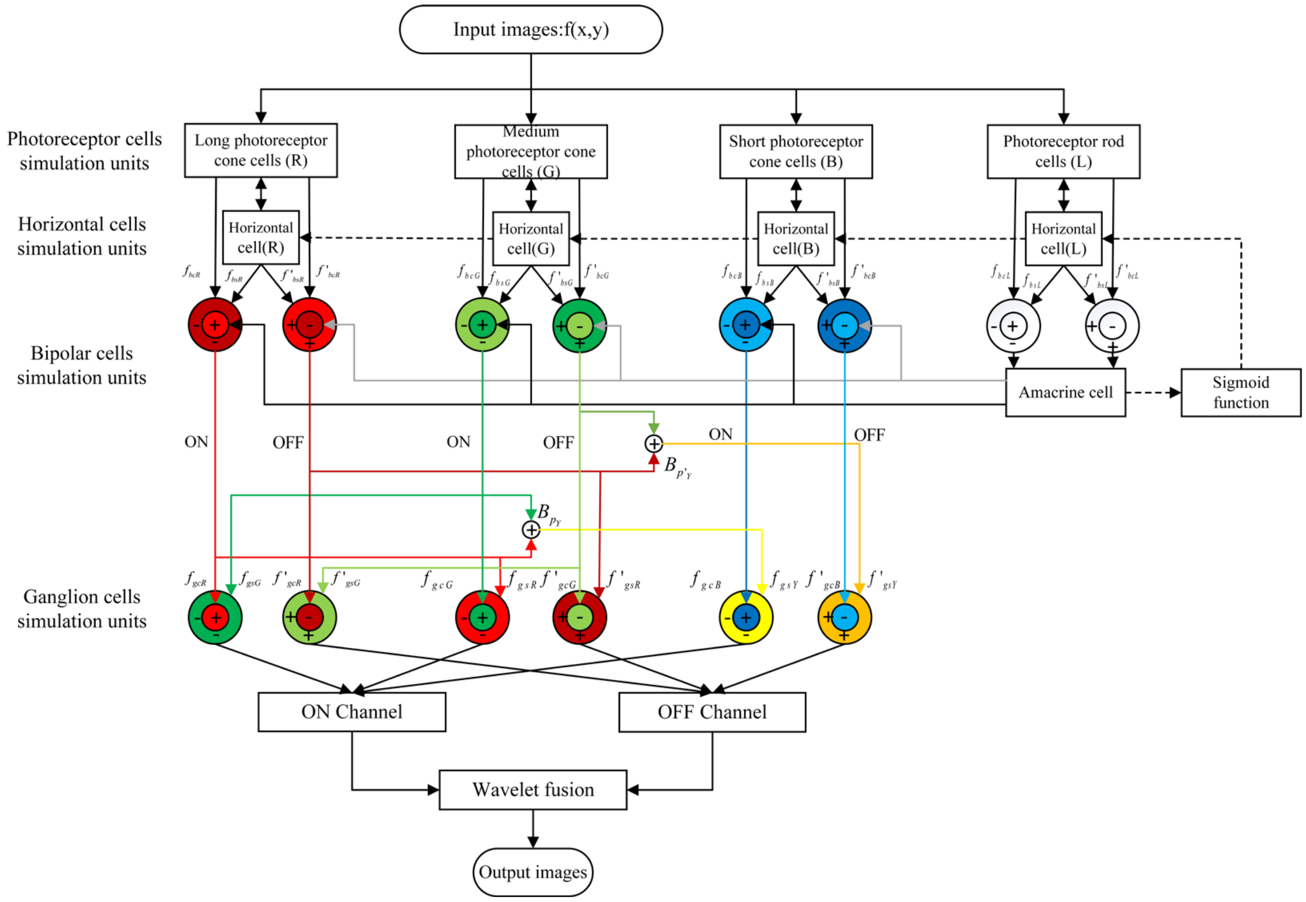
Input images:f(x,y)
Photoreceptor cells simulation units
Long photoreceptor cone cells (R)
Medium photoreceptor cone cells (G)
Short photoreceptor cone cells (B)
Photoreceptor rod cells (L)
Horizontal cells simulation units
Horizontal cell(R)
Horizontal cell(G)
Horizontal cell(B)
Horizontal cell(L)
f_{bcR} f_{bsR} f'_{bsR} f'_{bcR}
f_{bcG} f_{bsG} f'_{bsG} f'_{bcG}
f_{bcB} f_{bsB} f'_{bsB} f'_{bcB}
f_{bcL} f_{bsL} f'_{bsL} f'_{bcL}
Bipolar cells simulation units
Amacrine cell
Sigmoid function
ON
OFF
ON
OFF
ON
OFF
$B_{p'_Y}$
B_{p_Y}
Ganglion cells simulation units
f_{gcR} f_{gsG} f'_{gcR} f'_{gsG}
f_{gcG} f_{gsR} f'_{gcG} f'_{gsR}
f_{gcB} f_{gsY} f'_{gcB} f'_{gsY}
ON Channel
OFF Channel
Wavelet fusion
Output images

Figure 1. The flow chart of the multi-scale retinex enhancement algorithm. The photoreceptor cells first receive the external light and then generate corresponding nerve signals. The horizontal cells are responsible for receiving the nerve signals of the photoreceptor cells. The receptive field of the bipolar cells is divided into two types: the central area and the outer circle area. Amacrine cells are responsible for normalizing the neural signals from the bipolar cells. Ganglion cells also have a center-outer receptive field structure in the shape of a concentric circle.

The receptive field of horizontal cells is simulated with a local mean filter to solve the problem of uneven color changes in underwater images. Moreover, considering that the retinal structure of marine fish is more sensitive to longer light wavelengths and that the red component is much weaker than the blue and green components, the horizontal cell feedback of the three-color channels is as follows:

$$\begin{cases} HCF_R(x,y) = \frac{\Sigma f_R(x,y)}{N^2}, f_R > \theta \\ HCF_G(x,y) = \frac{\Sigma f_G(x,y)}{N^2} \\ HCF_B(x,y) = \frac{\Sigma f_B(x,y)}{N^2} \end{cases}, \quad (1)$$

where θ is used to control the local brightness of the $N \times N$ window in the red channel. The current single-scale retinex enhancements result in the light source having a uniform color distribution area, which limits underwater image recovery enhancement. By adding horizontal cell feedback, we have $f_\lambda(x,y)$ divided by $HCF_\lambda(x,y)$. Moreover, in the retinal structure of marine fish, if the fish is in a dark environment, the amacrine cells will release dopamine to inhibit the activity of the horizontal cells and to improve the contrast of images under dark conditions [28]. As such, we used the Sigmoid function to suppress the output signals of the photoreceptor cells modulated by horizontal cell feedback. The output of the neural signals by the photoreceptor cells is as follows:

$$\mathrm{CS}_{\lambda}(x,y) = \frac{f_{\lambda}(x,y)}{HCF_{\lambda}(x,y)}, co_{\lambda}(x,y) = \frac{1}{1+e^{-10(CS_{\lambda}(x,y)-0.5)}}, \lambda \in \{R,G,B\}. \tag{2}$$

The visual nerve signal processing channels in the retina of marine fish are divided into ON and OFF channels in the bipolar cells layer. The response of the bipolar cells can be simulated by convolving the input signal on the rod-shaped connection channel with the Difference of Gaussian (DOG), and the expression is as follows:

$$\begin{cases} B_P(x,y) = \max[0, (f_{bc} \otimes g_{\sigma_c})(x,y) - k * (f_{bs} \otimes g_{\sigma_s})(x,y)] \\ B_{P'}(x,y) = \max[0, (f'_{bc} \otimes g_{\sigma_c})(x,y) - k * (f'_{bs} \otimes g_{\sigma_s})(x,y)] \\ B_{P_{rod}}(x,y) = \max[0, (f_{bcL} \otimes g_{\sigma_c})(x,y) - k * (f_{bsL} \otimes g_{\sigma_s})(x,y)] \\ g_{\sigma}(x,y) = \frac{1}{2\pi\sigma^2} e^{(-\frac{x^2+y^2}{2\sigma^2})} \end{cases}, \tag{3}$$

where B_P and $B_{P'}$ represent the output signal of the bipolar cells on the ON and OFF channels. $\otimes$ is the convolution operator. f_{bc} and f_{bs} represent the input of the receptive field of the bipolar cells on the ON. f'_{bc} and f'_{bs} represent the input of the receptive field of the bipolar cells on the OFF. k represents the weight of the influence of the receptive field in the outer circle area on the receptive field in the central area. g_{σ} is the function of the two-dimensional Gaussian distribution, which consists of σ, x, and y. $B_{P_{rod}}$ is the output signal of the bipolar cells on the rod-shaped connection channel, and the nerve output signal f_{bcL} is generated by the photoreceptor cells on the rod-shaped connecting channel, and the output signal f_{bsL} is modulated by horizontal cell feedback.

Furthermore, the input to the central area of the receptive field of the bipolar cells on the cone-shaped connection channel $f_{bc}(x,y) = CO_{\lambda}(x,y) * B_{P^{\gamma}_{rod}}$, where γ is used to simulate the non-linear feedback regulation of amacrine cells, and in the experiment, we set $\gamma = 0.5$. Moreover, the output signal of the horizontal cells on the cone-shaped connection channel after local feedback adjustment $f_{bs}(x,y) = \{CO_{\lambda}(x,y)\}mean_{N\times N}$. In Equation (3), when σ_s is set to three times the number of σ_c, the loss of image detail is lower, and setting the parameter σ_c to about 0.3 results in the image-related detail information being more complete after the DOG.

As shown in Figure 1, the receptive field of ganglion cells can be divided into four groups. We define the nerve signals received by the ganglion cells on the ON Channel and OFF Channel as B_P and $B_{P'}$. The signal of the yellow-light channel is obtained by averaging the red-light signal and the green-light signal from the bipolar cells, and the nerve signal of the yellow-light can be expressed as $B_{P_Y} = (B_{P_R} + B_{P_B})/2$, $B_{P'_Y} = \left(B_{P'_R} + B_{P'_B}\right)/2$. After incorporating the color gamut information into the DOG for the calculations, the output signal of the neuron cells can be expressed with the following formula:

$$\begin{cases} G_g(x,y) = max\left[0, \left(f_{gc} \otimes g_{\sigma_c} + m * \left(f_{gc} \otimes g_{\sigma_c} - f_{gs} \otimes g_{\sigma_x}\right)\right)(x,y)\right] \\ G'_g(x,y) = max\left[0, \left(f'_{gc} \otimes g_{\sigma_c} + m * \left(f'_{gc} \otimes g_{\sigma_c} - f'_{gs} \otimes g_{\sigma_x}\right)\right)(x,y)\right] \end{cases}, \tag{4}$$

where G_g and G'_g are the nerve signal output by the ganglion cells on the ON Channel and OFF Channel. f_{gc}, f_{gs}, f'_{gc} and f'_{gs} are the nerve signals received by the ganglion cells on the ON Channel and OFF Channel. σ is the size of the ganglion cell receptive field in the Difference of Gaussian, and in this study, it was set to the same value as the bipolar cells. m represents the weight of the influence of the receptive field in the outer circle area on the receptive field in the central area, and it is used to enhance image color correction and to further enhance the effects.

The values k and m are the keys to the preprocessing algorithm in this paper. According to [29], we bind the adaptive change trends of k and m to the noise intensity of the underwater images (indicators such as image contrast). According to the dark channel prior (DCP) algorithm, the transmittance $t(x, y)$ is expressed as follows:

$$t(x,y) = 1 - \underset{x,y \in \Omega}{min}\left(\underset{\lambda}{min}\,\frac{I_\lambda(x,y)}{A_\lambda(x,y)}\right), \lambda \in \{R, G, B\}, \tag{5}$$

where $I_\lambda(x, y)$ is the output image after denoising. A represents the uniform background light, and Ω represents a square area centered on (x, y). Here, we use $A_\lambda(x, y)$ to represent a non-uniform light source color map defined as $A_\lambda(x, y) = HCF_\lambda(x, y), \lambda \in \{R, G, B\}$. According to Equation (5), we have

$$k = \frac{\sum\left[\underset{x,y \in \Omega}{min}\left(\underset{\lambda}{min}\,\frac{I_\lambda(x,y)}{A_\lambda(x,y)}\right)\right]}{N^2}. \tag{6}$$

In order to further correct the color gamut of the output image, we use the color saturation of the output image according to the bipolar cells algorithm to define m as follows:

$$m = \frac{\sum[3min(B_{PR} + B_{PG} + B_{PB})]}{N^2}, \tag{7}$$

where B_P represents the color contrast in the color channel. Therefore, m is inversely proportional to the color saturation of the output image through the bipolar cell algorithm. If the saturation is low, then the value of m will increase, which will increase the suppression effect to help correct the color gamut information of the output image.

The output signals on the ON and the OFF channels at the ganglion cell level will be normalized to their respective bright and dark channels. To highlight the image contrast information, we use the wavelet weighting method to fuse the normalized bright channel information and dark channel information and to avoid the over-saturation of the contrast in a single channel from affecting the image quality of the final output quality. The output of the algorithm is defined as follows:

$$Output(x,y) = wavelet\{\omega_{\mathrm{ON}}(x,y) * G_g(x,y) + \omega_{\mathrm{OFF}}(x,y) * G'_g(x,y)\}, \tag{8}$$

where ω_{ON} and ω_{OFF} are the weights of the ON Channel and OFF Channel. Because the bright channel section has an enormous weight value, the Sigmoid function is introduced to normalize the weight of the bright channel:

$$F(G_g(x,y)) = \left[1 + e^{-10(G_g(x,y) - 0.5)}\right]^{-1}. \tag{9}$$

Additionally, ω_{ON} and ω_{OFF} are transformed as follows:

$$\begin{cases} \omega_{\mathrm{ON}}(x,y) = \frac{F(G_g(x,y))}{G_g(x,y) + F(G_g(x,y))} \\ \omega_{\mathrm{OFF}}(x,y) = 1 - \omega_{\mathrm{ON}}(x,y) \end{cases}. \tag{10}$$

The results of the image preprocessing module, are shown in Figure 2. A comparison of the histogram comparison shows that the histogram distribution of the original image processed by the multi-scale retinex enhancement algorithm is more uniform, indicating that the enhanced image has a better color gamut value. Moreover, the processed image feature details have a higher degree of discrimination, effectively improving the detection model's feature extraction performance.

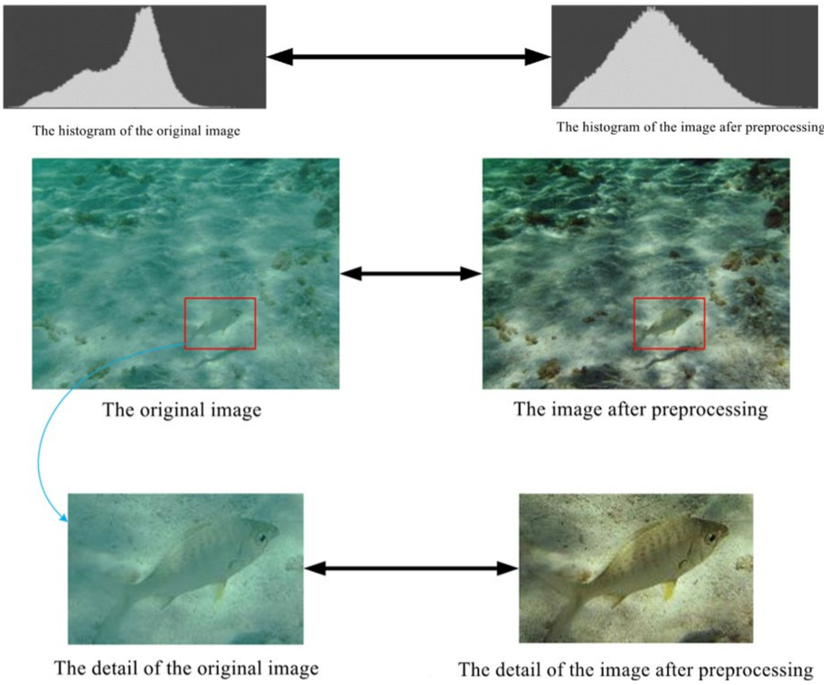
The histogram of the original image
The histogram of the image afer preprocessing
The original image
The image after preprocessing
The detail of the original image
The detail of the image after preprocessing

Figure 2. The original image before and after image preprocessing and histogram comparison. The first row shows that the histogram distribution becomes more uniform, indicating that the enhanced image has a better color gamut value. The last two rows show a higher degree of discrimination, improving the feature extraction performance of the detection model.

2.2. The Multi-Scale Feature-Based Fish Detection Model

Figure 3 illustrates the architecture of the multi-scale feature-based fish detection model based on the Convolution Neutral Network (CNN) model. The detection model consists of a feature extraction module (Section 2.2.1), region proposal module, and region classification module (Section 2.2.2). The processed image is imported into the convolutional neural network for image feature extraction. The region proposal module uses a small neural network with shared parameters to obtain the region proposal information, and the region classification module is operated synchronously with bounding box regression. The model will generate a set of multi-scale feature maps with corresponding distinguishing features during training and will identify and classify images of various species of fish according to the distinguishing features extracted from the set of multi-scale feature maps.

2.2.1. Feature Extraction Module

The information retrieved from the images, such as pixel position and color channel information, can be converted into numerical data using a computer after the feature model extracts the key information. The model, which includes multi-scale feature information, is defined as $M = \{P, X, S\}$. Here, we use $P = \{p_1, \cdots, p_K\}$ to describe a feature map set P containing the feature maps generated from a single image. We use $X = \{x_1, \cdots, x_K\}$ to describe the center coordinate information set of the feature maps and $S = \{s_1, \cdots, s_K\}$ to describe the size information of the feature maps, where K is the number of feature maps generated by a single image. For the image I^m, $P^m = \{p_1^m, \cdots, p_K^m\}$ is the feature map set extracted from $X^m = \{x_1^m, \cdots, x_K^m\}$ and $S^m = \{s_1^m, \cdots, s_K^m\}$ and is normalized according to the size of the feature map. The training of model M can be transformed into a minimization constraint programming problem as follows:

$$(P^*, X^*, S^*) = \text{arg}min\ J(P, X, S) \text{ subject to } \begin{cases} 0 \le s_i^m \le 1, i \in \varepsilon_1, m \in \varepsilon_2 \\ 0 \le x_i^m \pm \frac{1}{2} s_i^m \le 1, i \in \varepsilon_1, m \in \varepsilon_2 \end{cases} \quad (11)$$

where $\varepsilon_1 = \{1, 2, \cdots, K\}$, $\varepsilon_2 = \{1, 2, \cdots, N\}$, and $J(P, X, S)$ are the accumulation of $J_{fitness}$, $J_{separation}$, and $J_{discrimination}$, which are used to describe the objective function of the model. $J_{fitness}$ is used to calculate the similarity of the corresponding feature regions of the same type of feature according to the distance between the feature map P_i and the actual local feature area. As such, $J_{fitness}$ can be defined as follows:

$$J_{fitness} = \sum_{m=1}^{N} \sum_{i=1}^{K} d(P_i, \phi(I(x_i^m, s_i^m))), \tag{12}$$

where $\phi(\cdot)$ is the feature description of the image region, and $d(P, Q)$ is the distance between the feature vector P and the feature vector Q. $J_{separation}$ is used to describe the degree to which the extracted feature maps are concentrated in the disjointed matching areas of the image instead of being concentrated in some of the local areas of the image. As such, $J_{separation}$ can be defined as follows:

$$J_{separation} = \sum_{m=1}^{N} \sum_{i=1}^{K} \sum_{j \neq i}^{N} v\left(I_i^m, I_j^m\right), \tag{13}$$

where I_i^m is the simplified form of $I(x_i^m, s_i^m)$ and I_i^m is the simplified form of $I\left(x_j^m, s_j^m\right)$. $v\left(I_i^m, I_j^m\right)$ is the overlap rate. $v_{i,j}^m$ is the simplified form of $v\left(I_i^m, I_j^m\right)$. $J_{discrimination}$ means that each feature map should represent different local features to capture as many features of the target object as possible as well as reduce the cost of repeated calculations. As such, $J_{discrimination}$ can be defined as follows:

$$J_{discrimination} = -\sum_{i=1}^{K} \sum_{j=1}^{K} d\left(P_i, P_j\right). \tag{14}$$

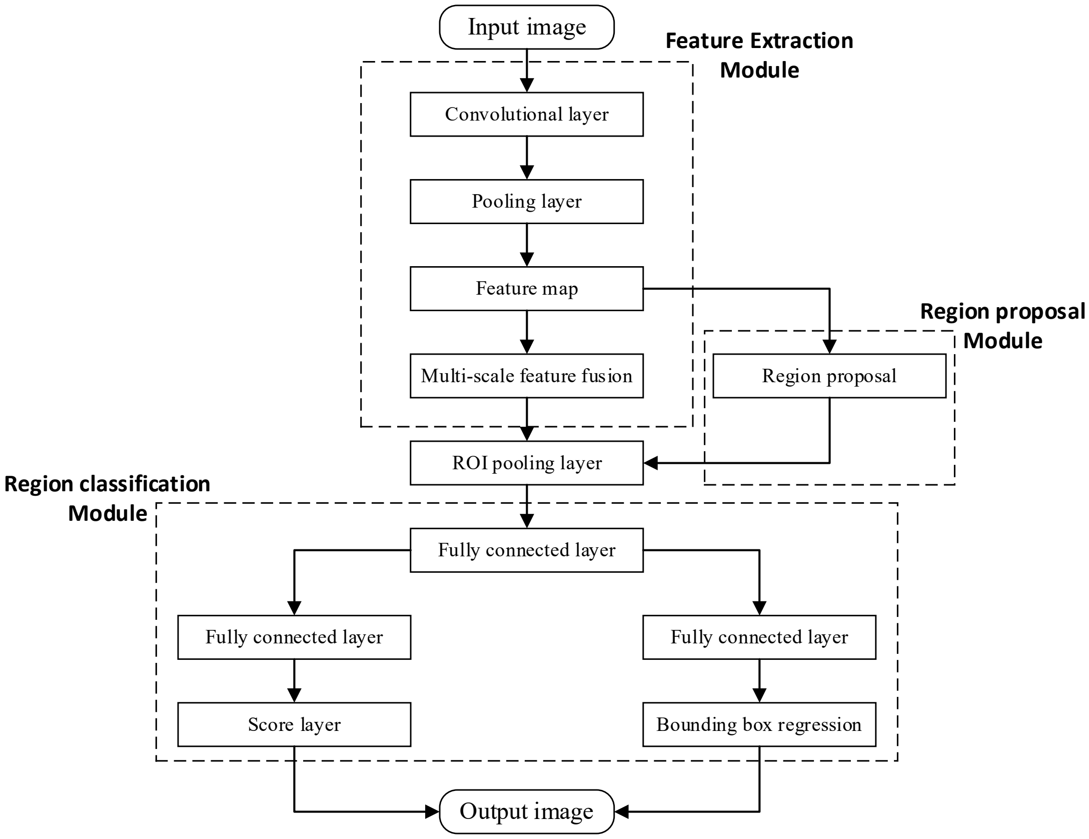

Figure 3. The architecture of the multi-scale fish features detection model.

We used denoising autoencoder and cluster analysis to design the unsupervised learning process. The network structure of the denoising autoencoder includes an input layer, encoder, decoder, and output layer. The structure can be seen in Figure 4.

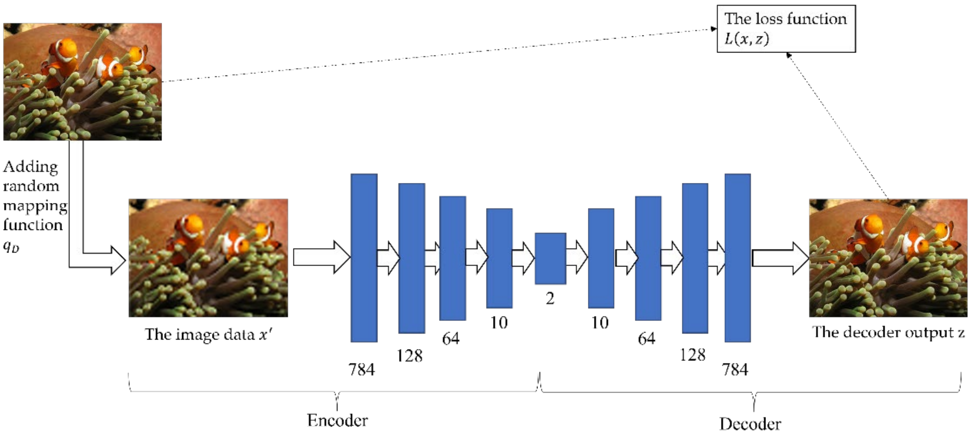

Figure 4. The structure of the denoising autoencoder. The input image data x is corrupted to x'. The autoencoder then maps it to the output data y and attempts to reconstruct x (z).

In the input layer, the input image data is defined as x. The random mapping function q_D is used to perform random destruction operations on the input image data x . The random destruction process is equivalent to adding noise to the input image data x, so we have the image data x'. The encoder output data y is generated after the encoding process of the encoding function f_θ ($y = f_\theta(x') = S(Wx' + b)$). The decoder output data is generated after the decoding process of the decoding function $g_{\theta'}$ (z $= g_{\theta'}(y) = S(W'y + b')$). S is the sigmoid activation function, and W, b or W', b' are the parameters of the encoder or decoder, respectively. The input data x and the decoder output z data jointly define the loss function $L(x,z)$ as follows:

$$L(x,z) = ||x - z||^2 + \lambda \sum_{j=1}^{n} |\theta_j|, \tag{15}$$

where θ is the coding parameter of the encoder after pretraining ($\theta = \{W, b\}$), and λ is the parameter for L^1 regularization. We can tune the parameter of L^1 regularization to make the autoencoder achieve a better fit and generalization. Then, we define the objective function $J = \sum_{i=1}^{n} L(x, g_{\theta'}(f_\theta(x')))/2n$ and use gradient descent to make it close to the optimal value. We compared the input x and output z of the denoising autoencoder to form the corresponding loss function, which is used as the constraint of the autoencoder, thus forming a nonlinear information loss. The internal structure prompts the encoder to continuously extract useful features to make the decoder output a qualified image after filtering out the noise. By applying a random mechanism to select neurons that is to 0 to add environmental noise to the input image, the autoencoder will not experience overfitting, and the extracted features will be more robust.

After the autoencoder extracts the feature set, it is also necessary to use cluster analysis to separate individual feature clusters from the discrete feature set according to the degree of feature similarity. We used the differences and similarities between features as a distance measurement method. The differences between features can update the corresponding model parameters so that they can better distinguish different types of fish. The similarities between the features can update the corresponding parameters by maximizing the differences between the cluster centers to make the same cluster feature move closer to the center, achieving the effect of gathering the feature sub-sets from the same kind of fish. After clustering analysis using differences and similarities between features, the clustering results of the differences between the clusters and the similarities within clusters can be improved simultaneously. The cluster analysis steps are shown in Figure 5.

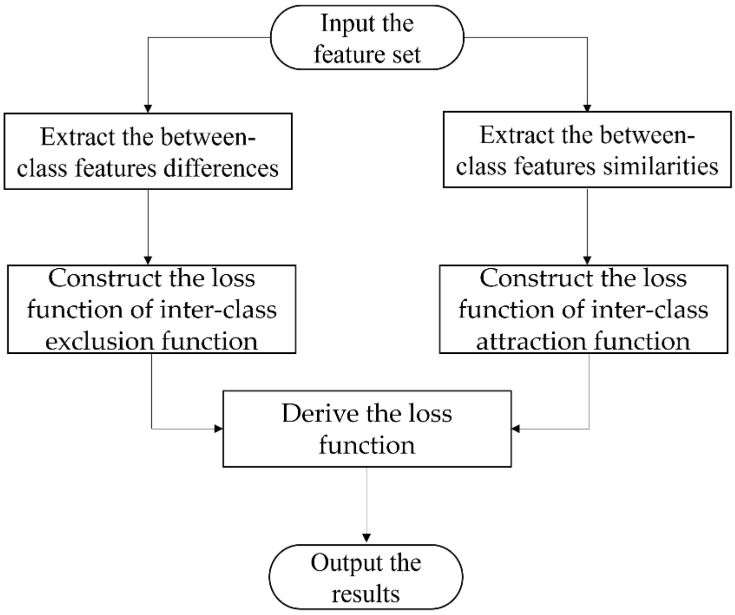
Input the feature set
Extract the between-class features differences
Extract the between-class features similarities
Construct the loss function of inter-class exclusion function
Construct the loss function of inter-class attraction function
Derive the loss function
Output the results

Figure 5. The steps of the cluster analysis. We use the between-class features differences to update the parameters to distinguish the multiple fish specimens and use the between-class features similarities to make the features of the same cluster move closer to the center so as to achieve the effect of clustering the features of the same species of fish.

We define the target in the image as x, so the probability that x belongs to the c cluster center can be expressed as follows:

$$p(c|x_i, V) = \frac{exp\left(V_c^T v_i/\tau\right)}{\sum_{j=1}^{C} exp(V_c^T v_i/\tau)}\ ,\ v_i = \frac{\phi(\theta, x_i)}{||\phi(\theta, x_i)||}, \tag{16}$$

where v_i is the L^2 regularization of x_i, V is the table of each clustering feature, and $V \in R^{C\times n_\phi}$. V_j is the feature of column j, and C is the number of clusters. τ is the scalar factor that controls the peak of the softmax probability distribution. As such, the loss function of inter-class exclusion is defined as $L_r = -log(p(c|x, V))$, and we use the Euclidean distance to define the loss function of inter-class attraction as $L_a = \sum ||v_i - c_{yi}||\ {}^{2}_{2}\ /2$, $c_{yi} \in R^d, i = 1, 2, \cdots, m$.

The loss function is defined as $L = L_r + \lambda L_a$, and λ is the hyperparameter for balancing the sub-loss function. In the training process for cluster analysis, different training samples are habitually discretized and dispersed in the learning space so that different samples are regarded as different clusters. Therefore, we used the structured information in the feature space to define the loss function of inter-class exclusion and the loss function of inter-class attraction and performed clustering ensemble operations on the sample data from the bottom up. During the clustering ensemble process, we used the minimum distance criterion between clusters to calculate the dissimilarity $D_{distance}(A, B)$. We defined the speed of training m as the product of the training parameter γ and the number of initial clusters N.

It is assumed that P and S are constant. We used the autoencoder to update the set X to locate the sub-region of the feature elements. Equation (13) becomes a fixed constant, so Equation (11) can be simplified as follows:

$$\underset{X}{min} \sum_{m=1}^{N} \left(\sum_{i=1}^{K} d(P_i, \phi(I_i^m)) + \sum_{i=1}^{K} \sum_{j\neq i} v_{i,j}^m\right),\ 0 \le x_i \pm \frac{1}{2}s_i \le 1, i = 1, \cdots, K\ . \tag{17}$$

We used the "Mean Shift" algorithm to determine the coordinate variable x. For the image of m, the coordinate is defined as follows:

$$x_i^m(t+1) = \frac{\sum_{j=1}^{n_p} k\left(z_j - x_i^m(t)\right) w_j \left(z_j - x_i^m(t)\right)}{\sum_{j=1}^{n_p} k\left(z_j - x_i^m(t)\right) w_j}, \tag{18}$$

where $k(\cdot)$ is the kernel function, n_p is the pixel value of the feature map, and w_j is the sample weight of z_j. The iteration stops when $\left|\left|x_i^m(t+1) - x_i^m(t)\right|\right|$ reaches the threshold. Then, if we assume the P is constant, and that Equation (14) becomes a fixed constant, Equation (11) can be simplified as follows:

$$\underset{s}{min} \sum_{m=1}^{N} \left(\sum_{i=1}^{K} d(P_i, \phi(I_i^m)) + \sum_{i=1}^{K} \sum_{j=1}^{K} v_{i,j}^m \right), 0 \le s_i \le 1, 0 \le x_i \pm \frac{1}{2} s_i \le 1, i \in \varepsilon. \tag{19}$$

Similarly, we use the "Mean Shift" algorithm to obtain the coordinate variable S. Given the coordinate base $b > 1$, the dimensional variable is defined as follows:

$$s_i^m(t+1) = s_i^m(t) b^{r'},\ r' = \frac{\sum_{r \in \Omega} \sum_{j=1}^{n_p} H(z_j, r) \omega(z_j) r}{\sum_{r \in \Omega} \sum_{j=1}^{n_p} H(z_j, r) \omega(z_j)} \tag{20}$$

where Ω is the search range in the scale space centered on feature map size $\mathrm{s}_i^m(t)$, H is the scale kernel and n_p is the number of pixels. The iteration stops when $||r'||$ reaches the threshold. After knowing X and S, we use the cluster analysis to determine the optimal feature set variable P. We assume $v\left(I_i^m, I_j^m\right)$ is constant, so P_i is as follows:

$$\underset{P_i}{min} \sum_{m=1}^{N} d(P_i, \phi(I_i^m)) - \sum_{j=1}^{K} d(P_i, P_j). \tag{21}$$

When the loss objective function is fitted to a specific threshold parameter in the cluster analysis process, the process ends. Otherwise, the model will repeat the steps and iterative learning. The process of the feature extraction module is illustrated in Figure 6.

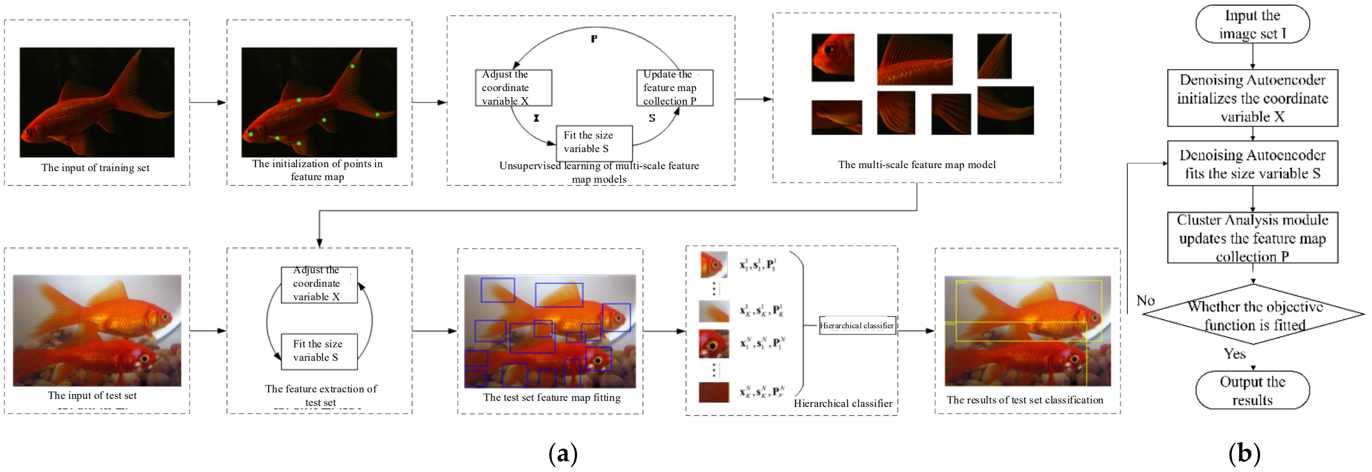
The input of training set
The initialization of points in feature map
P
Adjust the coordinate variable X
Update the feature map collection P
X
S
Fit the size variable S
Unsupervised learning of multi-scale feature map models
The multi-scale feature map model
The input of test set
Adjust the coordinate variable X
Fit the size variable S
The feature extraction of test set
The test set feature map fitting
x_1^1, s_1^1, P_1^1
x_K^1, s_K^1, P_K^1
x_1^N, s_1^N, P_1^N
x_K^N, s_K^N, P_K^N
Hierarchical classifier
Hierarchical classifier
The results of test set classification
(a)
Input the image set I
Denoising Autoencoder initializes the coordinate variable X
Denoising Autoencoder fits the size variable S
Cluster Analysis module updates the feature map collection P
Whether the objective function is fitted
No
Yes
Output the results
(b)

Figure 6. The process of the feature extraction module. The left picture (**a**) is an example using a multi-scale feature-based fish detection algorithm. The right picture (**b**) is the flowchart of the algorithm. The structure of denoising autoencoder is shown in Figure 4, and the steps of cluster analysis are shown in Figure 5.

2.2.2. Region Proposal and Classification Module

The region proposal module comprises a convolutional neural network that divides multiple small areas in the feature map. We can determine the approximate coordinates of the foreground area by comparing the degree of overlap between these small areas and the target area in the learning sample. Then, we pass the foreground area to the ROI pooling layer for region classification and target bounding box regression. The regional classification module consists of an ROI pooling layer, a two-way fully connected layer, a scoring layer, and a bounding box regression layer.

Coral reef fish are often densely packed into the same image frame in submarine coral reef environments. We used this spatial structure layout information to collect multiple confidence scores from similar target objects in multiple directions. The module completes the region proposal and classification operation by sliding a window on the shared feature map. The regional classification module consists of an ROI pooling layer, a two-way fully connected layer, a score layer, and a bounding box regression layer.

In the submarine coral reef environment, there are often many coral reef fish in the same frame of an image. We used this spatial structure layout information to collect confidence scores from similar target objects from multiple directions. The module completes the region proposal and classification operations by sliding a window on the shared feature map. The region classification uses a 1 × 1 sliding window as input to the last convolutional layer to reduce the dimensional information of the region. Then, we input the region feature into two 1 × 1 convolutional layers. One is used for feature localization storage, and the other is used to determine whether the target in the current box belongs to the background or foreground. Moreover, for the particular case of densely packed fish, we introduce spatial regularization weights for each box in the loss function to reduce the loss function of the multi-task objective function below the threshold range. As such, our loss function for an image is defined as follows:

$$\begin{gathered} L(\{u_i\},\{q_i\},\{p_i\}) = \tfrac{1}{N_{fg}}\textstyle\sum_i K(c_i,N_i^*,u_i^*)\cdot L_{fg}(u_i,u_i^*) + \gamma\tfrac{1}{N_{bg}}\textstyle\sum_i L_{bg}(q_i,q_i^*) + \\ \lambda\tfrac{1}{N_{loc}}\textstyle\sum_i L_{loc}(u_i^*,p_i,g_i^*). \end{gathered} \tag{22}$$

Here, $L_{fg}(u_i,u_i^*) = -log[u_iu_i^*]$ and $L_{bg}(q_i,q_i^*) = -log[(1-q_i)(1-q_i^*)]$. The three terms are normalized by N_{fg}, N_{bg}, and N_{log} and are weighted by two balancing parameters, γ, and λ. i is the index of an anchor in a mini-batch, and K is the constant calculated by re-weighting the objective score of each predicted box. c_i is the center coordinate of the predicted box. If an anchor with an Intersection-over-Union (IoU) overlaps with any ground-truth box higher than 0.7, or the IoU overlap is the highest, then $u_i^* = 1$. Otherwise, if the IoU overlap is below 0.3, then $q_i^* = 0$. L_{loc} is valid when $u_i^* = 1$ and the expression is as follows:

$$L_{loc}(u_i^*,p_i,g_i^*) = \textstyle\sum_{i\in fg}\sum_{v\in\{x,y,w,h\}} u_i^* smooth_{L1}(p_i^v, g_i^{v*}), \tag{23}$$

where p_i is the predicted box with data for $x, y, w,$ *and* h. g_i^{v*} can be parameterized as follows:

$$\begin{gathered} g_i^{x*} = \tfrac{g_i^x - d_i^x}{d_i^w}, \qquad g_i^{y*} = \tfrac{g_i^y - d_i^y}{d_i^h} \\ g_i^{w*} = log\left(\tfrac{g_i^w}{d_i^w}\right), \quad g_i^{h*} = log\left(\tfrac{g_i^h}{d_i^h}\right) \end{gathered}. \tag{24}$$

After the region proposal, we used the end-to-end approach for the weighted classification of the region proposal. In the region proposal stage, we adopt K from Equation (23) to calculate the confidence weight of the foreground-predicted box. Additionally, in the region classification stage, we calculated all of the weights of the foreground-predicted box, and the score function is defined as follows:

$$K(c_i,N_i^*,u_i^*) = \begin{cases} \sum_{\theta\in D}\sum_{j\in N_i^*}^{m} G(j,\theta), u_i^* = 1 \\ 1 \qquad\qquad\qquad\quad , u_i^* \neq 1 \end{cases}, G(j,\theta) = \alpha\cdot e^{-(\frac{x_j^\theta}{2\sigma_x^2}+\frac{y_j^\theta}{2\sigma_y^2})}. \tag{25}$$

where α is the amplitude of the Gaussian function and $G(j,\theta)$ is the Gaussian kernel with a different rotation radius $D = \{\theta_1,\cdots,\theta_r\}$.

3. Results

3.1. Experiment Setup

In order to test the performance of the multi-scale fish features detection model, this paper uses the LifeCLEF underwater fish target image data set in ImageCLEF [30], which is shown in Figure 7 below. The LifeCLEF underwater fish target image data set is taken from the Fish4Knowledge underwater video set. Considering the large gap in the number of fish species images, we selected 13 fish sub-datasets with a large number of images (320 × 240 or 640 × 480) as the dataset in this paper. Moreover, the dataset was divided into three parts proportionally, and the detailed information is shown in Table 1. Considering the non-balanced datasets, we used random rotation, random horizontal flip, random vertical flip, random cropping, and other methods to conduct the data augmentation. After data augmentation, the number of samples in the train set is three times the number of samples in the initial train set.

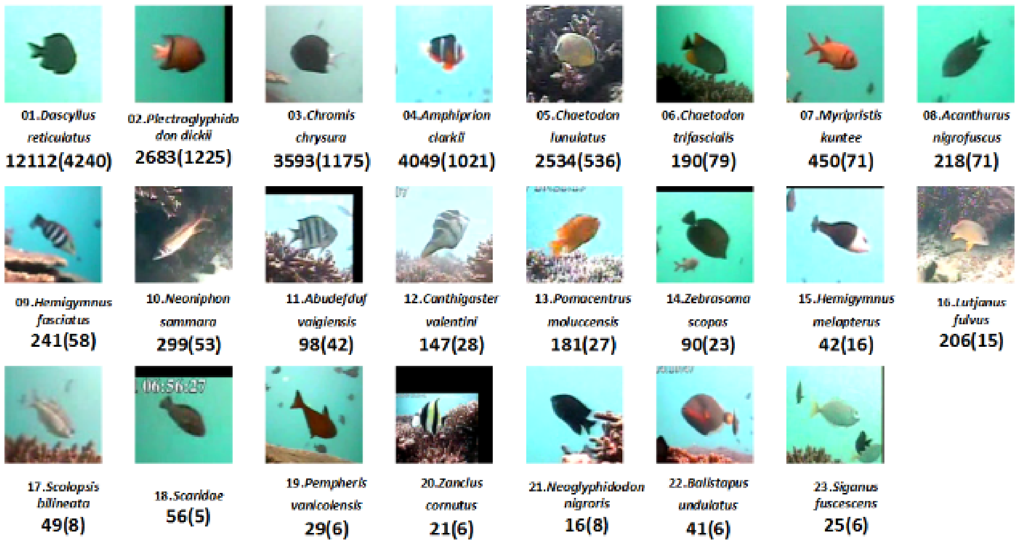
01.Dascyllus reticulatus 12112(4240)
02.Plectroglyphidodon dickii 2683(1225)
03.Chromis chrysura 3593(1175)
04.Amphiprion clarkii 4049(1021)
05.Chaetodon lunulatus 2534(536)
06.Chaetodon trifascialis 190(79)
07.Myripristis kuntee 450(71)
08.Acanthurus nigrofuscus 218(71)
09.Hemigymnus fasciatus 241(58)
10.Neoniphon sammara 299(53)
11.Abudefduf vaigiensis 98(42)
12.Canthigaster valentini 147(28)
13.Pomacentrus moluccensis 181(27)
14.Zebrasoma scopas 90(23)
15.Hemigymnus melapterus 42(16)
16.Lutjanus fulvus 206(15)
17.Scolopsis bilineata 49(8)
18.Scaridae 56(5)
19.Pempheris vanicolensis 29(6)
20.Zanclus cornutus 21(6)
21.Neoglyphidodon nigroris 16(8)
22.Balistapus undulatus 41(6)
23.Siganus fuscescens 25(6)

Figure 7. Underwater fish target image data set in ImageCLEF [30].

The Precision, Recall, Miss Rate (MR), and False Positive Per Image (FPPI) rates are often used as standard evaluation parameters in target detection tasks. In order to respond to the detection performance of the model more intuitively, we used PR Curve, in which the abscissa is Recall and the ordinate is Precision. The MR-FPPI Curve, in which the abscissa is FPPI and the ordinate is MR, is also used to evaluate the detection model.

Experiments were carried out using an Nvidia 1080Ti GPU with a graphics memory of 8 G. Considering that the use of the Fish4Knowledge underwater video data set may result in an over-fitting phenomenon, during the experiment, random rotation, random horizontal flip, random vertical flip, and other methods were used to augment the image data. There are three times more image data than the initial sample size.

During the training process of the model, there was a total of 140,000 training times. The learning rate for 0–79,999 pieces of training data in the initial stage was set to 0.001, the learning rate for 80,000–109,999 pieces of training in the intermediate stage was set to 0.0001, and the learning rate for 110,000–139,999 pieces of training in the final stage was set to 0.00001. We set the Batch Size to 4 and the Iter Size to 4. During the test, the Batch Size was set to 1, and the Iter Size was set to 400. Finally, the IoU threshold was set to 0.5, the confidence threshold was set to 0.1, and the non-maximum suppression threshold was set to 0.45.

Table 1. The species used in the dataset.

Fish Species	Training Set	Validation Set	Test Set
dascyllus reticulatus	4032	4042	4037
plectroglyphido don dickii	894	890	898
chromis chrysura	1192	1202	1197
amphiprion clarkii	1349	1355	1344
chaetodon lunulatus	844	839	849
chaetodon trifascialis	63	68	58
myripristis kuntee	145	155	150
acanthurus nigrofuscus	78	68	73
hemigymnus fasciatus	85	75	80
neoniphon sammara	94	104	99
canthigaster valentini	44	54	49
pomacentrus moluccensis	55	65	60
lutjanus fulvus	63	73	68
total number of sample	8938	8990	8962

At last, we applied our model to the underwater equipment, which mainly consisted of a Raspberry Pi, a CCD camera, and an underwater pressure chamber. Then, we conducted the field experiment, as shown in Figure 8.

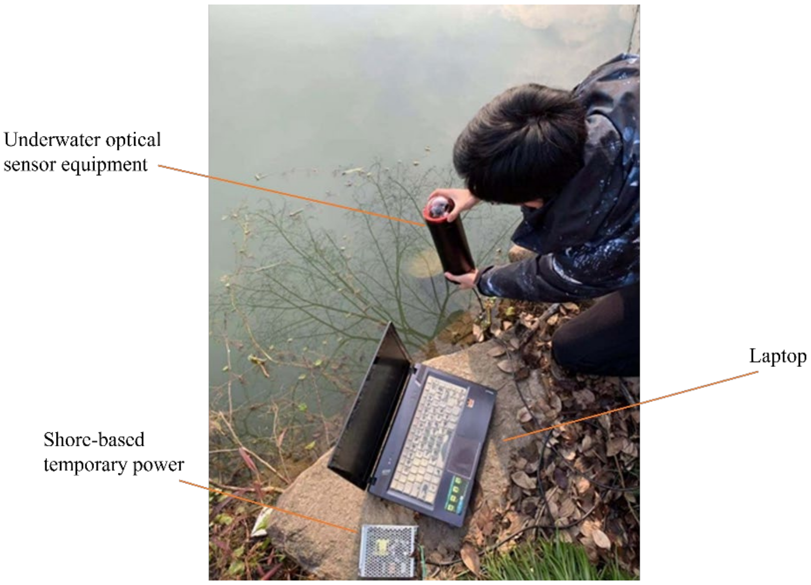
Underwater optical sensor equipment
Laptop
Shore-based temporary power

Figure 8. The field experiment for the underwater equipment applying our model. The underwater optical sensor equipment consists of a Sony IMX322 optical sensor, a raspberry pie with ARM Cortex-A53, and a high hydrostatic pressure chamber.

3.2. Result and Discussion

We put the model algorithm designed in this article and five other current mainstream target detection model algorithms together in the same experimental environment to conduct the experimental analysis. The six mainstream target detection model algorithms used as experimental controls included R-CNN and Fast R-CNN, Faster R-CNN, YOLO, SSD, and the RetinaNet algorithm.

The MR-FPPI Curve and the PR Curve of the model algorithm designed in this paper and those of the R-CNN, Fast R-CNN, Faster R-CNN, YOLO, SSD, and RetinaNet algorithms in the selected Fish4Knowledge fish sub-data set with the IoU threshold parameter with the expected value of 0.5 are illustrated in Figures 9 and 10. It can be seen from Figure 9 that the experimental analysis of the model algorithm designed in this paper has an MR-FPPI curve that is below those of the other algorithms, proving that the detection performance (Recall) of the model algorithm designed in this paper is better than that of the other mainstream models in the underwater environment. In Figure 10, the PR curve of the model algorithm designed in this paper is to the upper right of the other algorithms, which proves that the detection performance (Precision) of the model algorithm designed in this paper is better than other mainstream models in underwater environments. Moreover, we used the MR-FPPI curve and the PR Curve obtained via the experimental analysis of each algorithm in the Fish4Knowledge fish sub-data set to calculate the corresponding mean Miss Rate (mMR) and mean Average Precision(mAP), which are shown in Tables 2 and 3. The test results of our model are illustrated in Figure 11.

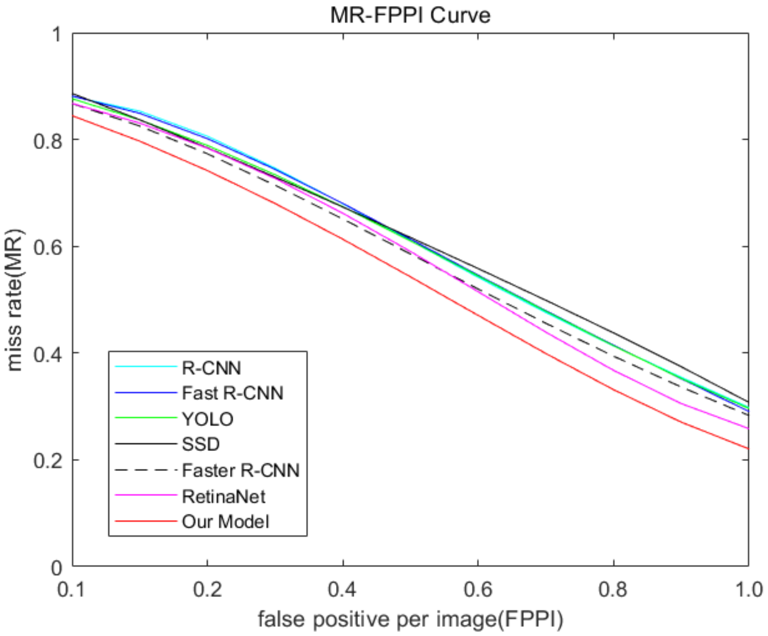

Figure 9. The MR-FPPI curves of the seven tested methods.

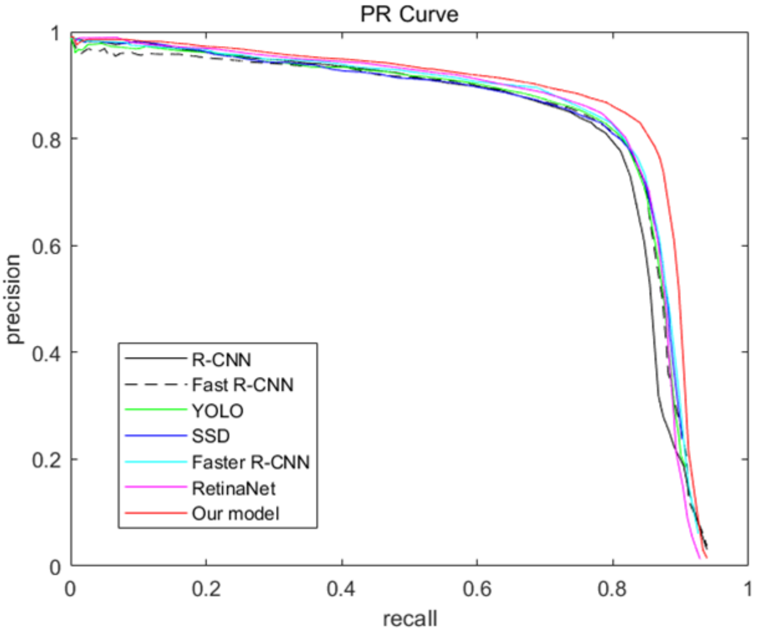

Figure 10. The PR curves of the seven tested methods.

Table 2. Detection performance (Recall) of the seven tested methods using the LifeCLEF test set.

Method	mMR	Improvement
R-CNN	63.42	/
Fast R-CNN	63.30	0.12
YOLO	63.20	0.22
SSD	62.76	0.66
Faster R-CNN	60.82	2.60
RetinaNet	59.44	3.96
Our Model	54.11	9.31

Table 3. Detection performance (Precision) of the seven tested methods using the LifeCLEF test set.

Method	mAP	Improvement
R-CNN	70.29	/
Fast R-CNN	71.56	1.27
YOLO	71.81	1.52
SSD	72.24	1.95
Faster R-CNN	72.97	2.68
RetinaNet	73.03	2.74
Our Model	78.31	8.02

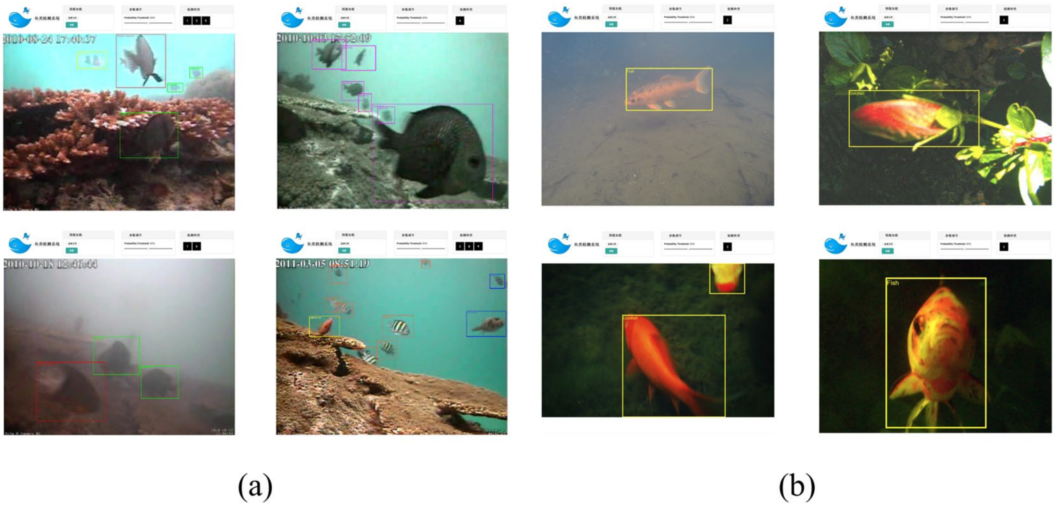

Figure 11. (**a**) Test results on the Fish4Knowledge [30] dataset; (**b**) test results on the data from the field experiment.

4. Conclusions

This paper investigated underwater target detection and tracking technology using a fish features detection system with an image preprocessing module and a multi-scale fish feature detection module. The experiment and comparison with other methods prove that our model performs better and meets real-time requirements.

In future work, we will investigate preprocessing algorithms that are better able to filter underwater noise and that implements an improved and more reasonable coupling method between the preprocessing algorithm and detection algorithm to further promote the performance of the detection system.

Author Contributions: Conceptualization, Y.L. and Y.C.; methodology, Y.L. and Y.C.; software, Y.C.; validation, Y.C. and L.Z.; formal analysis, Y.C.; investigation, Y.L.; data curation, Y.L.; visualization, L.Z.; writing—original draft preparation, Y.L.; writing—review and editing, Y.L., L.Z. and Y.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by National Natural Science Foundation of China (No.51979246), the National Key Research and Development Program of China (No. 2021YFC2800202) and Scientific Research Fund of Zhejiang Provincial Education Department (No. Y202148362).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The LifeCLEF underwater fish target image data can be found at https://homepages.inf.ed.ac.uk/rbf/Fish4Knowledge/ (accessed on 4 August 2021). The data from the field experiment presented in this paper are available upon request from corresponding author.

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

References

1. Li, P. Research progress in high-value utilization of marine biological resources. *Oceanol. Limnol. Sin.* **2020**, *51*, 750–758. [CrossRef]
2. Cong, Y.; Gu, C.; Zhang, T.; Gao, Y. Underwater robot sensing technology: A survey. *Fundam. Res.* **2021**, *1*, 337–345. [CrossRef]
3. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587. [CrossRef]
4. Uijlings, J.R.R.; Van de Sande, K.E.A.; Gevers, T.; Smeulders, A.W.M. Selective Search for Object Recognition. *Int. J. Comput. Vis.* **2013**, *104*, 154–171. [CrossRef]
5. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef] [PubMed]
6. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448. [CrossRef]
7. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
8. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [CrossRef]
9. Lin, C.-H.; Lin, Y.-S.; Liu, W.-C. An efficient license plate recognition system using convolution neural networks. In Proceedings of the 2018 IEEE International Conference on Applied System Invention (ICASI), Chiba, Japan, 13–17 April 2018; pp. 224–227. [CrossRef]
10. Zhang, M.; Qiao, B.; Xin, M.; Zhang, B. Phase spectrum based automatic ship detection in synthetic aperture radar images. *J. Ocean Eng. Sci.* **2020**, *6*, 185–195. [CrossRef]
11. Al-Aboosi, Y.Y.; Sha'Ameri, A.Z. Improved signal de-noising in underwater acoustic noise using S-transform: A performance evaluation and comparison with the wavelet transform. *J. Ocean Eng. Sci.* **2017**, *2*, 172–185. [CrossRef]
12. Wang, H.; Zhang, S.; Zhao, S.; Wang, Q.; Li, D.; Zhao, R. Real-time detection and tracking of fish abnormal behavior based on improved YOLOV5 and SiamRPN++. *Comput. Electron. Agric.* **2021**, *192*, 106512. [CrossRef]
13. Li, X.; Shang, M.; Qin, H.; Chen, L. Fast accurate fish detection and recognition of underwater images with Fast R-CNN. In Proceedings of the OCEANS 2015—MTS/IEEE Washington, Washington, DC, USA, 19–22 October 2015; pp. 1–5. [CrossRef]
14. Cai, K.; Miao, X.; Wang, W.; Pang, H.; Liu, Y.; Song, J. A modified YOLOv3 model for fish detection based on MobileNetv1 as backbone. *Aquac. Eng.* **2020**, *91*, 102117. [CrossRef]
15. Kottursamy, K. Multi-scale CNN Approach for Accurate Detection of Underwater Static Fish Image. *J. Artif. Intell. Capsul. Netw.* **2021**, *3*, 230–242. [CrossRef]
16. Sung, M.; Yu, S.-C.; Girdhar, Y. Vision based real-time fish detection using convolutional neural network. In Proceedings of the OCEANS 2017-Aberdeen, Aberdeen, UK, 19–22 June 2017. [CrossRef]
17. Levy, D.; Belfer, Y.; Osherov, E.; Bigal, E.; Scheinin, A.P.; Nativ, H.; Tchernov, D.; Treibitz, T. Automated Analysis of Marine Video with Limited Data. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, USA, 18–22 June 2018; pp. 1466–14668. [CrossRef]
18. Knausgård, K.M.; Wiklund, A.; Sørdalen, T.K.; Halvorsen, K.T.; Kleiven, A.R.; Jiao, L.; Goodwin, M. Temperate fish detection and classification: A deep learning based approach. *Appl. Intell.* **2021**, *52*, 6988–7001. [CrossRef]
19. Ben Tamou, A.; Benzinou, A.; Nasreddine, K. Multi-stream fish detection in unconstrained underwater videos by the fusion of two convolutional neural network detectors. *Appl. Intell.* **2021**, *51*, 5809–5821. [CrossRef]
20. Zhang, S.; Yang, X.; Wang, Y.; Zhao, Z.; Liu, J.; Liu, Y.; Sun, C.; Zhou, C. Automatic Fish Population Counting by Machine Vision and a Hybrid Deep Neural Network Model. *Animals* **2020**, *10*, 364. [CrossRef] [PubMed]
21. Zheng, H.; Sun, X.; Zheng, B.; Nian, R.; Wang, Y. Underwater image segmentation via dark channel prior and multiscale hierarchical decomposition. In Proceedings of the OCEANS 2015-Genova, Genova, Italy, 18–21 May 2015; pp. 1–4. [CrossRef]
22. Marburg, A.; Bigham, K. Deep learning for benthic fauna identification. In Proceedings of the OCEANS 2016 MTS/IEEE Monterey, Monterey, CA, USA, 19–23 September 2016; pp. 1–5. [CrossRef]

23. Priyankan, K.; Fernando, T.G.I. Mobile Application to Identify Fish Species Using YOLO and Convolutional Neural Networks. In *Proceedings of International Conference on Sustainable Expert Systems*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 303–317. [CrossRef]
24. Song, M.; Qu, H.; Zhang, G.; Tao, S.; Jin, G. A Variational Model for Sea Image Enhancement. *Remote Sens.* **2018**, *10*, 1313. [CrossRef]
25. Chen, W.; Wang, L.; Zhang, Y.; Li, X.; Liu, J.; Wang, W. Anti-disturbance grabbing of underwater robot based on retinex image enhancement. In Proceedings of the 2019 Chinese Automation Congress (CAC), Hangzhou, China, 22–24 November 2019; pp. 2157–2162. [CrossRef]
26. Douglas, R.H.; Hawryshyn, C.W. Behavioural studies of fish vision: An analysis of visual capabilities. In *The Visual System of Fish*; Douglas, R., Djamgoz, M., Eds.; Springer: Dordrecht, The Netherlands, 1990; pp. 373–418. [CrossRef]
27. Brown, C. Fish intelligence, sentience and ethics. *Anim. Cogn.* **2014**, *18*, 1–17. [CrossRef] [PubMed]
28. Mangel, S.C.; Dowling, J.E. The interplexiform–horizontal cell system of the fish retina: Effects of dopamine, light stimulation and time in the dark. *Proc. R. Soc. Lond. Ser. B Biol. Sci.* **1987**, *231*, 91–121. [CrossRef]
29. Angelucci, A.; Bijanzadeh, M.; Nurminen, L.; Federer, F.; Merlin, S.; Bressloff, P.C. Circuits and Mechanisms for Surround Modulation in Visual Cortex. *Annu. Rev. Neurosci.* **2017**, *40*, 425–451. [CrossRef]
30. Fisher, R.; Chen-Burger, Y.; Giordano, D.; Hardman, L.; Lin, F. *Fish4Knowledge: Collecting and Analyzing Massive Coral Reef Fish Video Data*; Springer: Heidelberg, Germany, 2016.