symmetry

MDPI

Article

An Embedding Skeleton for Fish Detection and Marine Organisms Recognition

Jinde Zhu [1], Wenwu He [1], Weidong Weng [1], Tao Zhang [2], Yuze Mao [3], Xiutang Yuan [4], Peizhen Ma [5] and Guojun Mao [1,*]

[1] School of Computer Science and Mathematics, Fujian University of Technology, Fuzhou 350011, China; 61201916@fjut.edu.cn (J.Z.); hwwhbb@163.com (W.H.); weidong_1996@126.com (W.W.)
[2] CAS Key Laboratory of Marine Ecology and Environmental Sciences, Institute of Oceanology, Chinese Academy of Sciences, Qingdao 266071, China; tzhang@qdio.ac.cn
[3] Yellow Sea Fisheries Research Institute, Chinese Academy of Fishery Sciences, Qingdao 266071, China; maoyuze@163.com
[4] Yantai Institute of Coastal Zone Research, Chinese Academy of Sciences, Yantai 264003, China; xtyuan@yic.ac.cn
[5] Department of Marine Organism Taxonomy and Phylogeny, Institute of Oceanology, Chinese Academy of Sciences, Qingdao 266071, China; mapeizhen@qdio.ac.cn
* Correspondence: 19662092@fjut.edu.cn

Abstract: The marine economy has become a new growth point of the national economy, and many countries have started to implement the marine ranch project and made the project a new strategic industry to support vigorously. In fact, with the continuous improvement of people's living standards, the market demand for precious seafood such as fish, sea cucumbers, and sea urchins increases. Shallow sea aquaculture has extensively promoted the vigorous development of marine fisheries. However, traditional diving monitoring and fishing are not only time consuming but also labor intensive; moreover, the personal injury is significant and the risk factor is high. In recent years, underwater robots' development has matured and has been applied in other technologies. Marine aquaculture energy and chemical construction is a new opportunity for growth. The detection of marine organisms is an essential part of the intelligent strategy in marine ranch, which requires an underwater robot to detect the marine organism quickly and accurately in the complex ocean environment. This paper proposes a method called YOLOv4-embedding, based on one-stage deep learning arithmetic to detect marine organisms, construct a real-time target detection system for marine organisms, extract the in-depth features, and improve the backbone's architecture and the neck connection. Compared with other object detection arithmetics, the YOLOv4-embedding object detection arithmetic was better at detection accuracy—with higher detection confidence and higher detection ratio than other one-stage object detection arithmetics, such as EfficientDet-D3. The results show that the suggested method could quickly detect different varieties in marine organisms. Furthermore, compared to the original YOLOv4, the mAP75 of the proposed YOLOv4-embedding improves 2.92% for the marine organism dataset at a real-time speed of 51 FPS on an RTX 3090.

Keywords: deep learning; computer vision; multi-class classification; fish detection; YOLOv4

check for updates

Citation: Zhu, J.; He, W.; Weng, W.; Zhang, T.; Mao, Y.; Yuan, X.; Ma, P.; Mao, G. An Embedding Skeleton for Fish Detection and Marine Organisms Recognition. *Symmetry* **2022**, *14*, 1082. https://doi.org/10.3390/sym14061082

Academic Editors: Chien-Hsing Chou and Yi-Zeng Hsieh

Received: 7 March 2022
Accepted: 17 May 2022
Published: 24 May 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With fish fields' evolution in artificial intelligence, the mechanic has attracted wide advertency in aquaculture [1]. The underwater robot is one of the most popular ones [2]. The underwater mechanic has been significantly high-ranking recently in speed and correctness, and aquaculture mechanics are utilized to harvest fish and marine organisms.

Computer vision (CV) utilize for fish weight estimation and recognition. The CV economy is the crux to realizing automatic crop, and precise detection presupposes follow-up actions. However, there is still a significant challenge in implement efficiently object detection in marine organisms due to the similarity of occlusion. Furthermore, the ocean background makes this task more challenging to premeditate the complication and the uncertainty of the ocean's situations.

Traditionally, masterpiece machine learning modes with multi-spatial input features such as Random Forest [3], Adaboost [4], or SVM [5], to achieve object detection that has used to conclude knowledge model mining. However, before the detection step, Haar [6], SIFT [7], or HOG [8] adopted image feature extraction operations to establish the input features. Therefore, object detection and feature extraction are entirely independent.

Since prescriptive image characteristic extractive operators are separated from the detection step and based on subjective judgments, weak generalization ability and task-dependent features in object detection are challenging.

Scholars cross to deep learning to incorporate object detection and feature extraction into one task and accomplish end-to-end learning [9] in the Convolutional Neural Network (CNN) [10]. Most of the present deep learning resources better the learning diathesis by increasing the depth of CNN, which accomplishes considerable success in works for images and texts. Furthermore, CNN introduces pooling operations and convolution to promote automatic features, unlike custumal neural networks. Generally, CV-related types plot CNN algorithms according to the adoption purpose: detection networks [11] and classification networks [12].

LeNet [11] is one of the incipient CNNs for classification. Alex et al. [9] intensified the network architecture and succeeded in ImageNet classification. Karen proposed VGGNet and successfully increased the depth of CNN into 16–19 layers, improving the classification capability. Finally, He et al. [13] trained a 152-layer deep neural network with Residual Units, and the number of parameters was lower than VGGNet, but the performance is better.

Meanwhile, Google also proposed a series of convolutional networks, i.e., Inception v1–v4 [14–17], where Inception modules have replaced the model, with depthwise separable convolutions [18], the model without increasing the network complicacy for better structure and performance.

Liu et al. [19] initiated DenseNet to decrease the number of parameters substantially for weight sharing. Khan et al. deliver Channel Boosting in a deep CNN for new pattern enhancement . Both Liu and Khan used the itinerary dimension of Transfer learning and CNN. Misra assumed activation effect MISH, which conducts better than ReLU on routine datasets in the most recommended deep networks. Channel Boosting CNN also appraised the medical image dataset (Aziz et al. [20]), showing better results.

Two types can further separate the detection networks, one is the basis of the candidate areas called two-stage detector, and the other is the basis of the regression manner means one-stage detector) [21]. R-CNN catenas are representative networks based on candidate areas. R-CNN was initiated by Girshick et al. [12]. They were extracting characteristics using CNN and SVM for classification. For better object detection behavior, Girshick [22] presented Fast R-CNN . In the same year, Ren et al. [23] suggested Faster R-CNN facilitate the expression of Fast R-CNN. Lin et al. [24]used FPN to heighten Faster R-CNN's capability further. He et al. aroused Mask R-CNN and increased work to Faster R-CNN. For an analogy, Mask R-CNN can see as a correct object detector at segmentation allocation.

As for regression-based delegate networks, SSD (Single Shot MultiBox Detector) and YOLO (You only look once) series are typical. Liu et al. [21] proposed SSD, where SSD's mean average precision (mAP) is better than that of Faster R-CNN. The first version of YOLO was proposed by Redmon et al. [25], which divided the image into an SxS grid and formulated object detection as a regression task in a one-shot, while YOLOv1 realized the idea of the target detection with greatly improved detection speed.

Redmon [26] allocated the anchor case and conveyed YOLOv2 with ameliorated multi-dimension objective detection. Lin et al. [27] initiated RetinaNet for the maladjustment of negative and positive specimens in the dataset. Finally, YOLOv3 with higher tiny object average precision ability on the small size objects distributed, reforming the difficulty of the former two versions and having a higher disclosure speed and correctness. Bochkovskiy et al. presented the newest release, YOLOv4, as more formidable than elapsed versions in FPS(Frames Per Second)and AP(Average Precision), a backbone network, neck network, and activation function reform with majorization.

This paper aims to grow a recognition economy for AUVs (Autonomous Underwater Vehicles) to identify marine organisms and the environment. We mainly use a regular RGB camera to get photos of marine organisms in the farming pool. Object invention assignments are carried out under different circumstances of occlusion and illumination. The primary dedication of this work involves:

1. Rapid and precise detection by the proposed YOLOv4-embedding structure for marine organisms under various environmental conditions.
2. We have compared Efficientdet [28] and discussed the fish detection results to verify the applicability and effectiveness, and recommended a method in marine organisms detection.

Section 1 is the introduction. Section 2 details related work on fish and marine organisms detection—the rest of the paper is organized as follows. In Section 3, we define the data and methods. We also briefly review the properties of the YOLOv4-embedding arithmetic. Different object detection arithmetic techniques are compared in Section 4. Finally, the experimental analysis is discussed and concluded in Sections 5 and 6, respectively.

2. Associated Work on Fish and Marine Organisms Discovery

Deep learning-based CV techniques and object detection algorithms have been widely exploited in aquaculture, such as fish size measuring, body analysis, quality calculation, illness diagnosis, etc. As a contactless method, high-precision CV techniques can monitor the farmed organisms' size, fabric, and physical condition, and become a vital monitoring method in aquaculture [29]. As mentioned before, CNN has been widely used in CV and mainly made a breakthrough in abstract cognitive problems [30]. For example, based on the Fish4 Knowledge dataset, Rathi [31] designed a three-level CNN to classify 21 types of tropical fish.

By combining the feature selection framework and image segmentation, Marini [32] assessed the affluence of fish and carried out classification for fish classes on the collected data. Mandal [33] combined Faster R-CNN with three classification networks (ZFNeT, CNN-M, and VGG16) to the realized regional prediction for fish and crustaceans collected from Queensland beaches. Konovalov [34] designed an Xception CNN-based fish detector for a fish group and realized underwater fish detection in multiple water areas. The study [35] based on YOLOv3 shows that it improves the detection performance for marine organisms by using target objects' color and texture features.

However, the ascertainment of marine organisms accustomed to CPU train pictures in early work and the training gallop was sluggish. Moreover, marine organisms may also lower the detection rate if the sparse backdrop noise is not removed. Meanwhile, GPU processing adopted in images database makes training faster and more valid.

This paper conveys the novel arithmetic YOLOv4-embedding for marine organisms to implement a fast breakthrough. As shown from the experiment result, distinct marine organisms can be classified and identified by using improved CSPNet [36] construction into the neck architecture of YOLOv4-embedding. In addition, optimizing the gradient backpropagation orbit ameliorates the network's learning capacity. Compared with the mastercopy YOLOv4 and other algorithms, YOLOv4-embedding is rapid and better precise. Furthermore, YOLOv4-embedding can neutralize fast velocity and high accuracy for marine organisms recognition tasks, which is incredibly generous for underwater mechanics to conduct fishing jobs. Another benefit of our effort is that using a standard RGB camera instead of the transducer reduces the cost of gathering images of marine organisms under shallow sea scenes.

3. Architecture Design of YOLOv4-Embedding

3.1. Data and Relevant Methods

To validate the suggested modus, we use the data supported by the IOCAS (Institute of Oceanology, Chinese Academy of Sciences), extracted in the natural aquaculture surrounding. Specifically, 1557 compelling marine organism images embrace four organisms (Figure 1), i.e., Abalone, Echinoidea, Holothuroidea, and Thamnaconus modestus. Data used a digital color camera called GoPro CHDHX with a resolution of 1280 × 720 pixels, and the Farming pond set the shooting view to the facade and flank. Furthermore, the contrast adopted some photos with an angle of 45. The training, validation, and test volumes include 1307, 100, and 150 images, respectively, and the mosaic photo enhancement means used to promote the image set.

Labelimg is an open-source and free labeling instrument to mark each photo. Once the marine organisms are labeled, an XML (Extensible Markup Language) file is generated, including the coordinates of boxes and the labels that boundary the target marine organisms in the photo.

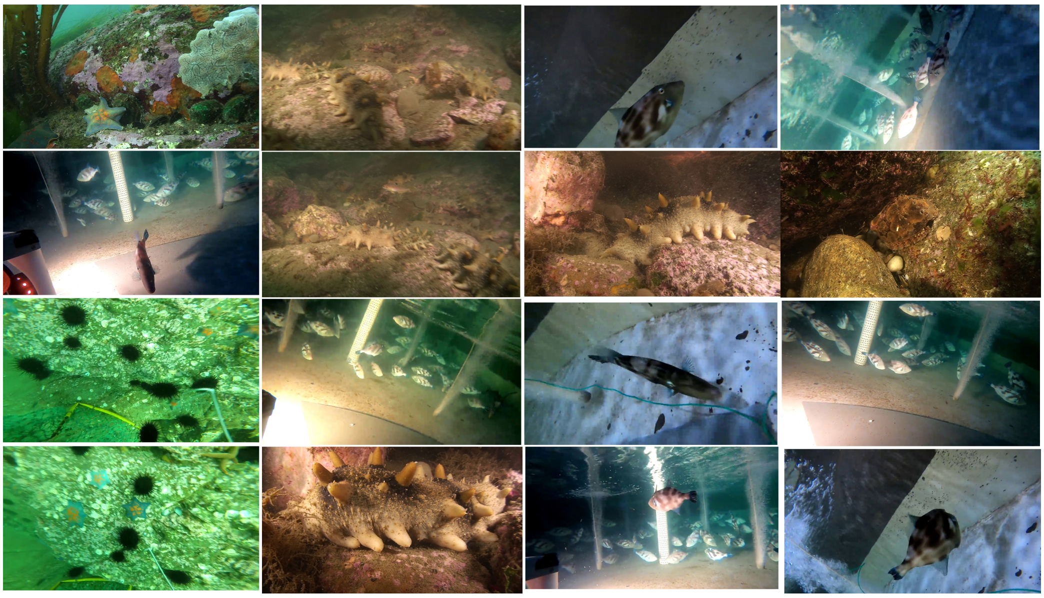


Figure 1. Marine organism images.

3.2. Detection Procedure

This subsection expresses the interior structure of YOLOv4-embedding. YOLOv4-embedding includes four main elements: the backbone network, neck network, input layer, and output layer. The produce layer with three different level prediction anchor boxes called YOLOv4 Head. The input layer adopts fixed-size photos collected through the backbone network and sent to the neck network for feature mixture. Figure 2 is the architecture of marine organisms disclosure based on the YOLOv4-embedding arithmetic. The discovery procedure is summarized as follows:

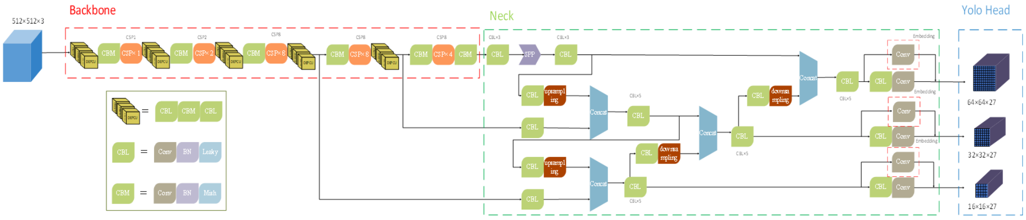


Figure 2. The overall network architecture of YOLOv4-embedding.

- Step 1: Feed Marine organisms photo into the network.
- Step 2: The CSPDarknet53 backbone maintains the Darknet53 skeleton and uses the CSP organization. The Leaky and Mish activation functions extract the image's info.
- Step 3: Assemble SPP(Spatial Pyramid Pooling) [37] module and FPN (Feature Pyramid Networks) + PAN (Path Aggregation Network) pattern to the feature collected by the backbone. PAN uses path aggregation and characteristic pyramid technique to make the propagation of low-level messages to the top-level easier [38]. The multi-scale forecasting for three styles of goals: small ones, medium ones, and large ones.
- Step 4: Embedding linear activation function and the convolution layer at the end of the YOLOv4-embedding Neck. Conv + Batch normalization + Liner(CBLR) engaged in the network. The structure shows in Figure 3 Concat is the addition of dimensionality and tensors, which add the characteristic of the two CBLR. After Conv processed and Batch Normalization, the data obtain new values, then put into the linear activation function after regulation.
- Step 5: The YOLOv4-embedding head executes predicting, which produces the final disclosure consequence. Here is the expounding of the concrete building blocks. The backbone CSPDarknet53 structure that incorporates 5 CSP (Cross Stage Partial connections) model, 11 Convolutional + Batch normalization + Mish(CBM) model and 10 Convolutional + Batch normalization + Leaky(CBL). The CBM model implements the convolution task using Mish activation functions and Batch Normalization. In contrast, the CBL module performs the convolution mission using Batch Normalization and Leaky Relu activation functions.

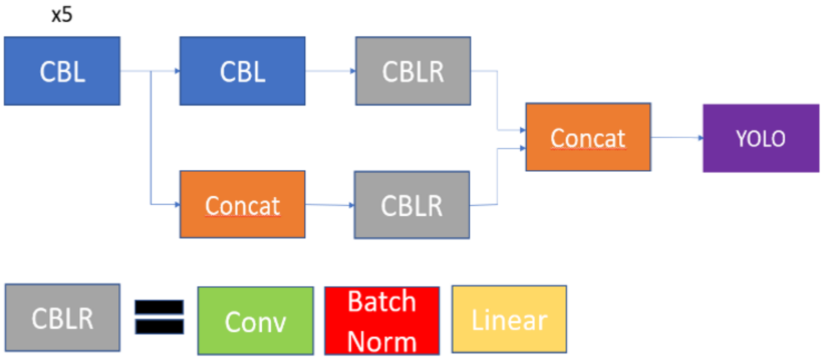


Figure 3. Proposed YOLOv4-embedding architecture.

The ReLU activation function is simple to implement and predicts various tasks well. ReLU provides an elementary nonlinear transformation. Given an element x, the ReLU function is defined as the maximum value of that element with 0. In other words, the ReLU function keeps only positive elements and discards all negative elements by setting the corresponding activity value to 0. The activation function is piecewise linear. When the input is negative, the derivative of the ReLU function is 0. When the input is positive, the derivative of the ReLU function is 1. Note that the input value is exactly 0 because the ReLU function is not differentiable. We use the derivative on the left by default when the input and derivative are 0. We can ignore this case because the input may never be 0. ReLU derivation behaves particularly well if either let the parameter disappears or the parameter is passed. This makes the optimization perform better, and ReLU alleviates the vanishing gradient problem that plagued previous neural networks.

The Leaky Relu activation function is widespread in deep learning, whereas the average expression is better of the Mish function. The Mish and Leaky Relu activation function combined in the YOLOv4-embedding backbone enhances detection correctness. The Leaky Relu activation function is used in the other part of the YOLOv4-embedding neck network. Specifically, the Mish function is:

$$f(x) = x * tanh(softplus(x)) \tag{1}$$

The Leaky ReLU function is:

$$\begin{cases} f(x) = x & if \quad x > 0 \\ f(x) = \lambda x & if \quad x \leq 0 \end{cases} \tag{2}$$

The Leaky Relu and Mish function figure is declared in Figure 4.

CSPx1 implies one Resnet part in the YOLOv4-embedding backbone; CSPx4 implies four Resnet parts. After data pass through the backbone, the input image size decreases from 512 to 16. According to Figure 2, the CSP module maps features into two sections for two convolution operations and combines the results, reducing the memory and increasing detection accuracy. The Resnet allows a deeper network, and extracted higher-level features.

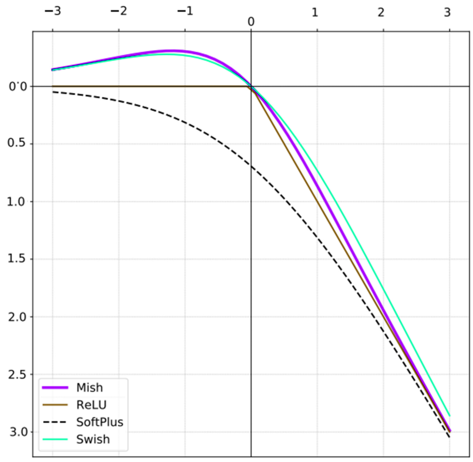


Figure 4. The Mish function and the Leaky ReLU function.

The external of marine organisms is abnormal, and marine organisms' hue is very similar to the circumstance. Furthermore, the background and attenuation significantly intervene with marine organisms' discovery in the ocean environment. Therefore, there is essential to extract special features to identify and detect marine organisms. The neck structure utilizes the SPP module (purple) and the FPN + PAN module in the network. In the SPP module, the property size of max-pooling employ 5 × 5, 9 × 9, and 13 × 13, and the stride progress subscribe 1 to keep the photo measurement unchanged.

Except for prescriptive max-pooling, the SPP module can enlarge the guarantee range of the backbone network feature. That also reforms the invention accurateness with lower computation expense. In YOLOv4-embedding Neck, FPN increases the size of the characteristic map through the up-sampling mission, merging tensor and dimension with the typical chart after the CSP task in the backbone network and transmitting the semantic target information. After down-sampled the aggregated characteristic chart through the convolution task, the PAN structure is merged with the feature map of the corresponding measurement in FPN to draw out positioning characteristics.

FPN + PAN combines specific disclosure and trunk level, using many dimensionalities to extract semantic information and deep positioning to find more small objects of various sizes. As an effect, the detection of tiny objects has been beneficial. For example, the fish sizes of different breeds exceed marine organisms' detection. Therefore, when other marine organisms' sizes show in one image, the detection generalization capability of the algorithm is essential, which will expound in the discussion part. The head structure in the network is the prediction part. The three-layer dimension feature maps (16 × 16; 32 × 32; 64 × 64) were produced from the convolution task and CBL module at the tail end of the YOLOv4-embedding Neck.

Each dimensionality forecast three anchor boxes, and there are five values per anchor: four box coordinates + one object confidence. Therefore, the percolator equals four classes + five values, multiplied by the prior box involving three anchors, and the prediction section has 27 produce. The output information can obtain the bounding box and dependence of the detected marine organisms.

When the bounding box dependence is inferior will delete the threshold, and the DIOU-NMS algorithm will choose the best candidate boxes. CIoU loss is brought forth by forcing the consistency of the field ratio. Formula (3) can restrict the loss function:

$$L_{CIoU} = 1 - IoU + \rho^2(b, b^{gt})/c^2 + \alpha v \tag{3}$$

α indicates the weight property, and v suggests the resemblance of the field ratio. After filtering by CIoU, the detection assignment and the detection educt are completed. Since CIOU takes the overlap rate, distance, penalty elements, and scale into account, the prediction box can quickly close in the field ratio of the actual frame during training, avoiding the issue of divergence in the training procedure.

4. Tentative Results

This paragraph illustrates the consequence of marine organisms detection in the training and detection period. Different occasions depicted the evaluation target, training parameters, and detection effects.

4.1. Experimental Setup

There are 500 training steps in MS COCO object detection experiments, while the batch size and the mini-batch size are 64 and 4, respectively. All architectures use a single GPU to execute multiscale training.

4.2. Assess Training Models

The training section assesses the recap capability and progressively optimizes the precision, recall, model, and the mAP score employed as an assessment goal.

$$Precision = (TP/(TP + FP)) * 100 \tag{4}$$

$$Recall = (TP/(TP + FN)) * 100 \tag{5}$$

$$mAP = (AP_1 + AP_2 + ...AP_n)/n \tag{6}$$

Precision indicates correct predictive value, TP indicates true right, FP represents false correct, and FN represents false negative.

The batch magnitude subscribed to eight, and each iteration fetched eight images. The network trained one thousand five hundred fifty-seven images. Therefore, one epoch demanded 103 iterations. The weight consequence of each era verified in the validation set. The threshold could receive a collection of recall and precision of the pattern. Numerous groups of accuracy and recall would receive different thresholds subscribed for the design, the area of the Recall-Precision region is the average exactness. Three training stages were subscribed, with the most extensive epoch of 100, 200, and 300.

The weight selected the maximum mAP conforming in each training, the precision and recall produced when the threshold was 0.5 and 0.75, compare the capability of the three epochs of training, the performance of the mAP in YOLOv4 and YOLOv4-embedding as shown in Table 1, the performance of the Precision and Recall in YOLOv4 and YOLOv4-embedding as shown in Table 2.

As the epoch and the number of iterations increased, the mAP increased and applied more time. The highest mAP reached 82.68% in YOLOv4 at the 300 epochs for mAP_{75},whereas the mAP_{75} got an mAP value of 85.6% in YOLOv4-embedding. Further, explain the assess indexes of 300 epochs in the YOLOv4-embedding training process. In the epoch training, the mAP numerical curves are shown in Figure 5. The mAP figure can be near the value of 0.9 and converge swiftly. After the 150th epoch, the precision figure is more steady. Therefore, the optimal weight pattern chose as the marine organisms detection pattern based on the YOLOv4-embedding arithmetic in the epoch 300 training.

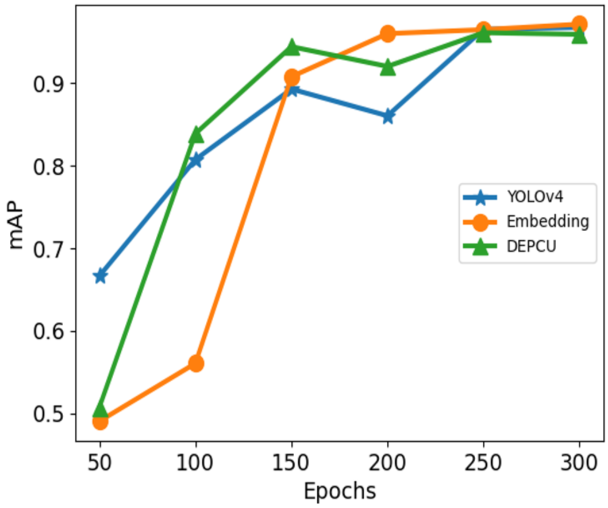


Figure 5. Evaluation index curve in mAP.

Table 1. Performance of the mAP.

Network	Epoch	mAP_{50}	mAP_{75}
YOLOv4	100	0.9676	0.625
	200	0.9717	0.7209
	300	0.9709	0.8268
YOLOv4-embedding	100	0.9673	**0.7085**
	200	0.9675	**0.7651**
	300	**0.9719**	**0.856**

Table 2. Performance of the Precision and Recall.

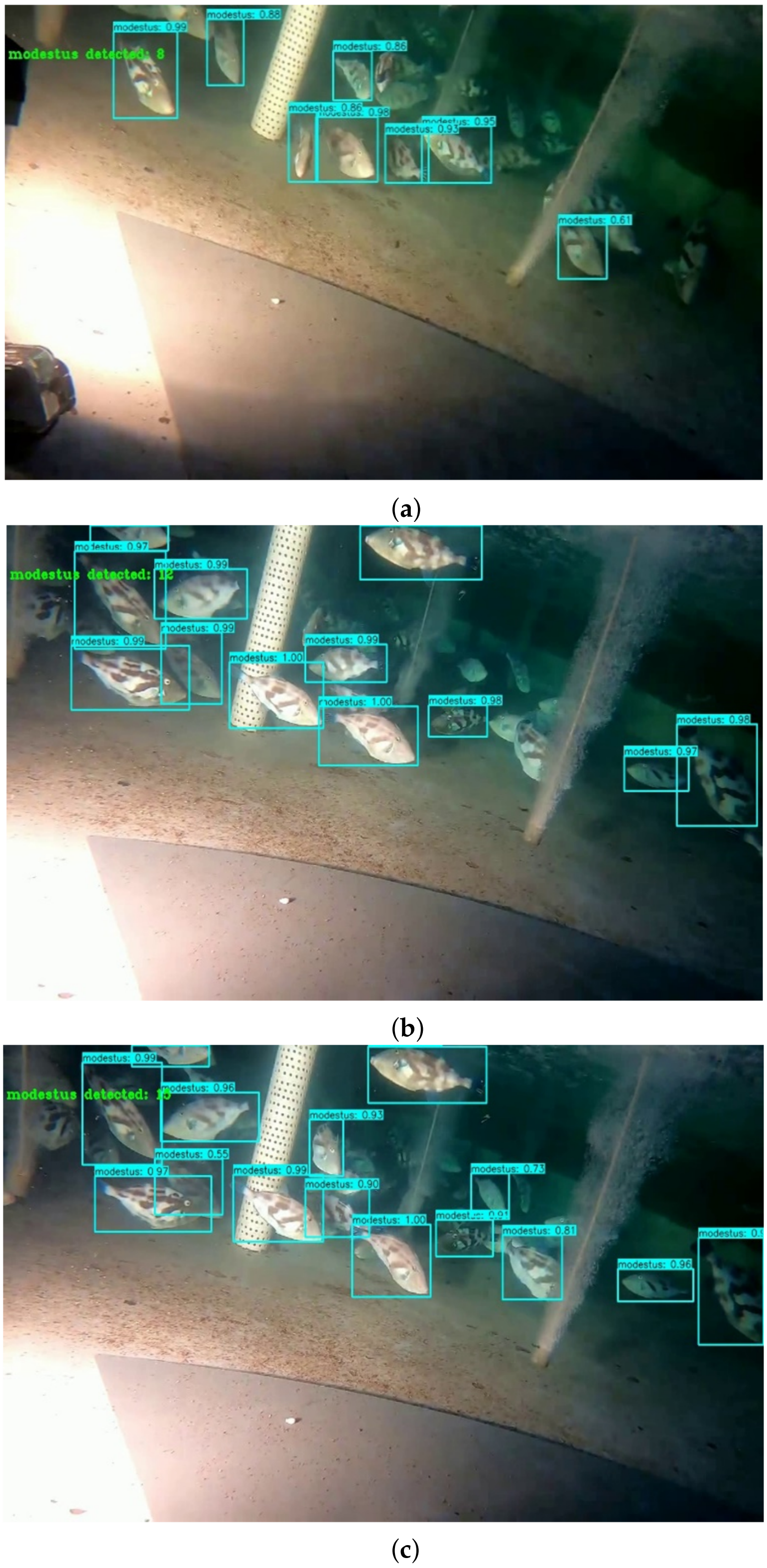
Network	Epoch	Precision	Recall
YOLOv4	100	0.87	0.92
	200	0.86	0.95
	300	0.86	0.95
YOLOv4-embedding	100	0.78	0.74
	200	0.85	0.76
	300	0.86	0.82

4.3. Detection Consequence

Different environmental conditions examined the trained marine organisms detection pattern. Three cases were chosen in each situation. Figure 6a–c show the detection consequences of the camera in each situation. Figure 6c shows camera is closer to marine organisms than Figure 6a,b, and more marine organisms are detected.

Figure 7a–c show that 8, 12, and 15 marine organisms were detected respectively. Whether the marine organisms are partially irradiated or completely irradiated by the light source, marine organisms could detect by YOLOv4-embedding. the detection confidence of marine organisms is higher if the light source is sufficient in the farming pool.

Figure 7 indicates the detection results far away from the light source condition. Figure 7b is farther away from the light source than Figure 7c. Figure 7a is farther away from the light source than Figure 7b. The orange wire demonstrates the distance from the light source to the detection goal. Marine object detection confidence in Figure 7c is 100%. The object detection confidence in Figure 7b is 98%. The object detection confidence in Figure 7a is 78%. The detection confidence of marine organisms is lower if the light source is insufficient. Otherwise, the detection confidence of marine organisms is higher if the light source is close to the detection object.

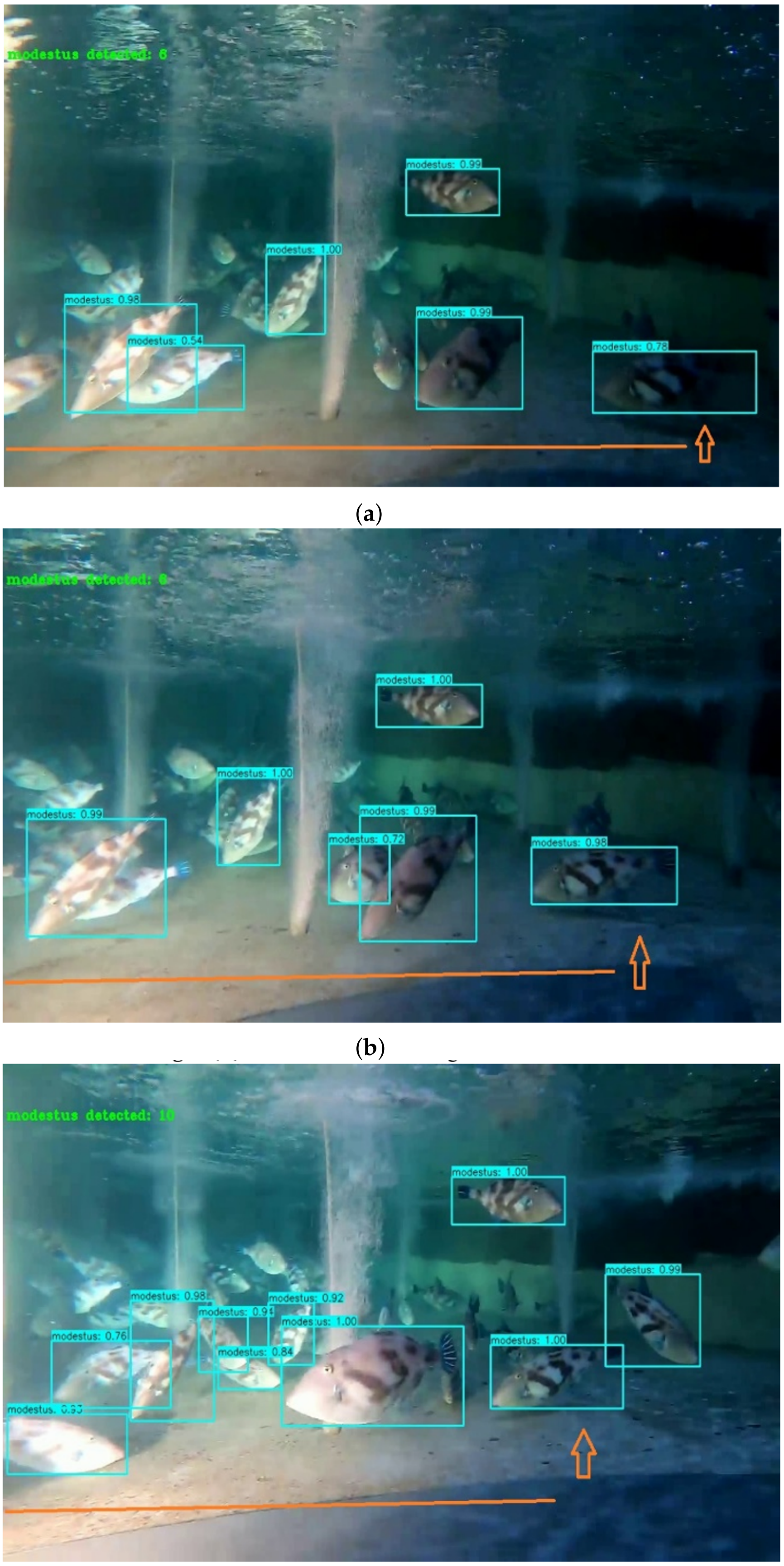


(**a**)

(**b**)

(**c**)

Figure 6. Frame detection results (camera) close to marine organisms. (**a**) Camera at a long distance from marine life; (**b**) Camera at a moderate distance from marine life; (**c**) Camera closest to marine organisms.



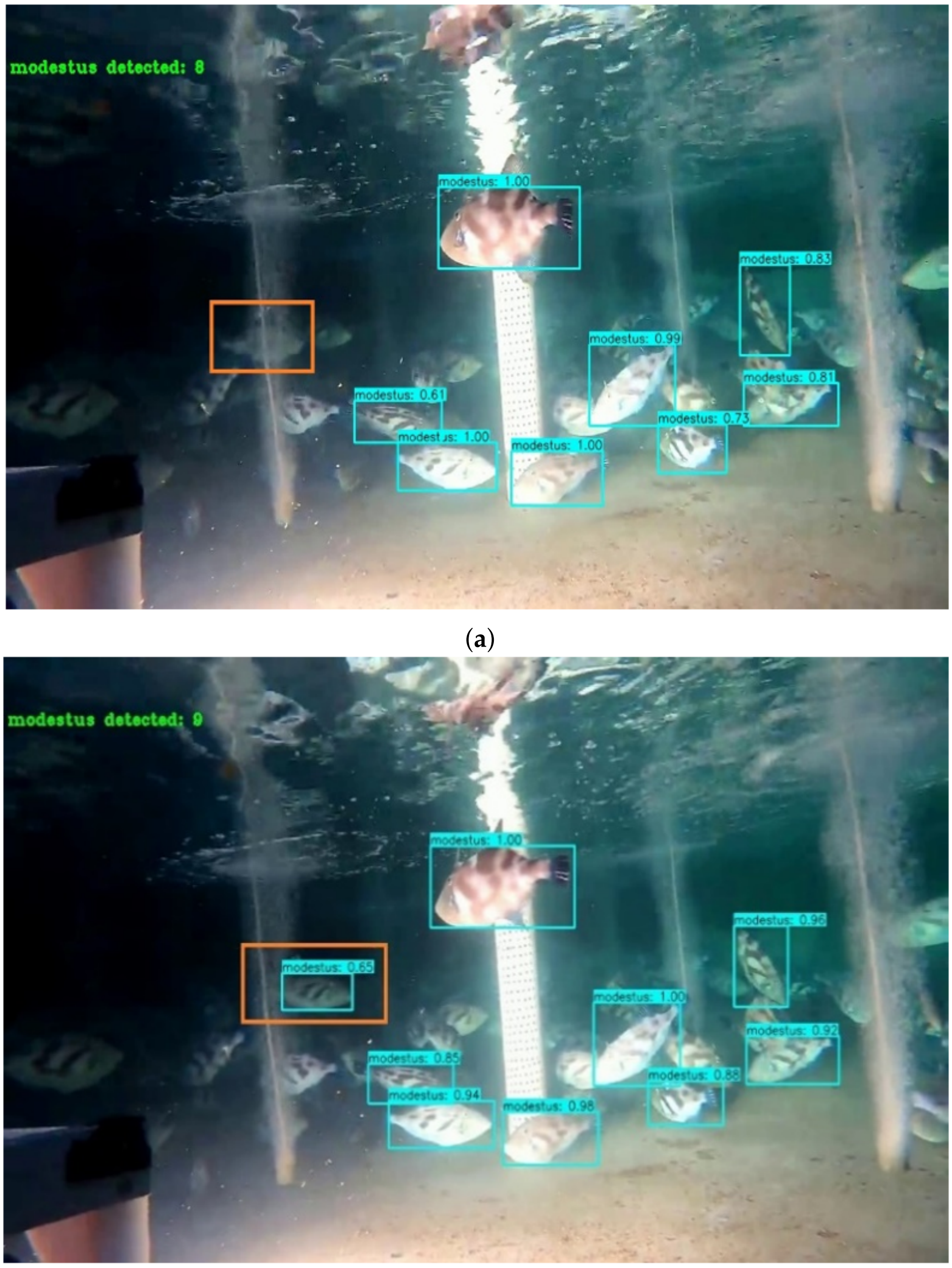
(**a**)

(**b**)

(**c**)

Figure 7. Detection results far away from the light source condition. (**a**) The longest distance to the orange line; (**b**) Middle distance to the orange line; (**c**) The shortest distance to the orange line.

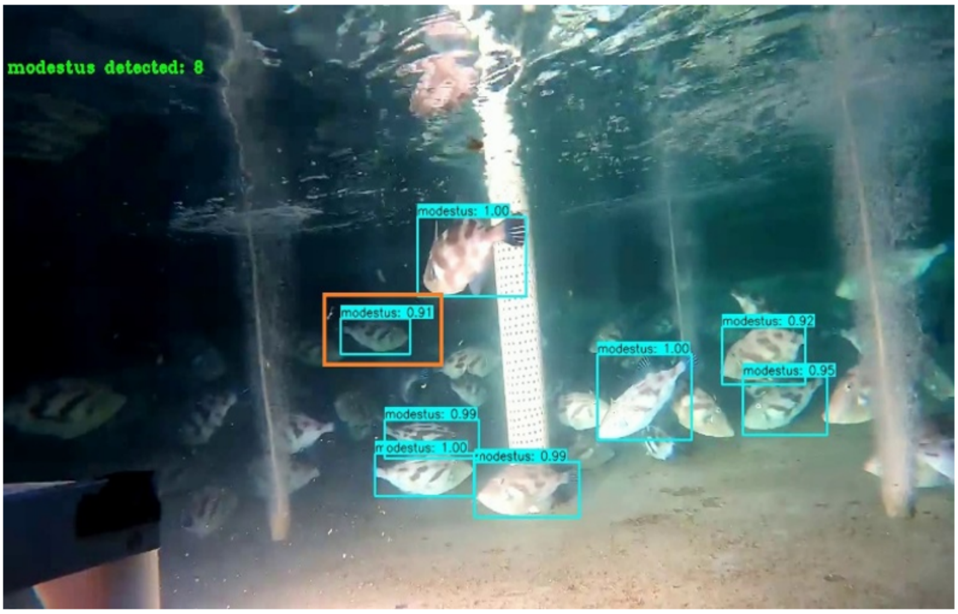
Marine organisms’ detection consequences under different occlusion extent were different, and classified as an occlusion conditions. Therefore, examining the trained detection pattern as shown in Figure 8a at different occlusion extent, extensive zone occlusion(noted by the orange frame) affects the detect result cause the information about the marine organism was little. Figure 8b showed when the occlusion zone decreased, the marine organisms’ confidence was 65%. The information on the marine organism in Figure 8c was almost complete; the marine organism’s confidence is 91%. Occlusion in Figure 8a,b often happens in continuous detection. Accurate detection in succeeding frames of all marine organisms with significant meaning for solving the issue of duplicated detection, while YOLOv4-embedding arithmetic has stronger robustness to the environmental conditions.



(**a**)

(**b**)

Figure 8. *Cont.*



(**c**)

Figure 8. Detection results far away from the light source condition. (**a**) Large zone occlusion by water column (noted by orange box); (**b**) Less occlusion by water column (noted by orange box); (**c**) No occlusion by water column (noted by orange box).

5. Discussion

As explained in the previous section, we revised the YOLOv4 to obtain the YOLOv4-embedding. As a tool for modifying the structure of the YOLOv4-embedding, we used the Netron tool. Netron is a viewer for neural networks, deep learning, and machine learning models. Our experiments were conducted on an Intel I7-10700 Processor with the main memory of 16GB and an NVIDIA GeForce RTX 3090. Based on the YOLOv4-embedding, we trained the model with the marine datasets. The hyper-parameters for the YOLOv4-embedding training are as follows; the momentum and weight decay are set as 0.949 and 0.0005, the batch size and the mini-batch size are 8 and 1. We also used the following values for the parameters:

- The training steps are 30,938.
- The scheduling strategy's polynomial decay learning rate adopted with an initial learning rate of 0.0013.
- The iteration steps are 1000.

5.1. Comparison of YOLOv4 and YOLOv4-Embedding in Marine Organism Detection

The YOLOv4-embedding neural network trained and detected the marine organism dataset. The epoch was set as 300, and validation selected the optimal training model with a mAP_{50} 97.19%. Figure 9 shows the YOLOv4-embedding curve wrapped on the outside of the YOLOv4 curve, and the position of the balance point (Precision = Recall) is closer to coordinate (1, 1), so the performance of the YOLOv4-embedding model is better. As referred upon, YOLOv4-embedding collects the trunk layer by using FPN + PAN architecture and detection layer features via multi-scales with significant meaning for improving small objects network detection again and again. YOLOv4-DEPCU changes the structure of CBM to DEPCU (deep convolution), which consists of 3 modules (CBL + CBM + CBL).

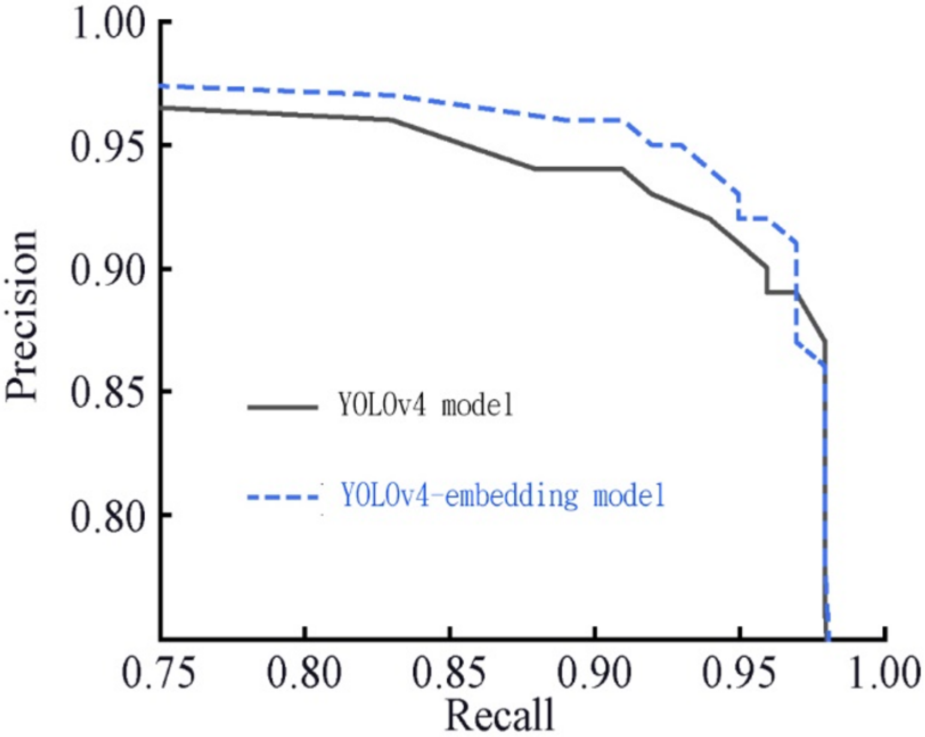
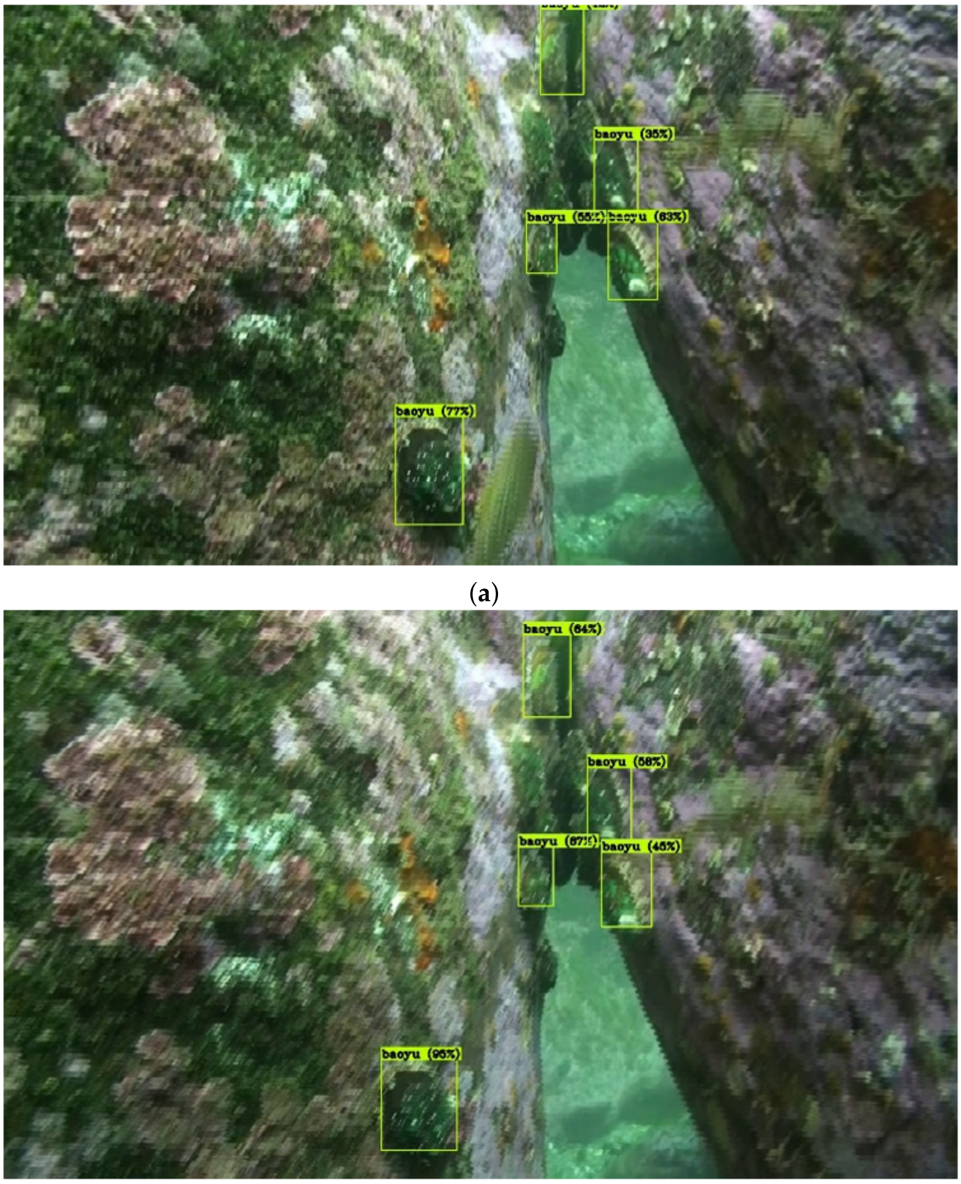


Figure 9. Precision/Recall curves of different methods on test sets.

Since many kinds of marine organisms in the dataset were deliberated, and the distance between marine organisms is comparatively vast, the measurement of marine organisms of different breeds and distances varies hugely in one image. Therefore, YOLOv4 could not detect teetotally, resulting in a lower mAP value. Two detection ways are slightly distinct in object detection from the detection consequences.

While YOLOv4-embedding use the real-time ocean environment as the background. Similarly, Figure 10 shows Haliotis is detected. Both algorithm have precise invention on Haliotis, and YOLOv4-embedding gets superior confidence in the same box. Compared to YOLOv4-embedding and YOLOv4, YOLOv4 will misjudge the measuring of small objective. However, YOLOv4-embedding made a precise invention. Because of the use of tricks and FPN + PAN architecture. Small marine organism object detection has significant implication to the operation board of farming pools. First, YOLOv4-embedding object invention can reasonably decide distinct varieties; Moreover, as shown in Figure 11 the accurate invention of the tiny marine organism can offer a advantageous message for continuous detection.

Finally, we caught the marine organism image at a different angle. Though a horizontal angle can detect most marine organisms, some may not swim horizontally. Therefore, we executed experiments on the marine organism images with an angle of 45 degrees to examine the marine organism detection. Figure 12 shows the detection result of YOLOv4-embedding and YOLOv4 in the 45 angles, YOLOv4-embedding precisely detected the marine organisms, but YOLOv4 detected fewer counts of marine organisms. Because YOLOv4-embedding has more generalization capability of marine organisms detection in different angles.



(**a**)

(**b**)

Figure 10. Detection results of Haliotis. (**a**) Marine organism detected by YOLOv4; (**b**) Marine organism detected by YOLOv4-embedding.

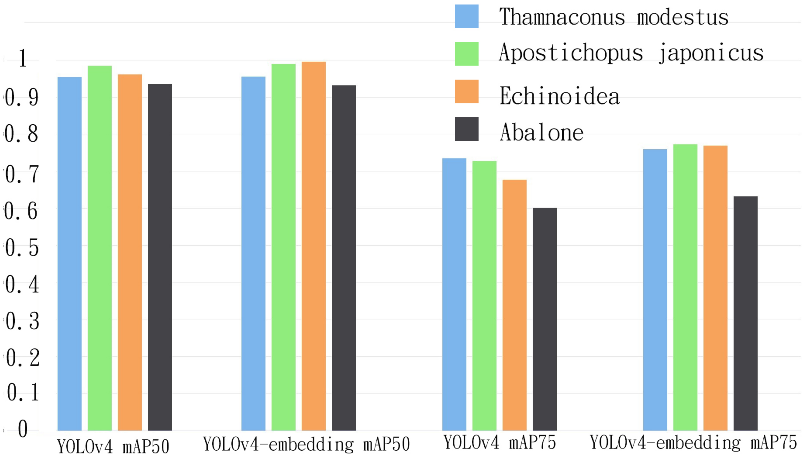
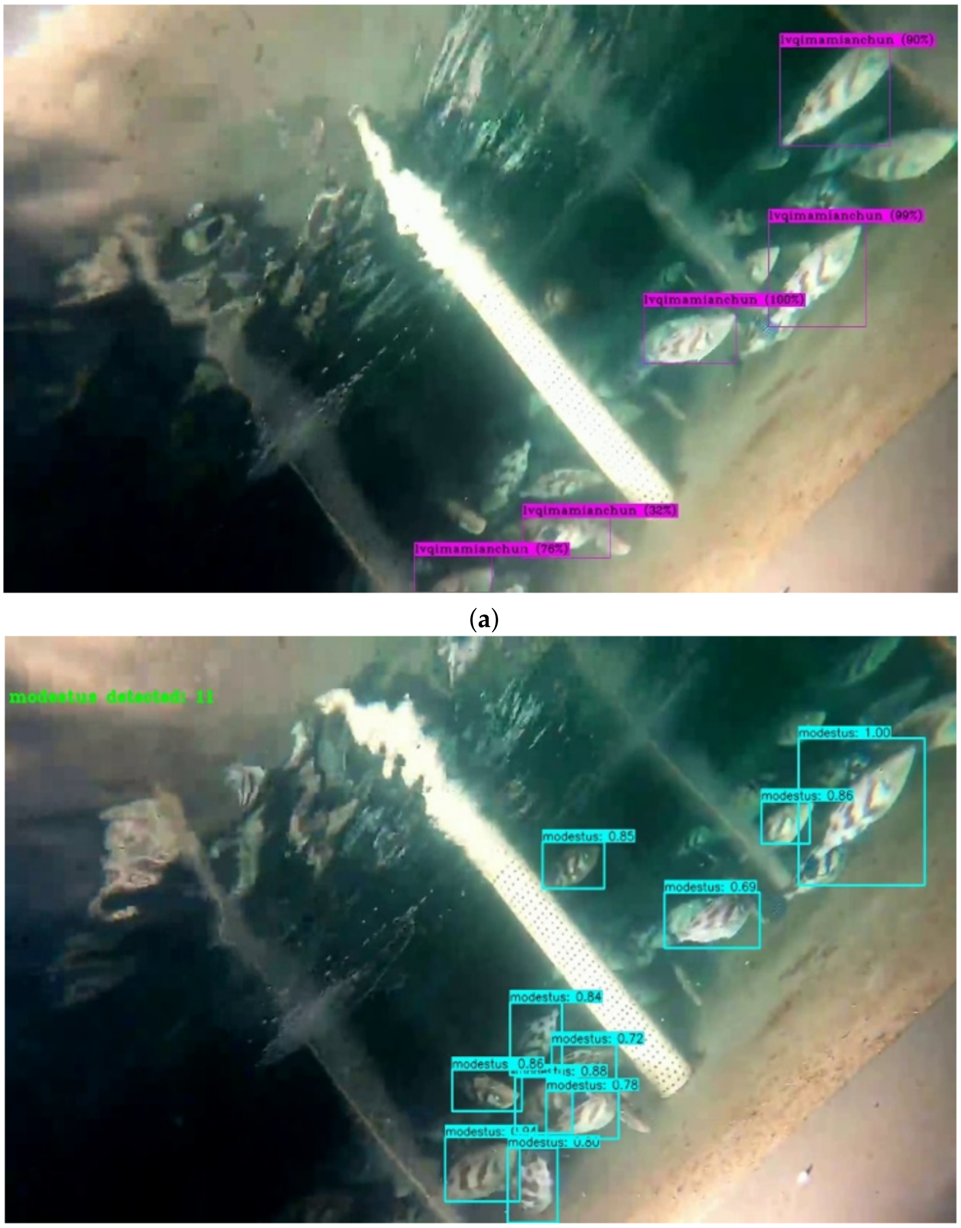


Figure 11. Accuracy results of each marine organism category in the model.



(**a**)

(**b**)

Figure 12. Comparison of the detection consequences taken at 45 angles. (**a**) YOLOv4 detection result; (**b**) YOLOv4-embedding detection result.

5.2. Comparison of YOLOv4 and YOLOv4-Embedding in Marine Organism Detection

Machine learning algorithm has a lower running cost, smaller weight measurement, shorter training time, and realized on the CPU, which does not require GPU, but lower detection ratio and longer detection time. Therefore, we compared marine organism targets in deep learning algorithms.

Table 3 compares the critical goal of the four algorithms in the same test set. YOLOv4-embedding requires GPU to get the optimal pattern and 300 epochs for training. The weight measurement and training time of YOLOv4-embedding were similar to YOLOv4, but mAP was higher.

Table 3. Detection indexes of the four arithmetics.

Arithmetic	EfficientDet-D3	YOLOv4	v4-Embedding	v4-DEPCU
mAP_{50}	71.20%	97.09%	**97.19%**	95.90%
mAP_{75}	61.60%	82.68%	**85.60%**	84.75%
Average detection time	72 ms	19.31 ms	19.46 ms	20.20 ms
Weight size	46.33 MB	244.22 MB	244.55 MB	270.90 MB
Training time (300 epoch)	5.47 h	5.98 h	6.08 h	8.40 h
FPS	43	51.8	51.4	49.5

The mAP_{75} of YOLOv4 was 82.68%, which was smaller than YOLOv4-DEPCU and YOLOv4-embedding algorithm. YOLOv4-embedding disclosed the marine organism with mAP_{75} 85.6%, YOLOv4-DEPCU disclosed the marine organism with mAP_{75} 84.75%. The average invention time of YOLOv4 was 19.31 ms, and the YOLOv4-DEPCU invention time was 20.20 ms. Since the network of YOLOv4-DEPCU is deeper than YOLOv4, the invention time also increased. The CIOU algorithm improves the confidence of marine organism invention results, and the YOLOv4-embedding algorithm performs 60.8% AP50 for the MS COCO dataset. The invention performance of YOLOv4-embedding is better than the EfficientDet-D3 algorithm for marine organism detection in the farming pool. Compared with YOLOv4, YOLOv4-embedding, and YOLOv4-DEPCU, the average invention time of the three algorithms was 19.31 ms,19.46 ms, and 20.20 ms. The mAP_{75} of marine organisms was 82.68%, 85.6%, and 84.75%, respectively. YOLOv4-embedding obtained the optimal weight model in the test section with 300 epochs iterations. Moreover, higher FPS is still the feature of YOLOv4-embedding. Overall, YOLOv4-embedding could get the optimal weight model with iterations in the training level, and superior to the traditional machine learning algorithm and deep learning algorithm like EfficientDet-D3, YOLOv4-embedding has high confidence and high detection ratio at the invention level.

6. Conclusions and Future Works

The accurate marine organisms detection is significant to the farming pool intelligent management. This paper proposed a method based on the one-stage neural network for marine organism invention in natural prerequisite. Besides, we analyzed the property of the YOLOv4-embedding and EfficientDet-D3 algorithm in marine organism invention. According to the experimental outcome, the following completion can summarize below:

(i) We found suitable deep learning algorithm for marine organism invention in the farming pool. This paper analyzed the structural characteristic of the YOLOv4-embedding neural network and the key issues of farming pool invention. In the network, CSP-Darknet53 deepens the network that could collect more deep marine organism features and reduce the interference of background; Embedding the linear activation and convolution layer at the end of the YOLOv4-embedding Neck. YOLOv4-embedding architecture increases the acceptance range of network characteristics with less computational expense. Furthermore, Conv + Batch normalization + Liner(CBLR) employed in the network, which extracts more profound semantic information and positioning information repeatedly, detects marine organisms more accurately. Therefore, accurate detection is achieved when the marine organism measurement are highly different in one image.

(ii) The marine organism detection arithmetic using the YOLOv4-embedding neural network in the farming pool, achieves precise detection under different occlusion and illumination for other matureness and breeds, providing accurate information for other animals breeds and maturity of the marine organism intelligent management and underwater machinery.

(iii) YOLOv4-embedding neural networks are built to achieve the fast diversity of marine organisms. The image feature extraction by deep learning and intermediate data of

CNN when the network training is steady are studied. This paper also discussed the effect of convolution embedded scheme, training rounds, and sample number on network training speed and accuracy. In the detection section, YOLOv4-embedding with high confidence and high mAP. In conclusion, the proposed architecture is suitable for marine organism detection in the farming pool. The future work will mainly get the assort value of marine organisms in the real world, realize the localization of marine organisms, compute the picking dot's position, deploy a marine organism invention model in tiny terminals, and develop moving mechanics in future work.

Author Contributions: Writing—original draft, J.Z., W.H., W.W. and G.M.; Writing—review & editing, T.Z., Y.M., X.Y. and P.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Key Research and Development Plan of China (2019YFD0900805) and in part by the construction of an information platform for the whole process of intelligent ecological aquaculture in a shallow sea (KY310169).

Institutional Review Board Statement: Not applicable

Informed Consent Statement: Not applicable.

Acknowledgments: This work was supported by the National Key Research and Development Plan of China (2019YFD0900805) and in part by the construction of an information platform for the whole process of intelligent ecological aquaculture in a shallow sea (KY310169).

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

References

1. Zhao, Z.; Liu, Y.; Sun, X.; Liu, J.; Yang, X.; Zhou, C. Composited FishNet: Fish Detection and Species Recognition From Low-Quality Underwater Videos. *IEEE Trans. Image Process.* **2021**, *30*, 4719–4734. [CrossRef] [PubMed]
2. Yang, L.; Liu, Y.; Yu, H.; Fang, X.; Song, L.; Li, D.; Chen, Y. Computer vision models in intelligent aquaculture with emphasis on fish detection and behavior analysis: A review. *Arch. Comput. Methods Eng.* **2021**, *28*, 2785–2816. [CrossRef]
3. Kim, S.; Jeong, M.; Ko, B.C. Self-supervised keypoint detection based on multi-layer random forest regressor. *IEEE Access* **2021**, *9*, 40850–40859. [CrossRef]
4. Chen, L.; Liu, Z.; Tong, L.; Jiang, Z.; Wang, S.; Dong, J.; Zhou, H. Underwater object detection using Invert Multi-Class Adaboost with deep learning. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–8.
5. Zhang, X.; Zhang, L.; Lou, X. A Raw Image-Based End-to-End Object Detection Accelerator Using HOG Features. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2022**, *69*, 323–333 [CrossRef]
6. Odaudu, S.N.; Adedokun, E.A.; Salaudeen, A.T.; Marshall, F.F.; Ibrahim, Y.; Ikpe, D.E. Sequential feature selection using hybridized differential evolution algorithm and haar cascade for object detection framework. *Covenant J. Inform. Commun. Technol.* **2020**, *8*, 2354–3507 [CrossRef]
7. Smotherman, H.; Connolly, A.J.; Kalmbach, J.B.; Portillo, S.K.; Bektesevic, D.; Eggl, S.; Juric, M.; Moeyens, J.; Whidden, P.J. Sifting through the Static: Moving Object Detection in Difference Images. *Astron. J.* **2021**, *162*, 245. [CrossRef]
8. Pramanik, A.; Harshvardhan; Djeddi, C.; Sarkar, S.; Maiti, J. Region proposal and object detection using HoG-based CNN feature map. In Proceedings of the 2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI), Sakheer, Bahrain, 26–27 October 2020; pp. 1–5.
9. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems 25 (NIPS 2012), 2012; Volume 25. Available online: https://www.researchgate.net/publication/319770183_Imagenet_classification_with_deep_convolutional_neural_networks (accessed on 1 February 2022).
10. He, T.; Zhang, Z.; Zhang, H.; Zhang, Z.; Xie, J.; Li, M. Bag of tricks for image classification with convolutional neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 558–567.
11. Duan, K.; Xie, L.; Qi, H.; Bai, S.; Huang, Q.; Tian, Q. Corner proposal network for anchor-free, two-stage object detection. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2020; pp. 399–416.
12. 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.
13. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.

14. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
15. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the International Conference on Machine Learning, 2015; pp. 448–456. Available online: https://proceedings.mlr.press/v37/ioffe15.html (accessed on 1 February 2022).
16. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
17. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, inception-resnet and the impact of residual connections on learning. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, 2017. Available online: https://ojs.aaai.org/index.php/AAAI/article/view/11231 (accessed on 1 February 2022).
18. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
19. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
20. Aziz, A.; Sohail, A.; Fahad, L.; Burhan, M.; Wahab, N.; Khan, A. Channel boosted convolutional neural network for classification of mitotic nuclei using histopathological images. In Proceedings of the 2020 17th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad, Pakistan, 14–18 January 2020; pp. 277–284.
21. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 21–37.
22. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
23. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems 28 (NIPS 2015), 2015; Volume 28. Available online: https://dl.acm.org/doi/10.5555/2969239.2969250 (accessed on 1 February 2022).
24. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
25. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
26. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
27. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. *IEEE Int. Conf. Comput. Vis.* **2017**, *42*, 318–327.
28. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10781–10790.
29. Niu, B.; Li, G.; Peng, F.; Wu, J.; Zhang, L.; Li, Z. Survey of fish behavior analysis by computer vision. *J. Aquac. Res. Dev.* **2018**, *9* . Available online: https://www.researchgate.net/publication/325968943_Survey_of_Fish_Behavior_Analysis_by_Computer_Vision (accessed on 1 February 2022). [CrossRef]
30. Woźniak, M.; Połap, D. Soft trees with neural components as image-processing technique for archeological excavations. *Pers. Ubiquitous Comput.* **2020**, *24*, 363–375. [CrossRef]
31. Rathi, D.; Jain, S.; Indu, S. Underwater fish species classification using convolutional neural network and deep learning. In Proceedings of the 2017 Ninth International Conference on Advances in Pattern Recognition (ICAPR), Bangalore, India, 27–30 December 2017; pp. 1–6.
32. Marini, S.; Fanelli, E.; Sbragaglia, V.; Azzurro, E.; Del Rio Fernandez, J.; Aguzzi, J. Tracking fish abundance by underwater image recognition. *Sci. Rep.* **2018**, *8*, 1–12. [CrossRef] [PubMed]
33. Mandal, R.; Connolly, R.M.; Schlacher, T.A.; Stantic, B. Assessing fish abundance from underwater video using deep neural networks. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–6.
34. Konovalov, D.A.; Saleh, A.; Bradley, M.; Sankupellay, M.; Marini, S.; Sheaves, M. Underwater fish detection with weak multi-domain supervision. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–8.
35. Uemura, T.; Lu, H.; Kim, H. Marine organisms tracking and recognizing using yolo. In *2nd EAI International Conference on Robotic Sensor Networks*; Springer: Cham, Switzerland, 2020; pp. 53–58.
36. Wang, C.Y.; Liao, H.Y.M.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W.; Yeh, I.H. CSPNet: A new backbone that can enhance learning capability of CNN. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020; pp. 390–391.
37. 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]
38. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.