


Article

Improved YOLOv8-Pose Algorithm for Albacore Tuna (*Thunnus alalunga*) Fork Length Extraction and Weight Estimation

Yuqing Liu ¹, Zhou Fu ¹, Liming Song ^{2,3,*} , Hengshou Sui ⁴ and Bin Li ⁴

¹ College of Engineering Science and Technology, Shanghai Ocean University, Shanghai 201306, China; yqliu@shou.edu.cn (Y.L.); rileyfuzhou@gmail.com (Z.F.)

² College of Marine Living Resource Sciences and Management, Shanghai Ocean University, Shanghai 201306, China

³ National Engineering Research Center for Oceanic Fisheries, Shanghai 201306, China

⁴ CNFC Overseas Fisheries Co., Ltd., Beijing 100032, China

* Correspondence: lmsong@shou.edu.cn

Abstract: Aiming at the problems of large statistical error and the poor real-time performance of catch weight in the ocean fishing tuna industry, an algorithm based on improved YOLOv8-Pose for albacore tuna (*Thunnus alalunga*) fork length extraction and weight estimation is proposed, with reference to the human body's pose estimation algorithm. Firstly, a lightweight module constructed using a heavy parameterization technique is used to replace the backbone network, and secondly, a weighted bidirectional feature pyramid network BIFPN is utilized. Finally, the upper and lower jaw and tail feature points of the albacore tuna (*Thunnus alalunga*) were extracted using the key point detection algorithm, and the weight of the albacore tuna (*Thunnus alalunga*) was estimated based on the fitted relationship between fork length and weight. The experimental results show that the improved YOLOv8-Pose algorithm reduces the number of model parameters by 13.63% and the number of floating-point operations by 14.03% compared with the baseline model without decreasing the accuracy of the target detection and key point detection and improves the model inference speed by 374%. At the same time, it reduces the drift of the key point detection, and the error of the comparison with the actual albacore tuna (*Thunnus alalunga*) body weight is not more than 10%. The improved key point detection algorithm has high detection accuracy and inference speed, which provides accurate yield data for pelagic fishing and is expected to solve the existing statistical problems and improve the accuracy and real-time performance of data in the fishing industry.

Keywords: key point detection; pelagic fishing; detection drift; lightweight



Citation: Liu, Y.; Fu, Z.; Song, L.; Sui, H.; Li, B. Improved YOLOv8-Pose Algorithm for Albacore Tuna (*Thunnus alalunga*) Fork Length Extraction and Weight Estimation. *J. Mar. Sci. Eng.* **2024**, *12*, 784. <https://doi.org/10.3390/jmse12050784>

Academic Editor: Francesco Tiralongo

Received: 10 April 2024

Revised: 2 May 2024

Accepted: 5 May 2024

Published: 8 May 2024



Copyright: © 2024 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

In the field of pelagic fisheries, to strengthen regulatory capacity building and enhance the efficiency of fishery management, technical means, such as vessel position monitoring, electronic fishing logbook, remote video monitoring, high seas transportation supervision, and product traceability, have been introduced to build a comprehensive regulatory system for pelagic fisheries. Among these, the statistics on the catch yield data of tuna fishing have become an important step in completing the work of electronic fishing logbooks. Traditionally, fishery statistics usually rely on the manual recording of the fishing logbook [1]. With the development of fishery science, modern fisheries have gradually adopted a “from science to management” approach, transitioning from past non-systematic, empirical fishery decisions to evidence-based fishery management [2]. This trend underscores the increasing significance of fishery resource assessment in fishery management. When under a severe fishing environment and heavy catch statistics task, manual recording is prone to visual fatigue and physical fatigue, which leads to the accuracy of the statistics being affected. This situation directly affects the fishery resource assessment and the accuracy of tuna

fishery forecasting [3]. Therefore, how to measure tuna weight quickly and accurately is one of the key problems to be solved.

In recent years, with the rapid development of artificial intelligence and deep learning technology, this technology has been well applied in complex tasks such as face recognition [4,5], target detection [6,7], and human posture estimation [8,9]. At the same time, the development of the fishery industry has also ushered in a change, in which the application of sensors, machine vision, pattern recognition, and other technologies in the fishery has brought technological solutions for tuna weight estimation. Machine vision is a new cross-disciplinary science that combines image processing, image understanding, pattern recognition, and artificial intelligence, using machines rather than human vision to perceive, identify, and analyze scenes [10]. However, these techniques have not been widely used in pelagic fisheries, especially for the length and weight measurement of fish. Strachan [11] was the first to apply computer vision technology to the study of the measurement of fish body size traits. By controlling the lighting conditions during shooting, Strachan adopted a simple light threshold to detect fish body length. Abdullah et al. [12] estimated the head and tail positions of *Rastrelliger brachysoma* and *Selar crumenophthalmus* using edge and angle detection methods and then used this information to estimate the length of fish. Most of the existing algorithms for estimating fish body length and weight built on machine vision techniques are based on instance segmentation algorithms, which are computationally intensive and costly. Garcia et al. [13] used Mask R-CNN to locate and segment the fish in the image and then used a local gradient to refine the segmentation results to obtain an accurate estimation of the fish body boundary to measure fish body size. Key point detection was initially applied in human posture estimation. Jeong et al. [14] used the Openpose algorithm to realize the detection of smoking behavior and judged whether there was smoking behavior through the skeleton map composed of human key points. At present, there are also many researchers applying the key point detection technology to fishery production. Yu et al. [15] proposed a method based on key point detection to solve the problem of automatic measurement of fish length, but the prediction results would be inaccurate due to deviation in the annotation of different key points. Because of the complex underwater environment, Suo et al. [16] proposed a fish key point location method based on target detection and a point regression model, which can accurately locate the key point area, but the prediction ability of specific points still needs to be verified.

The YOLOv8-Pose algorithm model has the advantages of faster convergence speed, lower error rate, and a strong ability to locate feature points; therefore, this paper proposes a tuna fork length extraction and weight estimation algorithm based on the key point detection algorithm. As a key component of global fisheries, tuna has significant economic, ecological, and cultural value. Globally, tuna is not only an important commodity for fishery exports but also provides significant employment opportunities and economic benefits to coastal states. In addition, tuna species play an important role in the marine food chain and are vital for maintaining ecological balance. Because albacore tuna (*Thunnus alalunga*) is the most important species fished, its catch and resource management directly affect the global marine ecosystem and fishery market. With the improved YOLOv8-Pose algorithm accurately counting the species and weight of albacore tuna (*Thunnus alalunga*) caught, the fishery sector can better grasp the production of the entire voyage and thus optimize transportation and market supply. At the same time, scientific fishery strategies and production allocation can prevent overfishing and ensure the sustainable use of fish resources, providing strong technical support for tuna fisheries and helping to promote the sustainable development of global fisheries.

2. Materials

2.1. Data Acquisition

The sample data were collected in the South Pacific Ocean, and the test vessel was “Zhongshui 747” of CNFC Overseas Fishery Co., Ltd., Beijing, China. The target was albacore tuna (*Thunnus alalunga*), the predominant species in pelagic fisheries, and the

camera was installed at a vertical height of 210 cm from the deck, as shown in Figure 1. A Hikvision camera with model number DS-2SC3Q120MY-TE (Hangzhou Hikvision Digital Technology Co., Ltd., Hangzhou, China) and a horizontal field of view of 97 degrees and a vertical field of view of 52.3 degrees was used for 79 days of shooting. To increase the robustness of the dataset, the tuna were photographed in various postures in the afternoon, evening, and other normal operating periods, including under sunny, cloudy, rainy, and backlit conditions. An external portable hard disk was used for image storage, and a total of 935 tuna images were taken during the test period, with a uniform resolution of 2560×1440 (pixels), and 318 tuna-related fork length and weight datapoints were collected at the same time.

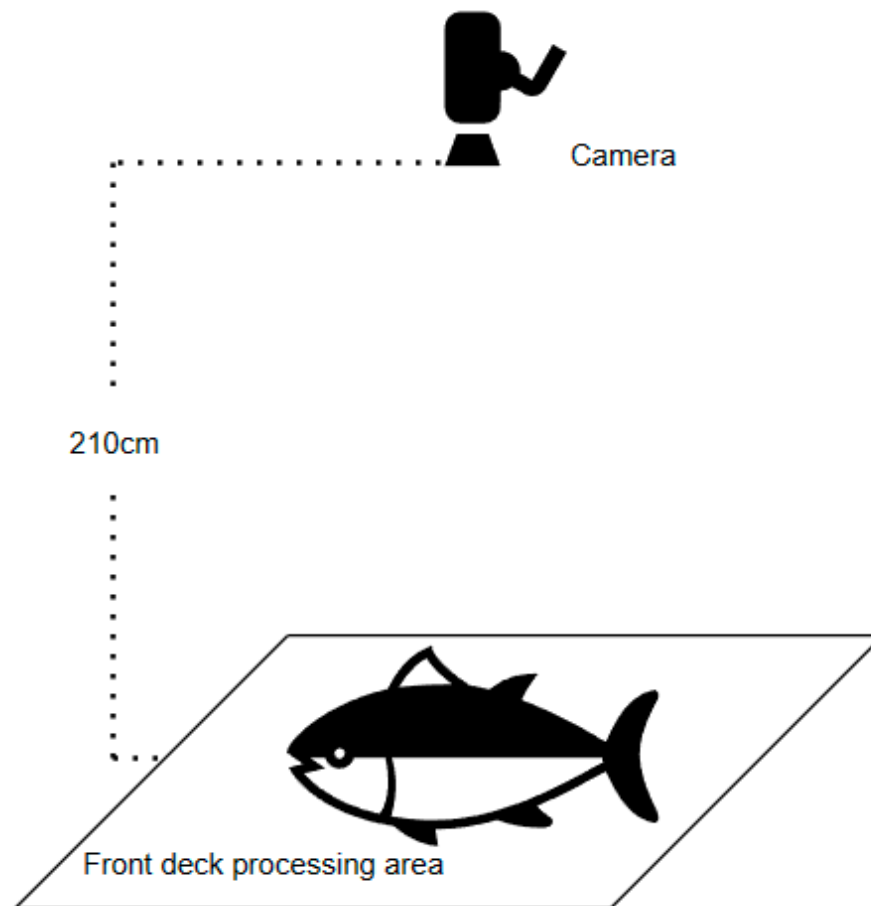


Figure 1. Data acquisition method.

2.2. Dataset Production

In this study, we used Labelme, a lightweight graphical annotation software, to label the albacore tuna (*Thunnus alalunga*) with the red minimum outer rectangular box of the target, labeled as “albacore”. The green key point “upper jaw” is labeled at the upper jaw of the tuna, the yellow key point “lower jaw” is labeled at the lower jaw of the tuna, and the blue key point “tail” is labeled at the fork of the tuna. The tuna labeling image is shown in Figure 2. The labeling results are stored in JSON standard format, and the stored information includes image path, width, and height dimensions, the number of channels, and the location information of the tuna labeling box and key points. The dataset is divided into a training set and a validation set according to the ratio of 8:2, and the final numbers of the training set and validation set are 748 and 187 images, respectively.



Figure 2. Tuna labeling image.

3. Tuna Fork Length Extraction and Weight Estimation Methods

3.1. YOLOv8-Pose Detection Algorithm

The YOLOv8 algorithm is a new SOTA model composed of the backbone, neck, and head, which has improved accuracy and flexibility compared to previous versions and can support a wide range of visual AI tasks. The network structure is shown in Figure 3.

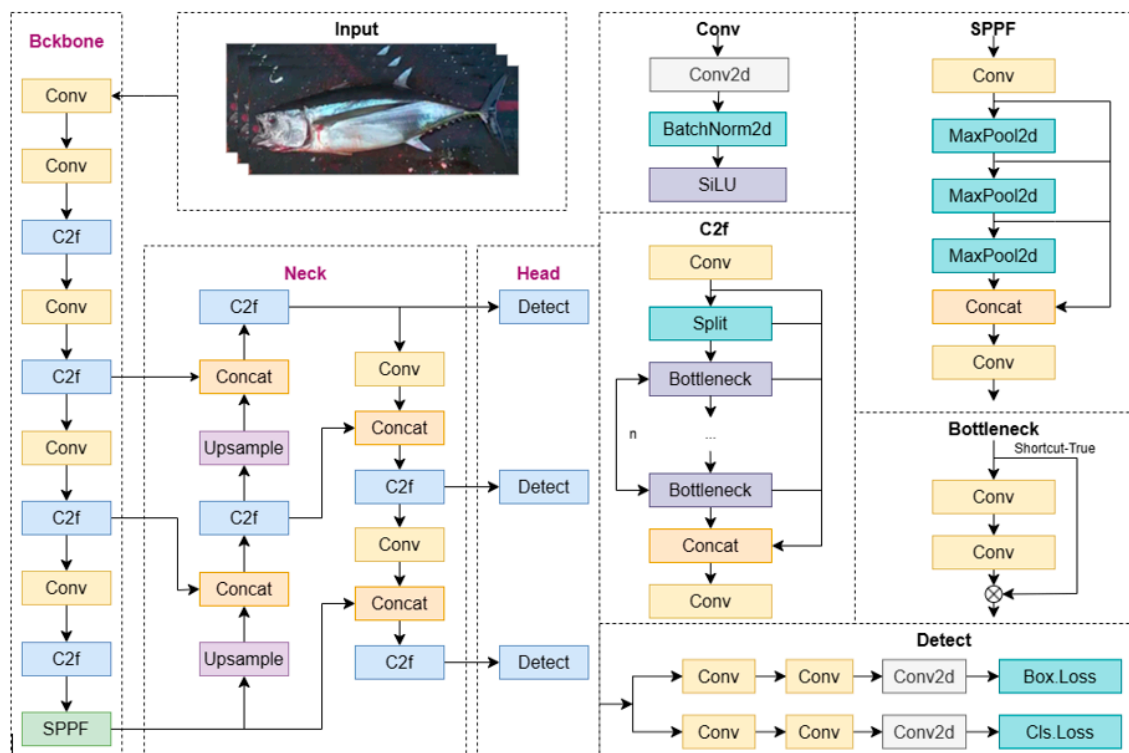


Figure 3. YOLOv8 network architecture diagram.

The backbone is mainly used for feature extraction, and YOLOv8 replaces the cross-stage local network (CSP) module in YOLOv5 with a lightweight C2f module, which enhances the feature representation through dense residual structure; reduces the computational complexity and the model capacity by changing the number of channels through splitting and splicing operations according to the scaling factor; and reduces the computational complexity and the model capacity by using a fast spatial pyramid pooling layer (SPPF) in the tail to increase the sensory field and capture feature information at different levels in the scene.

The neck part uses the path aggregation network (PAN) with the C2f module for the fusion of feature maps at different scales output from the three phases of the backbone, which helps to aggregate shallow information to deeper features. It enhances the ability of the network to fuse features of objects with different scaling scales, which is used to fuse different levels of features and propagate feature information to improve the performance of model detection. The C2f module is a key module in YOLOv8, which is used to fuse low-resolution feature maps with high-resolution feature maps to improve the accuracy of target detection. The head part uses the decoupled header structure, which is divided into the classification and localization prediction end, which is used to mitigate the conflict that exists between the classification and localization tasks. The anchor-free framework is used to improve the detection performance, which is more advantageous in detecting targets with irregular lengths and widths.

YOLOv8-Pose is a deep convolutional network based on YOLOv8, which adds the estimation of human pose key point locations, along with target detection [17], which is mainly used for human body detection and pose estimation. The target task of YOLOv8-Pose is to perform the prediction of image key points based on the completion of target detection. To better train the model by calculating the location of key points, the network uses a key point loss computation module for training the model with key point detection capability and optimizes the parameters of the model by weighted loss.

3.2. Improved YOLOv8-Pose Detection Method

To address the issues of poor detection speed, low detection accuracy, false positives, and critical point misses in maritime fishing environments due to ship movement, lighting, and backdrop influences on tuna bodies, a tuna key point detection algorithm combining the lightweight RepGhost module and the weighted bidirectional feature pyramid network BIFPN constructed using heavy parameterization technology was proposed based on the YOLOv8-Pose network model. The specific improvements are as follows:

- Replace the original C2f with C2f_repghost in layers 2, 4, 6, and 8 of the backbone network, and utilize the cascade operator to cheaply maintain a large number of channels by reusing feature maps from other layers to improve the inference speed and reduce the number of parameters.
- The weighted bidirectional feature pyramid network BIFPN is used to adaptively fuse features of different scales and transfer context information. Considering that the original BiFPN model uses the summation operation for feature fusion, which makes it easy to lose some detailed features, we use the weighted splicing operation, denoted as “BiFPN_Cat2”, to increase the feature granularity by combining the channel information of the tensor to improve the accuracy of multi-scale target detection and improve the drifting of key point detection results.

The improved network structure is shown in Figure 4.

3.2.1. Lightweight RepGhost Module

The RepGhost module is an improved version based on the Ghost module, which introduces reparameterization technology based on the Ghost module, effectively reducing the computational complexity of the model [18]. The module consists of two key components, including the Ghost module and the reparameterization operation. The Ghost module mainly contains three sub-modules, including 1×1 convolution, 3×3 convolution, and

1×1 convolution. It reduces the number of input channels using 1×1 convolution, then uses 3×3 convolution for feature extraction, and finally restores the number of channels using 1×1 convolution. The design of the Ghost module can effectively reduce the number of parameters and improve the performance of the model. The specific steps of the fusion of RepGhost and C2f are shown in Figure 5.

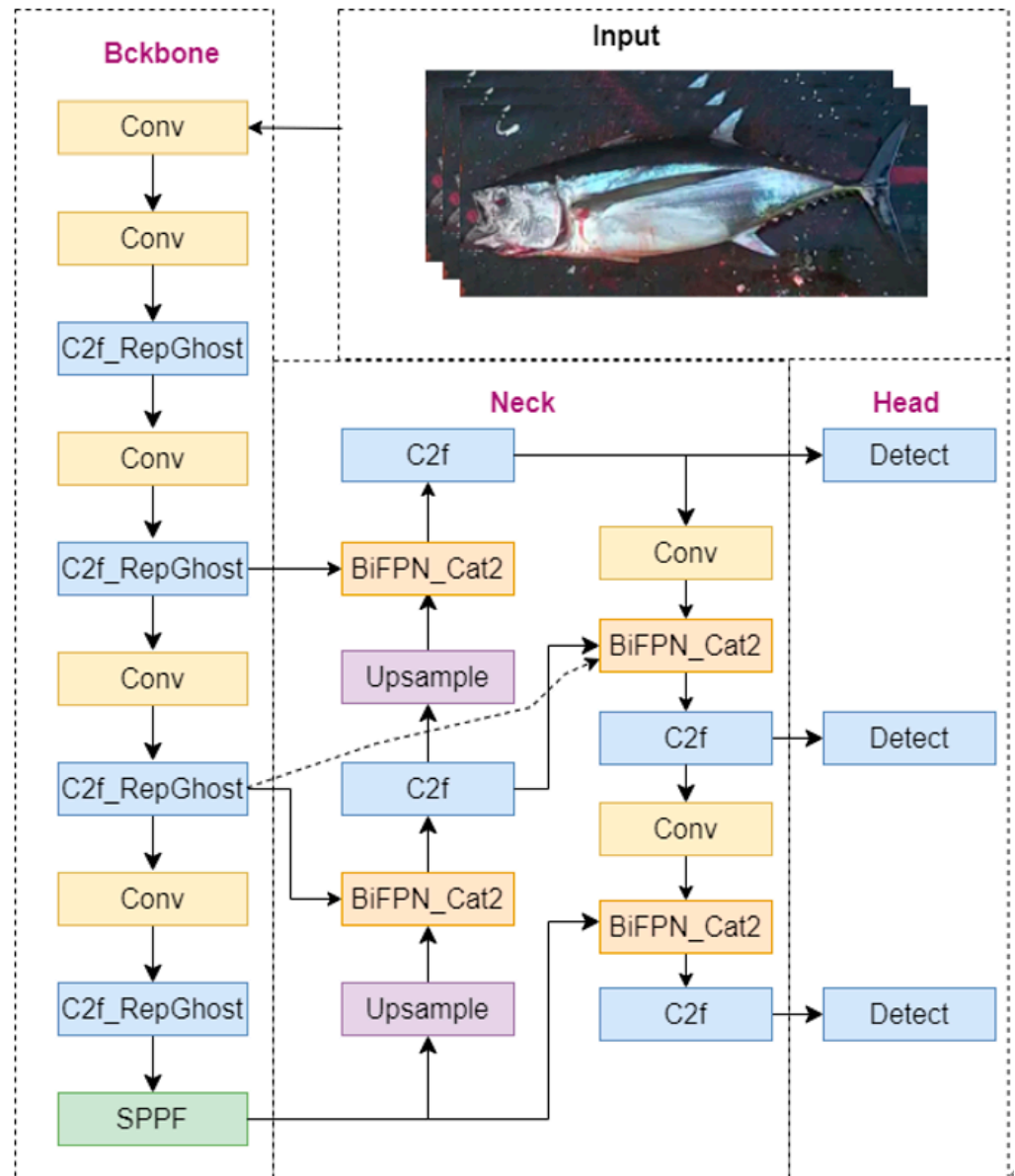


Figure 4. Improved YOLOv8 network structure diagram.

Firstly, replace the Concat operation module of the original module with the add operation. In the principle of structural reparameterization, the Relu module is moved backward after the add operation, and then the normalization module BN is added to the reparameterized structural branch. This operation can be quickly inferred during training. Finally, a lightweight and heavily parameterized module called RepGhost is generated, which only contains convolutional layers and Relu.

The Ghost module proposes to generate more feature maps from inexpensive operations, so the capacity of the model can be expanded cost-effectively. The RepGhost module proposes a more efficient way to generate and fuse different feature maps through reparameterization. Unlike the Ghost module, the RepGhost module removes the inf-

efficient Concat operation, which saves a lot of inference time. Moreover, the information fusion process is implicitly performed using the add operation instead of leaving it to other convolutional layers.

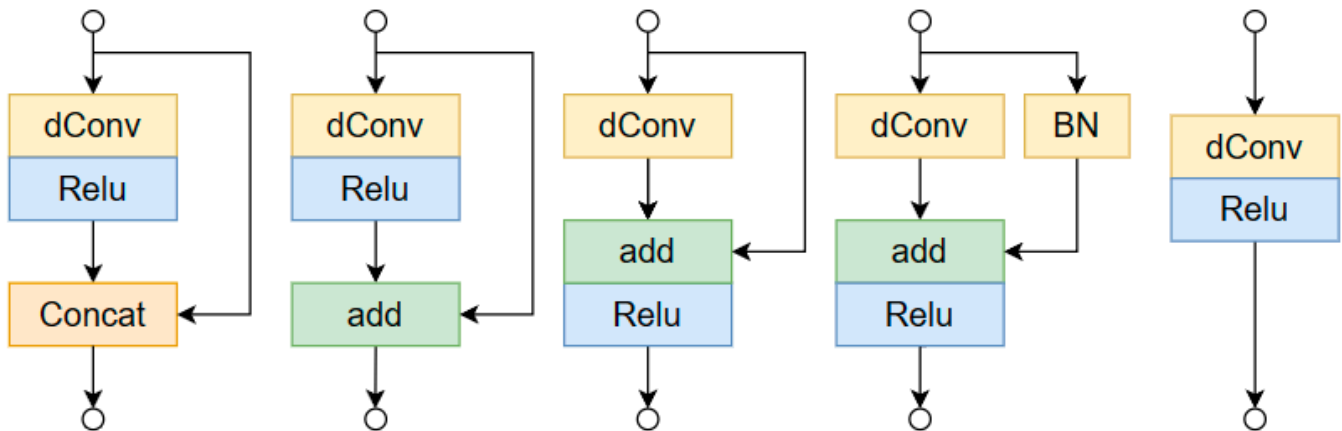


Figure 5. RepGhost and C2f fusion steps.

3.2.2. Weighted Bidirectional Feature Pyramid Network BiFPN

BiFPN is an improved feature fusion model based on PANet, mainly used for target detection tasks. The structure is weighted and bi-directionally connected, i.e., top-down and bottom-up structures, and cross-scale connectivity is achieved by constructing bi-directional channels, which directly fuse the features in the feature extraction network with the relative sized features in the bottom-up paths, retaining the shallower semantic information without losing too much deep semantic information [19]. Its structure is shown in Figure 6.

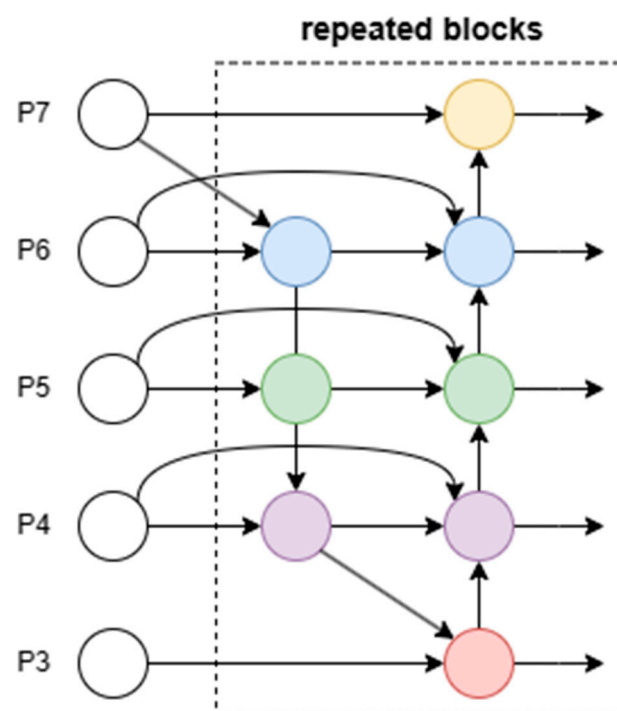


Figure 6. Structure of BiFPN.

The original BiFPN network fuses features from layers P3 to P7 and removes the nodes in layers P3 and P7 that have only one input path and are not involved in feature fusion to simplify the PANet model. At the same time, the information is transferred between

the same layers through “shortcut” connections, so that the original image features can be fused from the bottom upward. BiFPN adopts cross-scale weighted connections, given the differentiation of the contribution of different feature layers to feature fusion, according to the size of their contribution to the learning of the corresponding weights assigned to learn the weight parameter after the fast normalization weighting parameters and then fuses the features using a fast normalization method [20]. Usually, the deeper semantic features are richer than the shallow ones; too much retention of the shallow semantic information will lead to serious loss of the deeper semantic information, so the deeper layers are given higher weights. Considering that only layers P3–P5 of the YOLO model are involved in feature fusion and the visibility of smaller-sized targets is low, this paper simplifies the BiFPN, and only layers P3–P5 are involved in the weighted feature fusion while retaining the “shortcut” connection in layer P4. The structure is shown in Figure 7.

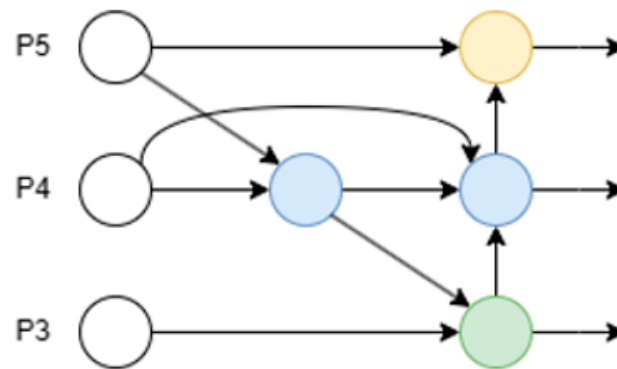


Figure 7. Simplified structure of BiFPN.

3.3. Tuna Fork Length Extraction Method

Tuna fork length is the length of the fish’s body from the end of the kiss to the deepest point of the tail fork, Ahmad suggested that the size of fish is usually more related to growth, as some ecological and physiological factors are more dependent on size calculations. The growth pattern of bigeye tuna (*Thunnus obesus*) landed in Benoa Harbor is equidistant, with an increase in length equal to an increase in weight [21]. Because bigeye tuna (*Thunnus obesus*) and albacore tuna (*Thunnus alalunga*) are both *Thunnus*, the weight of albacore tuna (*Thunnus alalunga*) can also be estimated based on information related to its body length. Taking into account the complexity of the ship’s operating environment, the image acquisition method needs to be simple and stable, so it is not suitable to use a depth camera to measure the tuna’s body height. Therefore, an ordinary camera was used to collect two-dimensional fork length information on tuna.

The natural death of the tuna in the deep sea and the treatment of catching up will lead to the tuna’s mouth being open, and the upper and lower kiss spacing is larger. Through the detection of the three key points to obtain the relative position of the fish’s upper and lower kisses, take the midpoint of the two to obtain the theoretical position of the closure (x_1, y_1), and with the tail key point (x_2, y_2) linkage to obtain the albacore tuna’s (*Thunnus alalunga*) fork length from the pixel distance based on the known pixel to actual distance scale factor, the actual distance between two points is obtained. The fork length extraction method is shown in Figure 8.

3.4. Tuna Weight Estimation Model

From the end of winter to the beginning of summer in 2003, Perçin, F. [22] used a 1500 m purse seine at a depth of 200 m in the Gulf of Antalya (Levantine Sea) to open the abdomens of 363 bluefin tuna (*Thunnus thynnus*) and separate them by sex. Through the log transformation of two parameters, a least squares linear regression estimation was

performed to establish a fitting equation for the length–weight relationship of bluefin tuna. The formula is expressed as follows:

$$y = ax^b \quad (1)$$

where y is the predicted weight of the target tuna in kg; a is the weight fitting coefficient 1; x is the fork length of the target tuna in cm; and b is the weight fitting coefficient 2.

In this study, the empirical formula for power function fitting described above was used to fit the least squares method to the fork length and weight sample data collected from 318 albacore tunas (*Thunnus alalunga*). The model first selected Equation (1), which defines its residual as the difference between each observation and the predicted value by the model, and the formula for the residuals of the i -th observation is e_i as follows:

$$e_i = y_i - ax_i^b \quad (2)$$

The residual sum of squares $L(a, b)$ is defined as follows:

$$L(a, b) = \sum_{i=1}^n (y_i - ax_i^b)^2 \quad (3)$$

where n is the number of observations, and the goal is to find the parameters a and b that minimize $L(a, b)$.

To minimize this residual sum of squares, the gradient descent method can be used to calculate the loss function $L(a, b)$ for the parameters a and b and then adjust the parameter values through iteration to gradually reduce the loss function. Meanwhile, because the data collected in the actual environment will be affected by the ocean wind and waves and the impact of the ship's rocking to produce corresponding errors, the variance between the fitted value and the actual value can be calculated to determine the degree of positive correlation to determine the degree of positive correlation, which is calculated as follows:

Total sum of squares:

$$TTS = \sum_i (y_i - \bar{y})^2 \quad (4)$$

Residual sum of squares:

$$RSS = \sum_i (y_i - f(x_i))^2 \quad (5)$$

Variance:

$$R^2 = 1 - \frac{RSS}{TTS}, \quad (6)$$

where y_i is each observation, and \bar{y} is the mean of the observations.

The final fitted equation is obtained as follows:

$$y = 1.38 \times 10^{-4} x^{2.5614}, R^2 = 0.8433 \quad (7)$$

To make the weight prediction model more accurate and stable, the data with excessive errors due to the collection process were excluded, and the exclusion threshold was set to the points at which the absolute value of the residuals was greater than twice the standard deviation. Standard deviation σ is defined as follows:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (y_i - \bar{y})^2}{n}} \quad (8)$$

where n is the number of observations; in this case, it is 318.

The final fitting equation after removing outliers is as follows:

$$y = 9.25 \times 10^{-5} x^{2.6488}, R^2 = 0.9287 \quad (9)$$

The initial fitting equation and the fitting equation after removing outliers are shown in Figure 9.

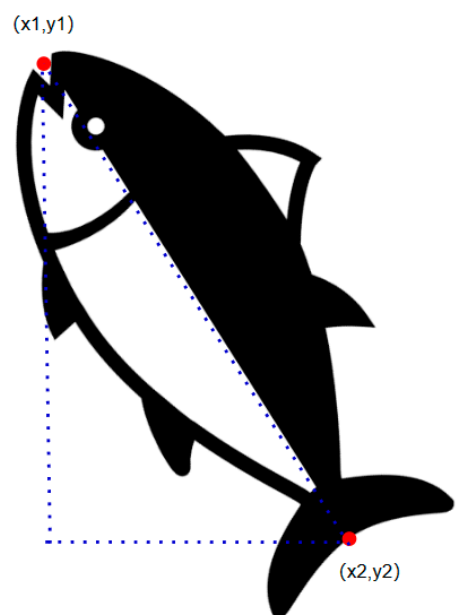


Figure 8. Method for extracting albacore tuna (*Thunnus alalunga*) fork length.

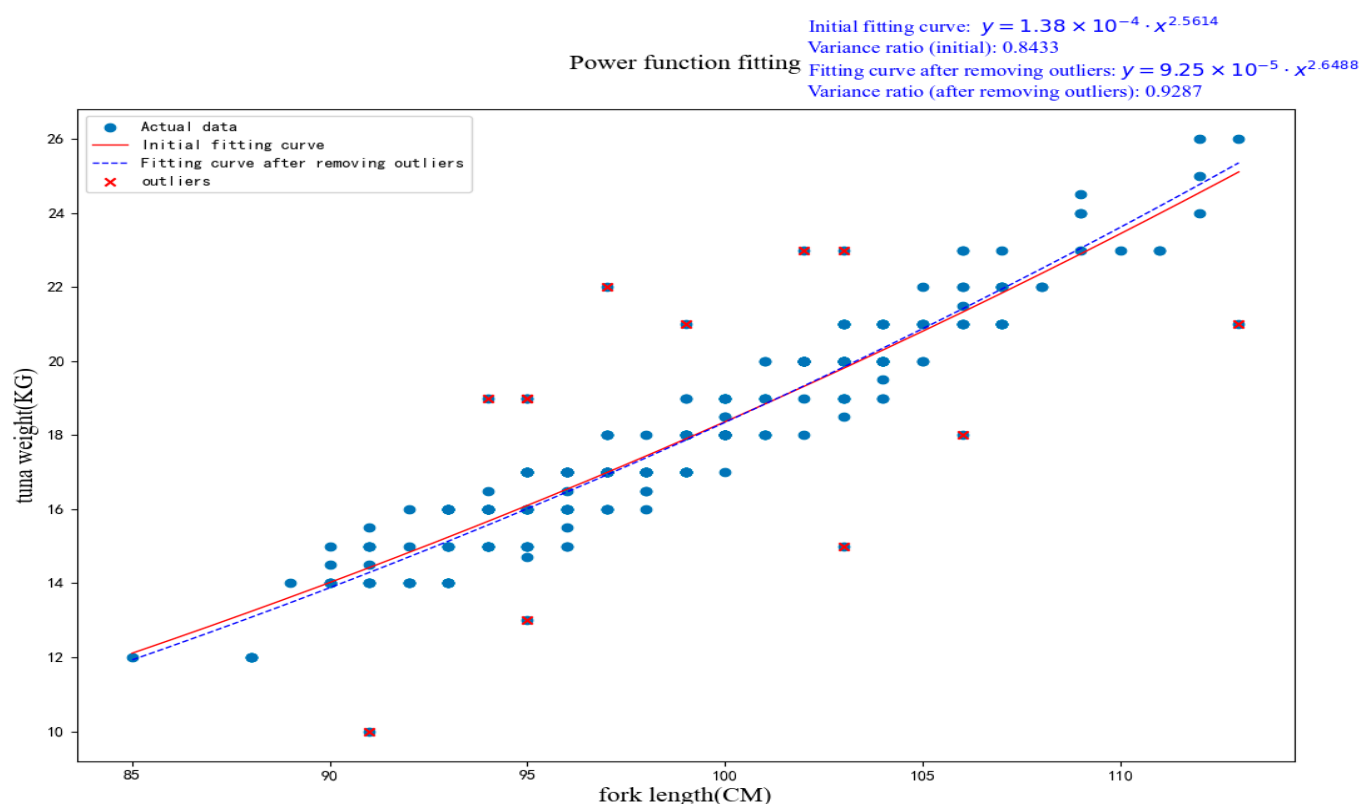


Figure 9. Albacore tuna (*Thunnus alalunga*) weight fitting curves and equations.

4. Results and Discussion

4.1. Training Environment and Parameter Settings

The test platform of this study is based on Windows 10 with the 64-bit operating system, 12th Gen Intel(R) Core(TM) i5-12400, 2.50 GHz processor, and NVIDIA GeForce RTX 2060 graphics card with 6 GB of video memory size. The GPU acceleration library is

CUDA11.8, the Python version is 3.8.18, and the deep learning framework is PyTorch 2.0.1. The model training parameter epoch is set to 300 rounds, the optimizer uses stochastic gradient descent, and the training adopts a warm-up strategy, using a learning rate of 0.0005 within the first three rounds to warm up the training, and then reverting to the initial learning rate after that, with the batch size set to 16, the initial learning rate set to 0.01, and the momentum to 0.937. All experiments use the pre-trained weights obtained from training on the COCO dataset for migration learning.

4.2. Evaluating Indicator

The experiment uses the average accuracy of the validation criteria based on the object key point similarity L_{oks} (object key point similarity) officially provided by MS COCO as the evaluation indicator [17]. In this case is L_{oks} defined as follows:

$$L_{oks} = \frac{\sum \left[\exp\left(-\frac{d_i^2}{2s^2k_i^2}\right) \delta(v_i > 0) \right]}{\sum \delta(v_i > 0)} \quad (10)$$

where i is the labeled key point number; d_i^2 is the square of the Euclidean distance between the detected key point and the true key point location; s^2 denotes the scale factor of the current fish target; k_i is the key point category used to control i of the attenuation constant; δ is the impulse function, indicating that only the visible key points in the true annotation are calculated for the L_{oks} value; and v_i is the value of the first i visibility of the first key point ($v_i > 0$ indicates that the key point is visible).

In terms of target detection, this study uses precision and recall as evaluation metrics. In terms of key point detection, $mAP_{0.5}$ and $mAP_{0.5:0.95}$ are used as the evaluation index, where $mAP_{0.5}$ represents the detection accuracy when the L_{oks} threshold is 0.5, and $mAP_{0.5:0.95}$ represents the average detection accuracy when the L_{oks} thresholds are 0.50, 0.55, ..., 0.90, 0.95. The test speed of a single image is used as an index for evaluating the inference speed of the model, while the number of parameters is used as an index for evaluating the model size.

4.3. Baseline Model Comparison Experiment

To balance the relationship between model accuracy and computational cost, YOLOv8n-Pose, YOLOv8s-Pose, YOLOv8m-Pose, YOLOv8-Pose, and YOLOv8x-Pose models are trained on the albacore tuna (*Thunnus alalunga*) dataset. The performance of the models is evaluated by comparing the $mAP_{0.5}$, $mAP_{0.5:0.95}$, giga floating-point operations per second (GFLOPS), and parameters for albacore tuna (*Thunnus alalunga*) recognition.

From the Table 1 analysis, it can be concluded that the YOLOv8n pose model has high accuracy in fish detection tasks, can effectively identify fish targets, and has relatively low computational and parameter requirements. Considering the practical application of this algorithm in the environment, YOLOv8n-Pose is chosen as the baseline model. It can infer more quickly without losing prediction accuracy and run efficiently on devices with limited resources.

Table 1. Baseline model comparison experiment results.

Experiment	$mAP_{0.5}/\%$	$mAP_{0.5:0.95}/\%$	$mAP_{0.5}kp/\%$	$mAP_{0.5:0.95}kp/\%$	Params/M	GFLOPS
YOLOv8n-Pose	99.4	87.5	99.4	99.3	3.08	8.444
YOLOv8s-Pose	99.5	86.7	99.4	99.2	11.42	29.637
YOLOv8m-Pose	99.5	87.8	99.4	99.4	26.41	81.219
YOLOv8l-Pose	99.4	88.1	99.3	99.3	44.48	169.108
YOLOv8x-Pose	99.5	88.3	99.5	99.5	69.48	263.905

4.4. Ablation Experiment

The experimental results in Table 2 show that the improved YOLOv8-Pose model is more effective than the original mode in terms of $mAP_{0.5}$, $mAP_{0.5:0.95}$, $mAP_{0.5kp}$, and $mAP_{0.5:0.95kp}$. This indicates that the improved YOLOv8-Pose model also has better performance in target detection and key point detection and localization, and at the same time, the model reduces the number of parameters and floating-point operations of model computation and greatly improves the inference speed, which reduces the number of parameters and floating-point operations by 13.63%, reduces the number of floating-point operations by 14.03%, and improves the inference speed by 374% compared with that of the original model. The inference speed is improved by 374%. At the same time, we compared the localization ability of albacore tuna (*Thunnus alalunga*) key point detection with the model before and after the improvement and found that the improved model can greatly reduce the drift of key points at the albacore tuna (*Thunnus alalunga*) head and the deepest part of the tail openings which are represented by red dots in the red circle. It reduces the loss of accuracy brought by detection errors, improves the accuracy of fork length extraction (as shown by the blue lines), which lays a solid foundation for the accurate prediction of tuna weight. The actual comparison effect of the key point detection of the model before and after improvement is shown in Figure 10. According to the observation of the actual effects before and after the improvement in Figure 10a–d, it can be seen that there is a certain drift deviation between the detection of key points in the mouth and tail of albacore tuna (*Thunnus alalunga*) before the improvement and the actual points, while the improved model can accurately locate in the actual position.

Table 2. Ablation test results.

Experiment	$mAP_{0.5}/\%$	$mAP_{0.5:0.95}/\%$	$mAP_{0.5kp}/\%$	$mAP_{0.5:0.95kp}/\%$	Params/M	GFLOPS	Speed/ms
YOLOv8-Pose	99.4	87.5	99.4	99.3	3.08	8.444	3.052
YOLOv8-Pose + C2f_RepGhost	99.2	85.5	99.2	99	2.66	7.259	2.845
YOLOv8-Pose + BIFPN	99.4	87.9	99.4	99.4	3.08	8.444	12.867
YOLOv8-Pose + C2f_RepGhost + BIFPN	99.5	87.1	99.2	99.2	2.66	7.259	14.475

4.5. Error Analysis

In the previous section, the detection results of the improved YOLOv8-Pose key point model were analyzed and proved to have a good performance on the albacore tuna (*Thunnus alalunga*) fish body key point localization task. In this section, weight prediction experiments based on the improved YOLOv8-Pose model are conducted to demonstrate the performance of this model in albacore tuna (*Thunnus alalunga*) weight prediction.

The weight prediction errors of the improved YOLOv8-Pose key point detection model were calculated for the same subjects and compared with those of the original YOLOv8-Pose model to visualize the difference between the two models for albacore tuna (*Thunnus alalunga*) weight prediction. The comparison of the measurement results between the two models is shown in Table 3.

To more intuitively demonstrate the effect of this paper's algorithm for albacore tuna (*Thunnus alalunga*) key point detection and weight prediction, the experimental results are randomly visualized and the experimental results are shown in Figure 11. Figure 11a,b corresponds to the 16th datapoint in the table, the error percentage of body weight predicted by the improved model is 1.71% lower than that of the pre-improved model, and Figure 11c,d corresponds to the 6th data in the table, and the error percentage of body weight predicted by the improved model is 4.17% lower than that of the pre-improved model. Therefore, the improved YOLOv8-Pose algorithm can more accurately realize the key point detection and weight estimation.

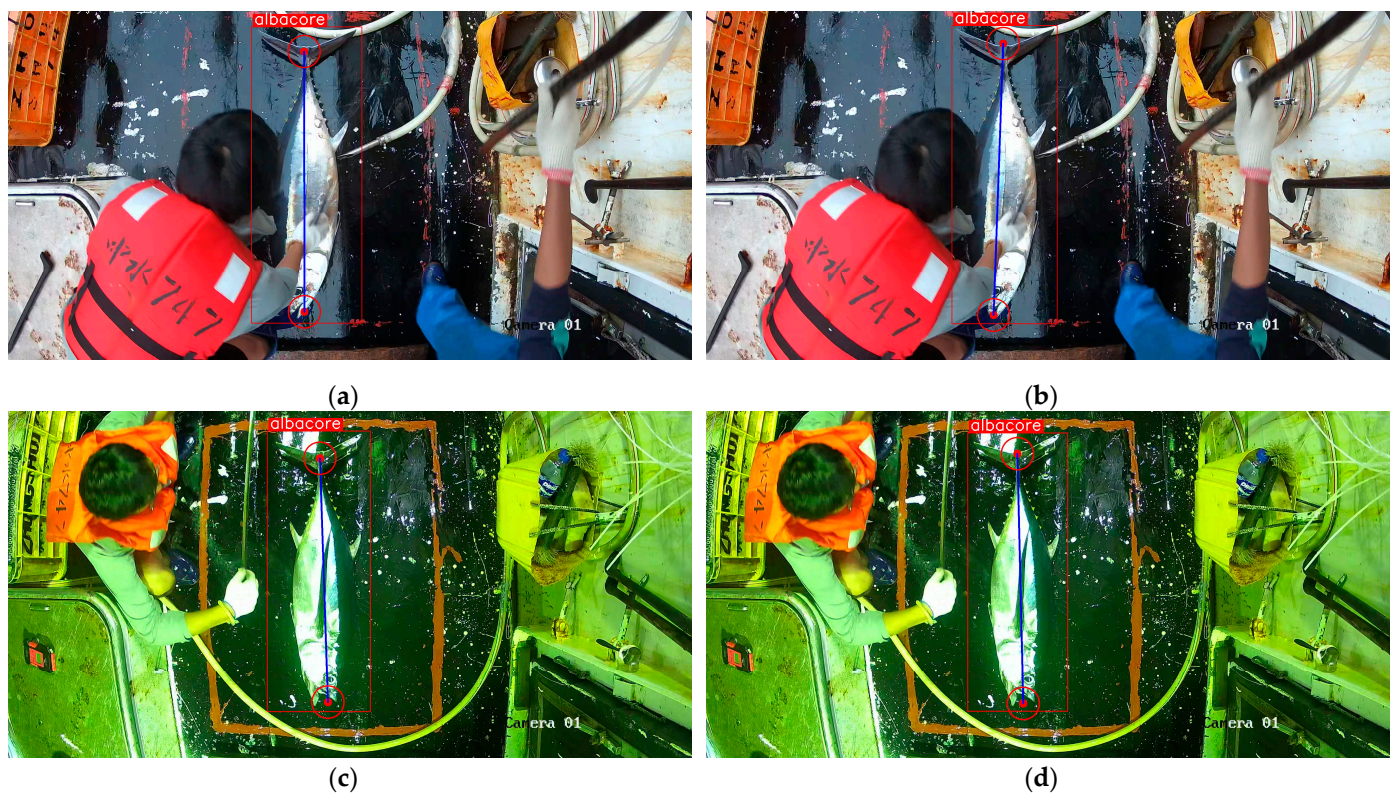


Figure 10. The actual comparison effect before and after the improvement. (a) Before improvements in the day; (b) after improvements in the day; (c) before improvements in the night; (d) after improvements in the night.

Table 3. Error before and after model improvement.

Serial Number	Actual Fork Length (cm)	Fork Length Extracted Improved Model (cm)	Actual Weight (kg)	Weight Estimated Improved Model (kg)	Prediction Error (Improved Model)	Prediction Error (Original Model)
1	100	97.96	18	17.37	3.51%	6.73%
2	102	102.40	20	19.65	1.77%	4.04%
3	95	96.15	17.5	17.67	0.97%	5.77%
4	93	93.69	16	15.34	4.10%	4.44%
5	101	104.51	19	20.79	9.43%	12.82%
6	96	96.17	17.5	17.71	1.20%	5.37%
7	101	104.30	21.5	20.67	3.85%	7.51%
8	98	101.36	19.5	19.10	2.07%	7.25%
9	104	101.92	20	19.39	3.05%	6.75%
10	97	96.30	17	16.56	2.57%	4.19%
11	109	112.89	24.5	25.75	5.13%	7.06%
12	98	97.76	17	17.27	1.58%	9.49%
13	103	103.75	20	20.37	1.85%	5.14%
14	102	100.01	17	18.39	8.21%	14.02%
15	107	105.99	22	21.62	1.73%	5.31%
16	101	98.51	17	17.64	3.75%	5.46%
17	106	104.94	21.5	21.03	2.18%	9.79%
18	96	99.75	19	18.27	3.86%	9.20%
19	104	102.35	20	19.62	1.90%	7.64%
20	101	99.63	18	18.20	1.13%	5.64%

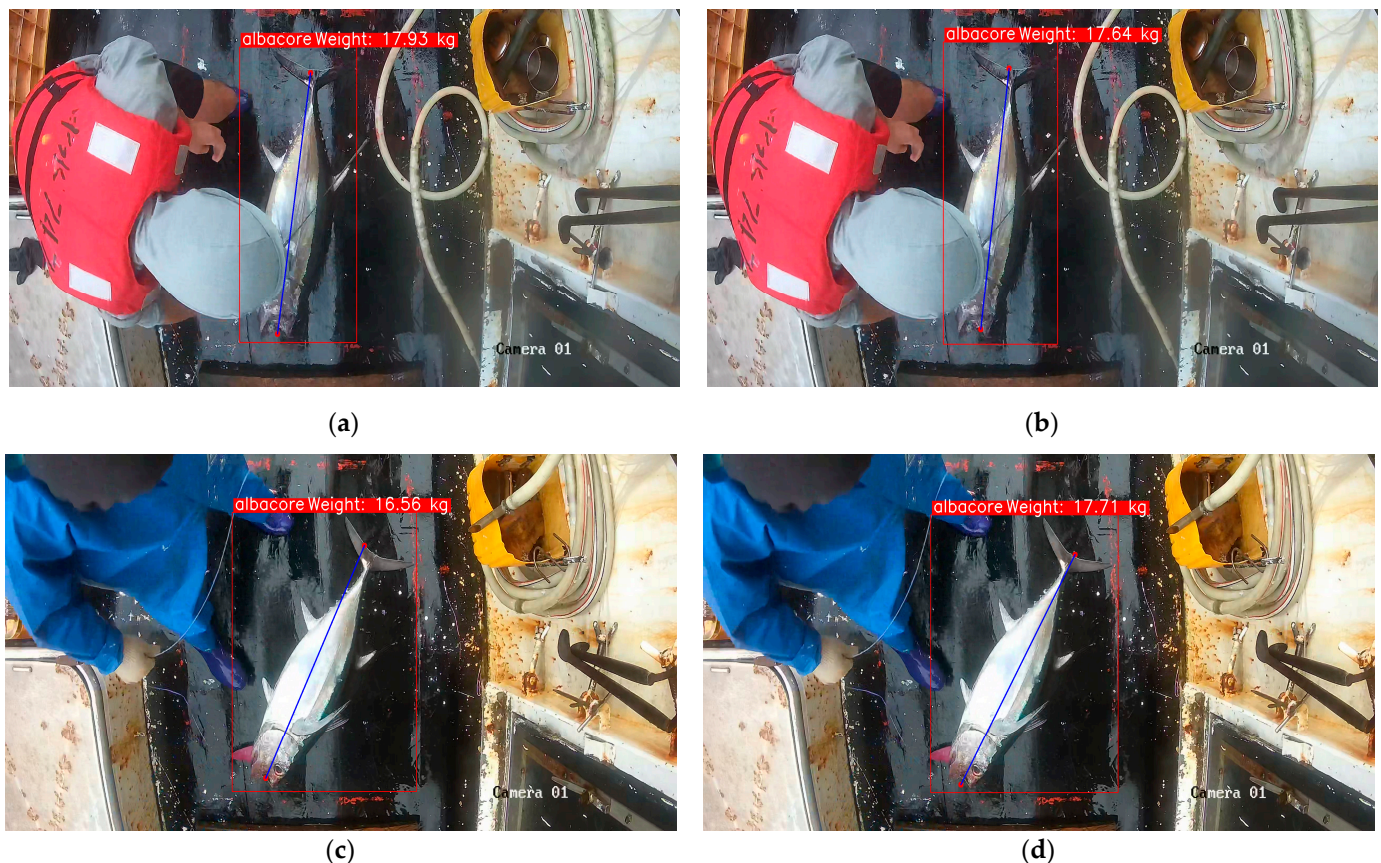


Figure 11. Visualization of experimental results. (a) Weight estimation before improvements; (b) weight estimation after improvements; (c) weight estimation before improvements; (d) weight estimation after improvements.

5. Conclusions

According to the problems of large statistical error and poor real-time performance of catch weight in the ocean fishing tuna industry, an algorithm based on improved YOLOv8-Pose for albacore tuna (*Thunnus alalunga*) fork length extraction and weight estimation is proposed concerning the human body posture estimation algorithm, which is verified by training and testing on a homemade albacore tuna (*Thunnus alalunga*) dataset. By introducing the RepGhost module in the backbone network part, the inefficient Concat operation is removed, which saves a lot of inference time. The weighted bidirectional feature pyramid network BIFPN is used to retain deeper semantic features, which effectively reduces the drift of keypoint detection. The results show that the improved algorithm reduces the number of parameters and computation of the model without sacrificing the detection accuracy and improves the inference speed of the model, and the error of comparing the predicted weight with the actual albacore tuna (*Thunnus alalunga*) weight is no more than 10%.

The key point detection model proposed in this paper can effectively realize the detection effect under the ocean fishing operation scenario, which can provide strong support and a certain theoretical basis for the management and decision-making of the ocean fishing tuna industry. However, the method in this paper also has shortcomings. The model in the night operation of the light reflection and the complexity of the situation between the different fish blockage will appear in the detection of the decline in accuracy and speed. Meanwhile, due to the limitations of the data sample, only the fork-length-weight equation for albacore tuna (*Thunnus alalunga*) was fitted analytically, and other different tunas will need to be studied in subsequent research to demonstrate the generalization of the model to different species. The following work can be aimed at this aspect under the premise

of guaranteeing the accuracy of detection in response to a variety of complex operating conditions and effectively realize the intelligent fishing statistics.

Author Contributions: Conceptualization, Y.L. and Z.F.; methodology, Z.F.; software, Z.F.; validation, Y.L. and Z.F.; formal analysis, Z.F.; investigation, L.S. and Y.L.; resources, Y.L. and B.L.; data curation, L.S.; writing—original draft preparation, Z.F.; writing—review and editing, Y.L.; visualization, Z.F.; supervision, Y.L. and H.S.; project administration, Y.L. and H.S.; funding acquisition, Y.L. and L.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Key R&D Program of China, grant number 2023YFD2401301 and National Key R&D Program of China, grant number 2023YFD2401305.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The authors would like to express their gratitude for the support of the Fishery Engineering and Equipment Innovation Team of Shanghai High-level Local University.

Conflicts of Interest: Hengshou Sui and Bin Li are employed by CNFC Overseas Fisheries Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Chatain, B.; Debas, L.; Bourdillon, A.J.A. A photographic larval fish counting technique: Comparison with other methods, statistical appraisal of the procedure and practical use. *Aquaculture* **1996**, *141*, 83–96. [\[CrossRef\]](#)
2. Rousseau, D.M. Is there such a thing as “evidence-based management”? *Acad. Manag. Rev.* **2006**, *31*, 256–269. [\[CrossRef\]](#)
3. Torres-Faurrieta, L.K.; Dreyfus-León, M.J.; Rivas, D. Recruitment forecasting of yellowfin tuna in the eastern Pacific Ocean with artificial neuronal networks. *Ecol. Inform.* **2016**, *36*, 106–113. [\[CrossRef\]](#)
4. Déniz, O.; Bueno, G.; Salido, J.; De la Torre, F.J. Face recognition using histograms of oriented gradients. *Pattern Recognit. Lett.* **2011**, *32*, 1598–1603. [\[CrossRef\]](#)
5. Srikantaswamy, R.; Samuel, R.S. A novel face segmentation algorithm from a video sequence for real-time face recognition. *EURASIP J. Adv. Signal Process.* **2007**, *2007*, 051648. [\[CrossRef\]](#)
6. Al Muksit, A.; Hasan, F.; Emon, M.F.H.B.; Haque, M.R.; Anwary, A.R.; Shatabda, S. YOLO-Fish: A robust fish detection model to detect fish in realistic underwater environment. *Ecol. Inform.* **2022**, *72*, 101847. [\[CrossRef\]](#)
7. 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.
8. Andriluka, M.; Pishchulin, L.; Gehler, P.; Schiele, B. 2d human pose estimation: New benchmark and state of the art analysis. In Proceedings of the IEEE Conference on computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 3686–3693.
9. Yang, S.; Quan, Z.; Nie, M.; Yang, W. Transpose: Keypoint localization via transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 11802–11812.
10. Flores-Fuentes, W.; Rodríguez-Quinones, J.C.; Hernandez-Balbuena, D.; Rivas-López, M.; Sergiyenko, O.; Gonzalez-Navarro, F.F.; Rivera-Castillo, J. Machine Vision supported by Artificial Intelligence Applied to Rotatory Mirror Scanners. In Proceedings of the 2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE), Istanbul, Turkey, 1–4 June 2014; pp. 1949–1954.
11. Strachan, N.J.C. Length measurement of fish by computer vision. *Comput. Electron. Agric.* **1993**, *8*, 93–104. [\[CrossRef\]](#)
12. Abdullah, N.; Shafry, M.; Rahim, M.; Amin, I.M. Measuring fish length from digital images (FiLeDI). In Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human, Seoul, Republic of Korea, 24–26 November 2009; pp. 38–43.
13. Garcia, R.; Prados, R.; Quintana, J.; Tempelaar, A.; Gracias, N.; Rosen, S.; Vågstøl, H.; Løvall, K.J.I. Automatic segmentation of fish using deep learning with application to fish size measurement. *ICES J. Mar. Sci.* **2020**, *77*, 1354–1366. [\[CrossRef\]](#)
14. Jeong, T.-Y.; Ha, I.-K.J. OpenPose based Smoking Gesture Recognition System using Artificial Neural Network. *Teh. Glas.* **2023**, *17*, 251–259. [\[CrossRef\]](#)
15. Yu, Y.; Zhang, H.; Yuan, F. Key point detection method for fish size measurement based on deep learning. *IET Image Process.* **2023**, *17*, 4142–4158. [\[CrossRef\]](#)
16. Suo, F.; Huang, K.; Ling, G.; Li, Y.; Xiang, J. Fish keypoints detection for ecology monitoring based on underwater visual intelligence. In Proceedings of the 2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV), Shenzhen, China, 13–15 December 2020; pp. 542–547.

17. Maji, D.; Nagori, S.; Mathew, M.; Poddar, D. Yolo-pose: Enhancing yolo for multi person pose estimation using object keypoint similarity loss. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 2637–2646.
18. Chen, C.; Guo, Z.; Zeng, H.; Xiong, P.; Dong, J. Repghost: A hardware-efficient ghost module via re-parameterization. *arXiv* **2022**, arXiv:2211.06088.
19. 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.
20. Ye, X.; Xiaohui, L.; Yannan, Q.; Tao, X.; Quanchun, Y.; Jian, P.; Saiké, J.; Xiaolan, L.J.S.A. Detection of pear inflorescence based on improved Ghost-Yolov5s-BiFPN algorithm. *Smart Agric.* **2022**, *4*, 108. [[CrossRef](#)]
21. Ahmad, F.; Dewanti, L.P.; Arnenda, G.L.; Rizal, A. Length-weight relationship and catch size of bigeye tuna (*Thunnus obesus*) landed in Benoa, Bali, Indonesia. *World News Nat. Sci.* **2019**, *23*, 34–42.
22. Perçin, F.; Akyol, O. Length-weight and length-length relationships of the bluefin tuna, *Thunnus thynnus* L., in the Turkish part of the eastern Mediterranean Sea. *J. Appl. Ichthyol.* **2009**, *25*, 782–784. [[CrossRef](#)]

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