Article

Feasibility Research on Fish Pose Estimation Based on Rotating Box Object Detection

Bin Lin ¹, Kailin Jiang ², Zhiqi Xu ¹, Feiyi Li ¹, Jiao Li ¹, Chaoli Mou ¹, Xinyao Gong ¹ and Xuliang Duan ^{1,*}

¹ College of Information Engineering, Sichuan Agricultural University, Ya'an 625000, China; linbin203279@gmail.com (B.L.); ameter.above.thesky@gmail.com (Z.X.); 202005483@stu.sicau.edu.cn (F.L.); 202005852@stu.sicau.edu.cn (J.L.); mouchaoli@stu.sicau.edu.cn (C.M.); 202005590@stu.sicau.edu.cn (X.G.)

² College of Science, Sichuan Agricultural University, Ya'an 625000, China; 202003978@stu.sicau.edu.cn

* Correspondence: duanxuliang@sicau.edu.cn; Tel.: +86-150-083-053-94

Abstract: A video-based method to quantify animal posture movement is a powerful way to analyze animal behavior. Both humans and fish can judge the physiological state through the skeleton framework. However, it is challenging for farmers to judge the breeding state in the complex underwater environment. Therefore, images can be transmitted by the underwater camera and monitored by a computer vision model. However, it lacks datasets in artificial intelligence and is unable to train deep neural networks. The main contributions of this paper include: (1) the world's first fish posture database is established. 10 key points of each fish are manually marked. The fish flock images were taken in the experimental tank and 1000 single fish images were separated from the fish flock. (2) A two-stage attitude estimation model is used to detect fish key points. The evaluation of the algorithm performance indicates the precision of detection reaches 90.61%, F1-score reaches 90%, and Fps also reaches 23.26. We made a preliminary exploration on the pose estimation of fish and provided a feasible idea for fish pose estimation.

Keywords: aquaculture automation; rotating box; fish detection; fish pose; computer vision

Citation: Lin, B.; Jiang, K.; Xu, Z.; Li, F.; Li, J.; Mou, C.; Gong, X.; Duan, X. Feasibility Research on Fish Pose Estimation Based on Rotating Box Object Detection. *Fishes* **2021**, *6*, 65. <https://doi.org/10.3390/fishes6040065>

Received: 24 October 2021

Accepted: 17 November 2021

Published: 19 November 2021

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

Copyright: © 2021 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

Fish usually have high nutritional value and can meet the needs of humans and other species. With the improvement of social levels, people put forward higher and higher requirements for the meat quality and taste of fish. To meet these high requirements, farmers need to accurately breed and monitor fish in real-time and accurately grasp the distribution, growth status, and behavioral characteristics of fish [1].

Due to the complex underwater environment, the adaptability of traditional and backward electronic equipment in water is very low, and even harmful substances may be produced, which interfere with the living environment of fish, affect their growth, change their physiological properties, and bring losses in breeding and sales [2]. Therefore, the realization of fishery intelligent detection by a computer vision method is the inevitable trend of the development of the fishery breeding industry chain in modern society. Object detection and pose estimation are important supporting technologies for fish distribution and condition observation and measurement [3]. Both object detection and pose estimation belong to the basic tasks of machine vision. The former is used to detect whether there are target objects of a given category in a given image, and the latter is used to predict the pose of the target object (human or animal) in the input image [4]. As a branch technology of computer vision and image processing, object detection is used to detect specific semantic objects (such as people, buildings, or cars) in digital images and videos. It has broad application prospects in the fields of video security, automatic driving, traffic monitoring, UAV scene analysis, and robot vision [5–7]. With the development of artificial intelligence, deep learning is becoming more and more popular in the field of target detection. At present, the mainstream target detection methods are mainly divided into

two-stage detection methods and one-stage detection methods [8]. Fast RCNN [9], Faster RCNN [10] and RefineNet [11] are classic two-stage detection methods. You Only Look Once [12–14], Single Shot MultiBox Detector (SSD) [15], RetinaNet [16], etc. are typical one-stage detection methods. Human pose estimation is widely used in human–computer interaction, behavior recognition, virtual reality, augmented reality, medical diagnosis, and other fields. In the field of human–computer interaction, human pose estimation technology accurately captures the details of human actions and can conduct contactless interaction with computers after obtaining human actions [17]. At present, there are two mainstream ideas in the field of pose estimation, that is, bottom-up or top-down methods, that are used to solve the task of pose estimation [17]. Due to the particularity of underwater object detection tasks, most of the existing detection algorithms rely on the gray information of the image. Olmos and Trucco [18] proposed an object detection method based on an unconstrained underwater fish video, which uses image gray and contour information to complete object detection, but the detection speed is slow. Zhang Mingjun et al. [19] proposed an underwater object detection method based on moment invariants, which uses the minimum cross-entropy to determine the threshold, which can ensure the integrity of gray information and uses gray gradient moment invariants to realize underwater image object detection. It has good robustness and high recall, but the accuracy still does not meet the expected requirements. Li, X. et al. [20] explained that underwater images may be of poor quality due to light scattering, color change, and shooting equipment conditions. Therefore, they applied Fast R-CNN [9] to fish object detection in a complex underwater environment. Xu, C. et al. [21] considered that an articulated object can be regarded as a manifold with point uncertainty, and proposed a unified paradigm based on Lie group theory to solve the recognition and attitude estimation of articulated targets including fish. The results show that their method exceeds the two baseline models of convolution neural network and regression forest. However, their method cannot be extended to datasets with more complex fish categories and postures and worse environmental quality (such as our golden crucian carp dataset). Xu, W. et al. [22] pointed out that underwater images are faced with difficulties such as low contrast, floating vegetation interference, and low visibility caused by water turbidity. They trained Yolo 3 with three different underwater fish datasets and deployed the model to a new dataset for testing. They found that the generalization ability of the model is not high. This also shows the challenge of the underwater environment to a certain extent. Knausgard et al. [23–25] combined the two tasks of fish detection and fish classification and proposed a phased in-depth learning method for the detection and classification of tropical fish: in the first stage, Yolo 3 was used to detect fish bodies, and in the second stage, CNN-SENet was used to classify the detection results of the previous stage. Our work is similar to this, but we use phased rotating box object detection and pose estimation, and the output is the integration of the results of the two stages. These works have not organically combined the mature object detection model and human pose estimation model in the current deep learning method and applied them to fisheries. Our work is committed to filling this gap.

However, the construction of an intelligent aquaculture system has been challenged and hindered to some extent. Firstly, the complex underwater natural environment such as the growth of algae and uneven distribution of light has caused some obstacles to the collection of visual data of aquatic animals [26]. Secondly, attitude estimation usually takes humans and vehicles with limited attitude changes as the target objects [27,28]; Although aquatic animals have no limb movement, their movement in the water is more open, can flip freely, and is not restricted by angle. The role of common data annotation becomes extremely limited.

To meet the above challenges, we use multi-object detection and animal pose estimation, real-time monitoring, early warning, and recording effective information to minimize the loss. In this regard, the aquatic animal we mainly study is the golden crucian carp. Based on its inherent advantages, this species plays a more distinctive role:

- (1) The physiological structure of golden crucian carp is relatively simple, there are no complex human-like joints and a high degree of freedom limbs, and the purposeful grass goldfish has high attitude recognition. Such as spawning, eating, skin infection, etc.
- (2) Although the body appearance similarity of golden crucian carp is high, the dataset based on artificial annotation was screened and analyzed, and the source is reliable, which is explained in detail in Sections 2.1 and 2.2.
- (3) The ecological fish tank with a high reduction degree has a high simulation of the aquaculture environment. In contrast, it is more in line with the requirements of the aquaculture industry chain, has no redundant interference, and can be freely captured from all perspectives.
- (4) Golden crucian carp can realize free movement in three-dimensional space in the aquatic environment. According to Figure 1, the turnover range of golden crucian carp is between $[0^\circ \sim 180^\circ]$. Generally, the deformation degree is large. As shown in Figure 2, 80% of the angle changes are above 40 degrees. Therefore, the traditional object detection pre-selection box is abandoned, and the rotating box is used for flexible box selection. This is the innovation of the dataset in our research process.

Figure 1. Analysis of crucian carp dataset. This figure is a heat map of the x, y, and width, height of the crucian carp image. The darker the color, the stronger the concentration, and the denser the distribution of crucian carp.

Figure 2. Analysis of crucian carp dataset. The angle distribution histogram in Figure 2 can be regarded as a normal distribution diagram on the whole, which counts the amount of grass gold at various angles, reflecting the randomness of the angle.

To sum up, our research is divided into two stages. The first stage is to take the multi-object detection image independent based on the rotating box as the basic input and

send it into the detection model based on Yolo 5 [29]. In the second stage, the pose of each golden crucian carp is detected separately to obtain the prediction subgraph, and then the output is superimposed and integrated to obtain the original picture. The new method can also be extended to other species except for aquatic animals and has strong ductility.

In short, our main contributions are:

- (1) The first dataset, we established a new large-scale golden crucian carp dataset; It contains 1541 pose estimation images from 10 golden crucian carp.
- (2) The recognition features are extracted from the database, and the related recognition algorithm based on computer vision is realized to recognize the golden crucian carp.
- (3) A comprehensive baseline is constructed, including golden crucian carp rotating box object detection and golden crucian carp pose estimation, to realize multi-object pose estimation.

2. Materials and Methods

2.1. Acquisition of Materials

Crucian carp have strong adaptability, have wide feeding habits, and are easy to raise. Wild Crucian carp are mainly distributed in Hangzhou and Jiaxing, China. It is more difficult to capture images, and its number is relatively rare compared with artificial rearing, so it has no sampling value. Therefore, the focus of this sampling is on artificially raised Crucian carp. We keep the fish in the fish tank and use the DJI pocket2 camera to capture and shoot from different angles and distances. The shooting time of the image is random, day and night, bright light, dark light environment are involved; the shooting angle is variable, including the change of shooting angle of the fish tank and the difference of shooting distance. These can ensure that the collected images cover more situations and enhance the adaptability of subsequent models to various environments. Using the above-mentioned sampling method, we captured more than thousands of images, but some of the images were discarded due to the occlusion of aquatic plants, turbid water, and failure to capture Crucian carp.

In the end, our dataset consists of 1541 images from 10 Crucian carp. Each fish has a corresponding label for multi-target detection and the number of images for each fish is also different. On average, each crucian carp has 1541 images in the dataset. Figures 1 and 2 are the analysis of the crucian carp dataset.

As shown in Figure 1, our description of the x, y, width, and height of the image relative to the original image's coordinate position and the width-to-height ratio all present a normal distribution. This shows that the distribution of crucian carp is concentrated and random on the whole; In the posture, most of the grass gold is free to tilt; There is a certain angle compared to the horizontal. As shown in Figure 2, the angle normal distribution histogram counts the amount of grass gold at various angles. It shows that only a few grass golds are in a horizontal posture, and most of the grass golds are in an oblique posture, and the angle is very random.

These images were annotated by 10 annotators under the guidance of professionals. The annotation process is divided into 3 stages. The first stage is to check the frame selection of the target. Using labeling and using the rotating frame to select the target frame of the crucian carp. For crucian carp showing one side, the dotted line in the rotating frame is aligned with the upper part of the dorsal fin; For the crucian carp showing the back or abdomen, the rotating frame is aligned with the back or abdomen. Crucian carp with different postures use the rotating frame flexibly according to the actual situation, as shown in Figure 3. Finally, transform the rotating frame with an angle to obtain a horizontal sub-image. In the second stage, the subgraph obtained in the first stage is used to annotate the fish mouth, fish eyes, front and rear ends of the dorsal fin, 4–8 points on the fishtail, and ventral fins in sequence with 10 key points of sequence numbers 0–9. Annotate the fish eyes, the front and rear ends of the dorsal fin, the 4–8 points on the fishtail, and the pelvic fin. The key points that do not appear in the image but exist are uniformly annotated in the upper left corner (coordinates (0,0)). In the subsequent training process, it will not

cause any influence on the model. If most of the fish body is visible, only some key points are blocked. We also estimate the occluded points and then label them to enhance the effect of the model in the case of a small amount of occlusion by the fish body. In the third stage, the bounding box repositions the crucian carp. The adjusted bounding box and posture keypoint annotations are used for network training. When training the network, the cropped image and the detected image are the input and target output.

Figure 3. Definition of key points of crucian carp (0–9).

2.2. Data Preprocessing

Excessive noise may confuse valid information. The crucian carp cultivated in the ecological fish tank is affected by the sound of the oxygen supply, the water quality care agent, and other factors, causing the collected images to always be noisy [30]. There are also noises in the encoding, transmission, and processing of these images. To improve the robustness and accuracy of the algorithm, the image effect tends to be in the natural breeding environment of crucian carp. We first detect and adjust the sharpness, color shift, and brightness difference of the image. At the same time, the newly proposed ACP item [31] is incorporated into the optimized method. As shown in Figure 4, the deployment strategy is used to provide information for the design of the deep network and comprehensively de-noise.

Figure 4. The general framework of the data preprocessing process. De-noise adjustments such as brightness, color shift, and sharpness of the image data first. The subsequent first-level denoising sub-image input incorporates ACP into the optimized new deep neural denoising architecture. It consists of a feature domain module, a reconstruction module, and k iteration stages based on a nonlinear operation (NLO) subnet and a dual attention mechanism (DEAM) module. The modules marked with * and # mean that the parameters of these modules are shared.

2.2.1. Training Data Enhancement

After preprocessing, we choose to perform a round of data enhancement before sending the dataset to the rotating object detector. We used a variety of data enhancement methods such as HSV color space enhancement, mosaic processing, image fusion, four-way flip, random scale transformation, etc.

2.2.2. HSV Color Space Enhancement

The crucian carp visual data we collected are all RGB images, and the RGB color space is represented by the combination of the linear components of the three colors of red, green, and blue. However, the HSV color space is more suitable for human observation. Therefore, we first scale the R, G, and B components of the crucian carp dataset to within the range of 0–1 and according to the following formula, the three components are converted into HSV components to obtain an HSV image. In this way, the image features can be expressed more intuitively, and the effect is enhanced.

$$S = \begin{cases} \frac{V - \min(R, G, B)}{V} & \text{if } V \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$H = \begin{cases} 60(G - B) / (V - \min(R, G, B)) & \text{if } V = R \\ 120 + 60(B - R) / (V - \min(R, G, B)) & \text{if } V = G \\ 240 + 60(R - B) / (V - \min(R, G, B)) & \text{if } V = B \\ 0 & \text{if } R = G = B \end{cases} \quad (1)$$

2.2.3. Mosaic

First, divide the crucian carp dataset into groups, and randomly take out 4 pictures in each group, perform random scaling, random inversion, random distribution, etc., and stitch the 4 pictures into a new picture. By repeating this operation, we get the corresponding Mosaic data-enhanced image, which greatly enriches the detection dataset, thereby improving the robustness of the model.

2.2.4. Mixup

First, we determine that the fusion ratio of the picture is λ according to the beta distribution, and λ is a random real number between [0, 1]. Then, for each batch of input images, we fuse it with randomly selected images according to the fusion ratio λ to obtain mixed tensor inputs. The calculation formula is shown in the following formula (2). Among them, the process of fusing the two pictures is to add each corresponding pixel value in the two pictures.

$$inputs = \lambda * images + (1 - \lambda) * images_random \quad (2)$$

Among them, λ is the fusion ratio; $images$ are each pixel value corresponding to the input image; $images_random$ is the value of each pixel corresponding to the randomly selected image.

As shown in Figure 5, we also use data enhancement methods such as four-way flipping and random scale transformation for images, and implicitly increase the amount of data collection through flipping, zooming, and improve the effectiveness of the detection model. To reduce the negative impact of category imbalance on the model, we introduced Focal Loss. This loss function is modified based on the standard cross-entropy loss. It can reduce the weight of easy-to-classify samples so that the model can focus more on difficult-to-classify samples during training, to measure the contribution of difficult-to-classify and easy-to-classify samples to the total loss, which eventually plays a role in accelerating the training process and enhancing the effect of the model.

Figure 5. Training images after mosaic and mixup operations.

2.3. Methods of Detection and Estimation

2.3.1. Target Detection

The traditional target detection preselection box is the standard box. When the target has a flip angle, the size and aspect ratio cannot reflect the true shape of the target. Crucian carp can realize free movement in three-dimensional space in the aquatic environment, and the turning range of crucian carp generally presents a large deformation, as shown in Figure 2, 80% of the angle changes are above 40 degrees. Therefore, in this case, the standard frame cannot fully fit the crucian carp and maximize the separation of the background. However, the rotating frame can solve this problem, as shown in Figure 6. Additionally, as shown in Figure 7, when multiple crucian carp overlap in the image, the use of a standard frame cannot effectively separate the crucian carp from the background pixels, which will cause errors such as reduced accuracy.

Figure 6. Target detection rotation frame and horizontal frame definition comparison chart. The red frame line represents the traditional horizontal frame, and the blue frame line represents the rotating frame.

Figure 7. Multi-level frame selection.

As shown in Figure 8, to explain the definition of the rotating frame, the following symbols are defined: height, width, angle, cy , cx . Where (cy, cx) represents the coordinates of the center point of the rotating bounding box. The special point is that the rotating frame is a matrix with the angle parameter, which is used to define the direction of the rotating frame.

The use of a rotating frame for the detection target can maximize the physical size of the target, minimize the existence of background pixels, and achieve a high-precision separation of the background and the detection target. This is very useful for the effective positioning of multiple targets.

Based on the Yolo 5 detector and according to the cosine similarity of the feature vector to realize the target recognition of the crucian carp, as shown in Figure 9.

Figure 8. The definition diagram of the rotating box. height and width respectively represent the length and width of the image, with the horizontal to the right as the positive direction. Take the point (X_3, Y_3) in the lower-left corner as the starting point, and rotate clockwise around the center point (cy, cx) .

Figure 9. The network structure and application of Yolo 5. The CBL component in the figure is composed of the Convolutional layer + BatchNormalization + Leaky_relu activation function. The Res-unit component draws on the residual structure in the Resnet network and can play a role in building a deeper network. The CSP_X component draws on the CSPNet network structure and is composed of a convolutional layer and X Res-unit modules. The focus component is to slice the data, which can play a role in the down-sampling operation without information loss. The SPP component adopts the maximum pooling method of 1×1 , 5×5 , 9×9 , and 13×13 for multi-scale fusion.

2.3.2. Pose Estimation

There are two main ideas in the field of pose estimation: top-down and bottom-up. In general, the former has a better effect, while the latter has a faster speed.

For our crucian carp research, the key influencing factors of top-down and bottom-up on the effect are compared as follows:

- (1) Crucian carp can realize free movement in three-dimensional space in the aquatic environment. The crucian carp's posture flip range is between $[0^\circ \sim 180^\circ]$, and as shown in Figure 2, 80% of the angle changes are above 40 degrees, so the overall degree of deformation is relatively large. This makes the posture change more complicated and has a greater impact on the subsequent activities of the crucian carp. From a research perspective, we can find that most of the crucian carp camps are active in clusters, and there is a lot of shelter and crowding. Based on using the rotating bounding box, the top-down method can continue to optimize, and better deal with the occlusion and crowding of the fish body, which is conducive to extracting the detection target from the pixel background.
- (2) The idea of bottom-up is to determine the location of the key points first, and then confirm the ownership of the key points. The main criterion is the affinity of key points, which is simply the distance between key points. Such a scheme can indeed achieve a considerable increase in speed, but when multiple targets are close in distance, it is extremely easy to divide the key points incorrectly, which greatly reduces the effect of the model. Therefore, we chose to use the top-down method.
- (3) The detection target of our research is crucian carp. Compared with humans, crucian carp is easier to identify, with more distinctive features, and is easier to extract. Top-down is used to train and output the key points of the complete image by extracting global target features, which is highly objective.
- (4) Top-down has higher accuracy, and bottom-up has faster speed. The use of a single top-down pose estimation model has a speed disadvantage, so while considering the speed and accuracy of the model at the same time, we used the Yolo 5 target detector in the early stage to obtain a significant speed blessing effect. In this way, the dual high-efficiency of the model's high precision and high speed can be achieved. DeepPose is a method that directly returns to the absolute coordinates of key points [32]. To express the posture of the fish body, we use the following symbols. We encode the positions of all $k = 10$ fish body joints into the definition $y = (\dots, y_i^T, \dots)^T, i \in \{1, \dots, k\}$, where y_i contains the horizontal and vertical of i^{th} coordinate. The marked image is represented by (x, y) , where x represents the image data, and y is the real posture vector of the fish body.

Since the joint coordinates are absolute image coordinates for the crucian carp, it is helpful to standardize them. The frame surrounding the fish body or part thereof is b . The rotating frame can maximize the representation of the complete crucian carp image, which is composed of its center b_c , width b_w and height b_h , defined as $b = (b_c, b_w, b_h)$.

$$N(y_i; b) = \begin{pmatrix} 1/b_w & 0 \\ 0 & 1/b_h \end{pmatrix} (y_i - b_c) \quad (3)$$

In addition, we can generalize the pose vector to the key points for all crucian carp, that is, $N(y; b) = (\dots, N(y_i; b)^T, \dots)^T$, which produces a normalized posture vector result.

Finally, we use $N(x; b)$ to crop the image x through the bounding box b . The bounding box b normalizes the crucian carp image by the box. For brevity, we use $N(*)$ to denote normalization, where b is the complete image frame.

We are based on the DeepPose network of crucian carp research, there are two stages. First, DNN regression pose estimation, the output of stage 1 is obtained as a sub-image, and the relatively rough crucian carp key point position is extracted. After that, the sub-image is sent to the cascade of the attitude regressor, and the refining operation is performed to further refine the regression results. For the points in the upper left corner that appear in the data collection, the corresponding regression criteria can be used from DeepPose to estimate the coordinate positions of the key points of the crucian carp that are occluded or not.

Input the normalized image data, after the key point coordinates are predicted by the AlexNet network, inversely normalize and restore to the original image. In DeepPose, the pose estimation problem is creatively regarded as a regression problem, and the neural network uses AlexNet. Among them, we also train and use the function $\psi(x; \theta) \in \mathbb{R}^{2k}$ to return to the normalized pose vector. Where θ represents the parameters of the model, and k is the coordinate value of the key point. Therefore, the normalized transformation from the equation is used. (1) Prediction of posing key point coordinates in absolute image coordinates y^* is

$$y^* = N^{-1}(\psi(N(x); \theta)) \quad (4)$$

The DNN network consists of several layers, each layer is a linear transformation, followed by a non-linear transformation. The first layer inputs a predetermined size image whose size is equal to the number of pixels multiplied by three color channels. The last layer outputs the returned target value, that is, the coordinates of the key points of the crucian carp.

The DNN network consists of 7 layers. As shown in Figure 10, use C to denote the convolutional layer, LRN to denote the local response normalization layer, P to denote the pooling layer, and F to denote the fully connected layer. Only the C and F layers contain learnable parameters, and the rest are parameterless. Both the C layer and the F layer consist of a linear transformation and a non-linear transformation. Among them, the nonlinear transformation is a rectified linear unit. For layer C, the size is defined as width $\times$ height $\times$ depth, where the first two dimensions have spatial significance, and depth defines the number of filters. The network input is a 256×256 image, which is input to the network through a set step size.

Figure 10. A schematic diagram of crucian carp's DNN-based posture regression in the DeepPose network. We use the corresponding dimensions to visualize the network layer, where the convolutional layer is blue and the fully connected layer is green.

What is achieved through the DeepPose network is the final joint absolute image coordinate estimation based on the complex nonlinear transformation of the original image. The sharing of all internal features in the key point regression also achieves the effect of robustness enhancement.

When training the crucian carp data, we chose to train linear regression on the last network layer and make predictions by minimizing the L_2 distance between the prediction and the crucian carp's real pose vector, rather than classification loss. The normalized definition of the training set is as follows:

$$D_N = \{(N(x), N(y)) \mid (x, y) \in D\} \quad (5)$$

Then, the L_2 loss used to obtain the best network parameters is defined as:

$$\arg \min_{\theta} \sum_{(x, y) \in D_N} \sum_{i=1}^k \|y_i - \psi_i(x; \theta)\|_2^2 \quad (6)$$

The loss function represents the L_2 distance between the normalized key point coordinates $N(y; b)$ and the predicted key point coordinates $\varphi(y; b)$. The parameter θ is optimized using backpropagation. For each unit of mini-batch training, calculate the adaptive gradient. Learning rate is the most important parameter, we set the initial learning rate to 0.0005.

Different stages of DeepPose use the same network structure φ , but the parameters θ of the network structure are different, and the regressor is denoted as $\psi(x; \theta_s)$, where $s \in \{1, \dots, S\}$ represents different stages, as shown in Figure 11.

Figure 11. In the DeepPose stage s , the refinement cascade is applied to the sub-image to refine the prediction of the previous stage.

In stage 1, the crucian carp we studied starts from surrounding the complete image or the bounding box B_0 obtained by the detector. The initial pose is defined as follows:

Stage 1:

$$y^1 \leftarrow N^{-1}(\psi(N(x; b^0); \theta_1); b^0) \quad (7)$$

b_0 represents the bounding box of the entire input image.

For the subsequent stage s ($s \geq 2$), $i \in \{1, \dots, k\}$, it will first be sent to the cascade through the subgraph defined in the previous stage, and return to the refinement displacement. Then estimate the new joint box b_i^s :

Stage s :

$$y_i^s \leftarrow y_i^{(s-1)} + N^{-1}(\psi_i(N(x; b); \theta_s); b) \quad (8)$$

$$b_i^s \leftarrow (y_i^s, \sigma \text{diam}(y^s), \sigma \text{diam}(y^s)) \quad (9)$$

where diam stands for diameter.

In the training phase of the cascade stage, complete training data enhancement will be carried out. First, an instance and a joint are uniformly sampled from the original data, and then simulated prediction is performed, and then the simulated prediction is generated according to the sampling displacement $N_i^{(s-1)}$ from the Gaussian distribution (GT) to define the following Equation (10):

$$D_A^s = \{ (N(x; b), N(y_i; b)) \mid (x, y_i) \sim D, \delta \sim \mathcal{N}_i^{(s-1)}, b = (y_i + \delta, \sigma \text{diam}(y)) \} \quad (10)$$

The enhanced data changed from D to D_A^s , and normalized again:

$$\theta_s = \arg \min_{\theta} \sum_{(x, y_i) \in D_A^s} \| y_i - \psi_i(x; \theta) \|_2^2 \quad (11)$$

2.3.3. Convert a Rotating Frame to a Horizontal Frame

To indicate the adjustment process of the rotating frame, we use the following symbols. First, the original image of a rotating frame can be defined by the center point coordinates (cy, cx), width, height, depth, and rotation angle θ . For the quadrilateral of the rotating box, draw the 4 corner points of the quadrilateral: $[X_0, Y_0]$, $[X_1, Y_1]$, $[X_2, Y_2]$, $[X_3, Y_3]$. Then, the coordinates of the four corner points are transformed and mapped through the rotation transformation matrix M to obtain the new coordinates of the four corresponding corner points of the rotated image. Finally, when the feature is missing after the transformation and there is a need to make up, choose to expand the canvas, and then perform operations such as translation based on the coordinates of the four new points through the translation parameter to avoid incomplete feature information. In this way, a complete horizontal frame can be obtained. As shown in Figure 12. For the rotation frame, there must be a rotation transformation matrix that can adjust the rotation frame to a horizontal frame. We perform the rotation transformation based on the center point. The matrix M is defined as follows:

$$\begin{aligned}x_1 &= \cos \theta \\y_1 &= \sin \theta \\x_2 &= -\sin \theta \\y_2 &= \cos \theta \\x_3 &= (1 - \cos \theta)cx + cy \sin \theta \\y_3 &= (1 - \cos \theta)cy - cx \sin \theta\end{aligned}\quad (12)$$

$$\begin{aligned}M &= \begin{bmatrix} 1 & 0 & cx \\ 0 & 1 & cy \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & -cx \\ 0 & 1 & -cy \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \cos \theta & -\sin \theta & (1 - \cos \theta)cx + cy \sin \theta \\ \sin \theta & \cos \theta & (1 - \cos \theta)cy - cx \sin \theta \\ 0 & 0 & 1 \end{bmatrix}\end{aligned}\quad (13)$$

$$M = \begin{bmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ 0 & 0 & 1 \end{bmatrix}\quad (14)$$

When expanding the canvas, the new height new_H and the new width new_W are defined as follows:

$$new_H = \text{int}(w * \text{fabs}(\sin(\text{radians}(\text{angle}))) + h * \text{fabs}(\cos(\text{radians}(\text{angle})))) \quad (15)$$

$$new_W = \text{int}(h * \text{fabs}(\sin(\text{radians}(\text{angle}))) + w * \text{fabs}(\cos(\text{radians}(\text{angle})))) \quad (16)$$

Based on the matrix M , the translation parameters are defined as follows:

$$\begin{aligned}M[0, 2] + &= (new_W - b_w)/2 \\ M[1, 2] + &= (new_H - b_h)/2\end{aligned}\quad (17)$$

Based on the above steps, a single complete crucian carp level map can be generated, and then the single sheet is continuously sent to the Yolo 5 detector. After the ID is recognized, the attitude estimation based on DeepPose is performed.

Figure 12. Rotation transformation flow chart. We construct a coordinate system whose direction is the same as that of a general image coordinate system. The upper left corner of the rotating frame is the origin, the positive x-direction is along the edge, and the positive y direction is along the edge down. The canvas includes all target features in the rotating frame. Left: Initial rotation box, labeling all related symbols. Right: Only the transformation process is included.

3. Experiment and Result

For the selection of object detection models, Table 1 lists the test results of existing mainstream object detection models one by one: After comprehensive consideration of various metrics such as Accuracy and Recall, we selected Yolo 5, and then tested it on the customized crucian carp dataset to verify its accuracy.

Table 1. Comparison of object detection models.

Model	P	R	F1	mAP@0.5	mAP@0.5:0.95	Inference @Batch_Size 1 (ms)
CenterNet	95.21%	92.48%	0.94	94.96%	56.38%	32
Yolo 4s	84.24%	94.42%	0.89	95.28%	52.75%	10
Yolo 5s	92.39%	95.38%	0.94	95.38%	58.31%	8
EfficientDet	88.14%	91.91%	0.90	95.19%	53.43%	128
RatinaNet	88.16%	93.21%	0.91	96.16%	57.29%	48

During the experiment, we found that the orientation of the fish in the dataset affected the detection result, and thus had a negative effect on pose estimation. Considering that the root cause of the problem lies in the direction of fish, we propose to use rotating object detection instead of common object detection to solve the problem. Table 2 lists the performance results of R-CenterNet versus R-Yolo 5 on the COCO dataset and the test results on the customized crucian carp dataset. Finally, R-Yolo 5s is selected as the model of rotating object detection.

Table 2. Comparison of rotating object detection models.

Model	P	R	F1	mIOU	mAngle	Inference@Batch Size 1 (ms)
R-CenterNet	88.72%	87.43%	0.88	70.68%	8.80	76
R-Yolo 5s	90.61%	89.45%	0.90	75.15%	8.26	43

At the same time, we also designed a synchronous control experiment of rotating object detection group and ordinary object detection group to verify the advantages of rotating object detection, refer to Figure 13 for detailed effect comparison. It can be seen that when there are multiple targets in the test picture and the fish body direction is not horizontal, the ordinary object detection often has problems such as misidentification of targets and incomplete recognition of key points; In this case, rotating object detection has obvious advantages over ordinary object detection, but in the actual environment, multi-target and non-horizontal fish orientation are very common. After all, we selected rotated-Yolo 5 as the main object detection model.

Figure 13. Comparison of rotating object detection results. The figure on the left is the result of the rotating object detection group, and the figure on the right is the result of the ordinary object detection group.

To further improve the effectiveness of the rotated Yolo 5 model and enhance its generalization ability, different tricks were used to deal with the model. Table 3 lists the effect of evaluation metrics after using HSV_Aug, Mosaic, MixUp, Fliplrud, RandomScale, and other tricks and Focal Loss. Experiments verify that the best prediction effect can be obtained when the tricks processing is used simultaneously.

Table 3. R-Yolo 5 with different tricks.

HSV_Aug	FocalLoss	Mosaic	MixUp	Other Tricks	mAP@0.5
×	×	×	×	×	77.32%
✓	×	×	×	×	77.98%
✓	✓	×	×	×	77.42%
✓	✓	✓	×	×	79.05%
✓	✓	✓	✓	×	81.12%
✓	×	×	✓	×	80.64%
✓	✓	✓	×	Fliplrud	79.68%
✓	✓	×	×	Fliplrud	80.37%
✓	×	✓	✓	Fliplrud	81.46%
✓	×	×	×	Fliplrud	78.99%
✓	×	×	×	RandomScale(0.5~1.5)	78.99%
✓	✓	✓	✓	Fliplrud	81.88%
✓	✓	✓	✓	RandomScale(0.5~1.5)	81.88%

No matter in the experimental condition or the actual environment, there are many multiple objects in the pictures taken by the camera, but our pose estimation is for a certain target in the image, so we propose to use rotating object detection. In addition, considering the poor performance of bottom-up in dealing with multi-objective situations, the methods used in this experiment are all top-down. That is, the target fish is identified by a rotating object detection frame, and then its pose is estimated. For pose estimation, DeepPose, Simplebaseline, hrnet, udp, darkpose, and other models are used to benchmark the subgraph obtained after rotating object detection.

3.1. Train Settings

In the R-Yolo 5 model, we used all the tricks in Table 3. Through experimental comparison, we determined the hyperparameters of the HSV color model as H (hue): 0.015, S (saturation): 0.7, V (value): 0.4; The probability of flipping up and down is set to 0 and the probability of flipping left and right is set to 0.5; Set Mosaic and Mixup to 1. During the training, we used the Adam optimizer uniformly and set the initial learning rate to 0.0005. After integrating the rotating object detection part (R-Yolo 5) and the pose estimation part (DeepPose), we first set the image input size as 256×256 , and then put the customized crucian carp dataset into the network at the batch size of 16 for training. Meanwhile, due to Adam's rapid convergence, we set epochs to 120 and save the weight file every 10 iterations to ensure effective convergence of the validation set.

3.2. Benchmark Results

The training is carried out by continuous debugging of hyperparameters until the evaluation metrics of the verification set converge. However, in the training process, the convergence process is too fast and the verification effect is very poor, but the metrics are almost the same, so we try to control variables for each model: The epoch, batch size, and input size were unified, and the input size was unified as 256×256 . Meanwhile, the size of the heat map was also changed accordingly, see Figure 14. It can be seen from the experimental results that DeepPose has great advantages over hrnet and Simplebaseline among several models that only use PCK for single indicator training. Therefore, we try to change hrnet to hrnetv2 with a stronger effect, and change the backbone of Simplebaseline to Mobilenetv2, and then carry out multi-indicator training, and finally get a 16.5% improvement in PCK compared with the single indicator training. Then, we added udp and Darkpose model multi-indicator training to compare the effects: Among them, the udp model is not significantly improved compared with hrnetv2, while Darkpose has a slight improvement in AUC and EPE under the condition that there is not much difference between PCK. It can also be seen from the verification renderings that the key points of Darkpose are more accurately positioned. Although Darkpose has the best effect in the model trained with multiple indicators, in the testing process, when there are multiple fish in the picture, Darkpose tends to have the situation of dislocation of key points or mass aggregation, while it is more accurate in the case of single fish. Therefore, we used the multi-indicator training model to test the effect again and found that the key point positioning was affected by noise points. Even if the rotating object detection was carried out first, it was difficult to avoid the presence of multiple fish in the image. Considering the limitation of dataset size, the multi-indicator training effect of the model is indeed poor, and it can not adapt to the multi-objective situation, so we finally choose the single-indicator training model. Of course, we also prepared a comparison term—NME for metric PCK. We used Wingloss in DeepPose and changed the metric to NME for training again. The test results were not significantly different from the DeepPose effect of the original PCK training, but the speed decreased significantly. Therefore, we finally chose the DeepPose using PCK for single indicator training as the model of pose estimation.

In this experiment, aiming at the problem of large positioning deviations of the key points in the dorsal fin and tail of fish, we proposed a scheme to process the images by using rotating object detection first. Through the comparison of experiments, it is proved that to a large extent, the problems of undetectable dorsal fin key points and misplaced tail key points can be improved. Secondly, a key point detection model with high accuracy was obtained by abandoning multi-indicator training. As shown in Table 4, PCK is selected as the final metric by comparing different single metrics. Of course, the effect of this model is not satisfactory in the case of multi-target images or multiple occlusion of fish bodies in the experiment, which also shows that there is still a lot of room for improvement of this model. However, considering that there are only 1541 images in our dataset this time, it is believed that a larger dataset covering more situations will play a great role in improving

the effectiveness of the model and enhancing the generalization ability. Figure 15 shows the final result of our experiment.

Figure 14. The metrics of each model changed during training.

Table 4. Pose estimation model comparison.

Model	Metric
Simplebaseline	PCK: 0.8131
hrnet	PCK: 0.8222
DeepPose	PCK: 0.9781
hrnetv2	PCK: 0.9585, AUC: 0.6994, EPE: 10.4704
Mobilenetv2 + Simplebaseline	PCK: 0.9480, AUC: 0.6878, EPE: 11.5483
udp	PCK: 0.9546, AUC: 0.7124, EPE: 10.2830
darkpose	PCK: 0.9559, AUC: 0.7127, EPE: 9.6965
DeepPose + Wingloss	NME: 0.1250

Figure 15. Final effect display.

4. Discussion

Object detection and pose estimation based on computer vision has always been the mainstream schemes of real-time detection. At the same time, it has great prospects in the field of surveillance and security. However, the pose estimation of fish is different this time. Compared with previous datasets, this fish dataset has stronger randomness and complexity in spatial distribution and relative location information. Thus, for the first time, we make a preliminary exploration of the dataset and propose a two-stage approach for pose estimation, the feasibility of fish pose estimation was investigated. According to the above experiments and analysis, we do the following discussion for this experiment:

4.1. Contribution to Pose Estimation of Fish

Pose estimation is widely used in various fields such as human–computer interaction, behavior recognition, and virtual reality. In the field of human–computer interaction, the human body pose estimation technology accurately captures the details of human movements, and after obtaining the human body movements, it can interact with the computer without contact. However, apart from these traditional applications, animal pose estimation also has great research value, such as behavior analysis, wildlife protection, etc. [26,27]. Although aquatic animals do not have limb movement, their movement in the water is more open and can flip freely without being restricted by angles. The usual data annotations have become extremely limited.

Therefore, this paper uses the top-down method to detect the rotating box object on the golden crucian carp data, detect each golden crucian carp, and then detect the key points of each golden crucian carp, to achieve the purpose of multi-target pose estimation step by step. Not only the performance is greatly improved, but also proves the applicability of multi-target pose estimation in aquatic animals.

4.2. Contribution to Underwater Real-Time Detection

As mentioned in the introduction, to further accelerate the reform of the fish farming industry, it is essential to realize accurate fish farming and real-time monitoring. However, the traditional and backward electronic equipment has very low adaptability in a complex underwater environment and even harms the growth of fish. Therefore, to achieve real-time detection of fish, researchers have proposed many methods. Hsiao et al. [33] applied the target detection and recognition algorithm to fish detection in an underwater environment by extracting multi-layer features of underwater visual data, and the recognition rate can reach 81.8%, but its real-time performance is far from meeting the requirements of practical application. Cutter et al. [34] applied multiple cascaded classifiers to fish detection in a seabed environment. Although the detection rate can reach 89%, the detection results are not ideal in more complex environments. Li et al. [35] applied the improved Faster R-CNN to underwater fish target detection. Although the detection rate has been improved, the FPS is only 11, which can not meet the basic requirements of real-time monitoring. In this regard, we propose a rotating box object detection method for golden crucian carp, which not only has the detection rate exceeded 90% but also the FPS is as high as 23.26, meeting the basic requirements of real-time monitoring.

4.3. Different from Existing Methods

4.3.1. Benefits of the Rotating Frame

Traditional target detection is based on standard boxes. However, in this study, when multiple fish overlap in the image, the standard frame cannot fully fit the fish, which will cause problems such as reduced accuracy. However, the rotating frame can solve this problem, can reflect the physical size of the target to the maximum, minimize the background pixels, and achieve high-precision separation of the background and the detection target.

4.3.2. Compare with Other Methods

Our research method is mainly divided into two stages. The first stage is to use the detection image based on the rotating frame as the basic input [29]. First of all, we are the first to test all existing mainstream target detection models, and comprehensively consider various indicators such as accuracy and recall rate. From this, two backbone models with higher indicators (R-CenterNet and R-Yolo 5s) were selected. In the later experiment process of target detection, we successively compared the test results of R-CenterNet and R-Yolo 5s on the grass gold dataset. We found that R-Yolo 5s takes priority in indicators such as accuracy and recall rate. The application effect of CenterNet is not very good, and the learning efficiency is low. Taking R-Yolo 5s as the target detection backbone, and using the residual structure of the Res-unit component to increase the

depth of network construction. During the down-sampling process, the slicing operation ensures that information is not lost. In the follow-up, we will discuss the process and results of different methods separately. Common target detection of grass goldfish often has problems such as misidentification of targets and incomplete recognition of key points.

In the second stage, we compare the two mainstream ideas of attitude estimation. To meet the high precision and high-speed effect of the final model, we use top-down. The bottom-up solution can indeed get a lot of speed improvement, but when multiple targets are close to each other, it is extremely easy to divide the key points incorrectly, which greatly reduces the effect of the model. So we use the top-down method to extract global target features to output key points and considering that the detection speed of the R-Yolo 5s target detector in the first stage is extremely fast, the use of top-down has higher accuracy.

For the pose estimation part, DeepPose has greater advantages compared to HRNet and Simple-baseline. We try to replace HRNet with HRNetv2, which is more effective. However, hrnetv2 aggregates all parallel convolutions, which often makes the final effect poor in the process of maintaining high resolution. Compared with the posture estimation method using Simple-baseline, it only combines the upsampling and convolution parameters into the deconvolution layer in a simpler way, without using a residual connection. This does not apply to the complex situation of multiple grass goldfish in the water body.

Under a variety of comparisons, we use DeepPose to estimate pose and directly based on the absolute coordinates of the key points of the DNN regression fish [32]. The advantage of this method is that it can regress to the joint coordinates in a DNN-based manner, and the regressive cascade has the function of capturing context and reasoning about the posture in an overall manner.

4.4. Limitations and Future Work

The manual labeling process of fish images is cumbersome, which directly leads to the small size of the dataset in this study. Moreover, the images collected from multiple angles are diverse, which inevitably makes it difficult for the naked eye to distinguish key points. Mislabeling of key points will inevitably affect the model effect and reduce the accuracy of pose estimation at some angles. In addition, the fish used for this training and testing are of a specific species. The resulting model is for this species of fish. While it has considerable accuracy and speed, the results can be wrong when applied to other species of fish. In this case, we need to expand the dataset of this time, conduct further training, and enhance the generalization ability and robustness of the model. In consideration of generalization ability, we will expand the types of fish and the number of images in the dataset in the future for data augmentation, to achieve a better model effect.

Secondly, Deeppose and single index training are finally selected in this paper, but the Deeppose model was proposed in 2014, which has a longer time than the current model. Moreover, single index training has no advantage in theory, but just in this small-scale data, the effect is superior. In the following research, we will conduct a more detailed study on the pose estimation model and try to introduce newer network modules to optimize the network structure and improve the model effect as much as possible.

5. Conclusions

A new large-scale dataset of ten different golden crucian carp was proposed, including boundary frames and pose estimation key points. This paper introduces a rotating object detection and poses an estimation algorithm for the golden crucian carp. Firstly, the object identification and detection of fish in the rotating pre-selection box is carried out, which achieves a better recognition effect compared with the traditional horizontal box. Then, the pose recognition is carried out based on the collected key point characteristics, and the behavior of the fish is predicted. Finally, we demonstrate the predictive effect of the fish in different networks. The technology can effectively identify fish and perform pose estimation, eliminating many limitations compared to humans. Once optimized, our

algorithmic system can provide a more efficient and accurate way to facilitate long-term studies of known individuals.

Author Contributions: Conceptualization, K.J.; Data curation, B.L., Z.X. and K.J.; Formal analysis, X.G.; Funding acquisition, F.L. and X.G.; Investigation, B.L.; Methodology, B.L. and K.J.; Project administration, X.D.; Resources, F.L. and X.D.; Supervision, C.M.; Validation, X.G. and C.M.; Writing—original draft, B.L., K.J., Z.X. and F.L.; Writing—review and editing, B.L. and J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: The animal use protocol listed below was reviewed and approved by the animal ethics and welfare committee (AEWC) and approved by the Sichuan Agricultural University Institutional Animal Care and Use Committee. The approval No. is 2020053.

Data Availability Statement: The data are available online at: <https://drive.google.com/file/d/1WvhrnLJVm18BwaYTzyJCJtEO8gxI9yth/view?usp=sharing> (accessed on 22 October 2021).

Acknowledgments: Thanks to Qinli Liu and Jie Liu for their help and suggestions on data annotation.

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

References

1. Sun, D.W. *Computer Vision Technology for Food Quality Evaluation*; Academic Press: Cambridge, MA, USA, 2016.
2. Vimala, J.S.; Natesan, M.; Rajendran, S. Corrosion and Protection of Electronic Components in Different Environmental Conditions—An Overview. *Open Corros. J.* **2009**, *2*, 105–113.
3. Walther, D.; Edgington, D.R.; Koch, C. Detection and tracking of objects in underwater video. In Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004, Washington, DC, USA, 27 June–2 July 2004.
4. Rashid, M.; Gu, X.; Yong, J.L. Interspecies Knowledge Transfer for Facial Keypoint Detection. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
5. Zou, Z.; Shi, Z.; Guo, Y.; Ye, J. Object detection in 20 years: A survey. *arXiv* **2019**, arXiv:1905.05055.
6. Oksuz, K.; Cam, B.C.; Kalkan, S.; Akbas, E. Imbalance Problems in Object Detection: A Review. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 3388–3415.
7. Hechun, W.; Xiaohong, Z. Survey of deep learning based object detection. In Proceedings of the 2nd International Conference on Big Data Technologies, Jinan, China, 28–30 August 2019; pp. 149–153.
8. Zhao, Y. Improved SSD Algorithm Based on Multi-scale Feature Fusion and Residual Attention Mechanism. In Proceedings of the 2021 3rd International Conference on Advances in Computer Technology, Information Science and Communication (CTISC), Shanghai, China, 23–25 April 2021; pp. 87–91.
9. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015.
10. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149.
11. Rajaram, R.N.; Ohn-Bar, E.; Trivedi, M.M. RefineNet: Iterative refinement for accurate object localization. In Proceedings of the IEEE International Conference on Intelligent Transportation Systems, Rio de Janeiro, Brazil, 1–4 November 2016.
12. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
13. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
14. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
15. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Proceedings of the European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016.
16. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal Loss for Dense Object Detection. In Proceedings of the 2017 IEEE Transactions on Pattern Analysis & Machine Intelligence, Venice, Italy, 7 August 2017; pp. 2999–3007.
17. Li, J.; Su, W.; Wang, Z. Simple Pose: Rethinking and Improving a Bottom-up Approach for Multi-Person Pose Estimation. In Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-20), New York, NY, USA, 7–12 February 2020; pp. 1354–1361.
18. Olmos, A.; Trucco, E. Detecting man-made objects in unconstrained subsea videos. In Proceedings of the British Machine Conference (BMVC), Wales, UK, 2–5 September 2002; pp. 1–10.

19. Wang, C.Y.; Liao, H.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W. CSPNet: A New Backbone that can Enhance Learning Capability of CNN. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020.
20. Xiu, L.; Min, S.; Qin, H.; Chen, L. Fast accurate fish detection and recognition of underwater images with Fast R-CNN. In Proceedings of OCEANS 2015-MTS/IEEE Washington, Washington, DC, USA, 19–22 October 2015.
21. Xu, C.; Govindarajan, L.N.; Zhang, Y.; Cheng, L. Lie-X: Depth Image Based Articulated Object Pose Estimation, Tracking, and Action Recognition on Lie Groups. *Int. J. Comput. Vis.* **2016**, *123*, 454–478.
22. Xu, W.; Matzner, S. Underwater Fish Detection using Deep Learning for Water Power Applications. In Proceedings of the 5th Annual Conf. on Computational Science & Computational Intelligence (CSCI'18), Las Vegas, NV, USA, 13–15 December 2018.
23. Knausgård, K.M.; Wiklund, A.; Sørvalen, T.K.; Halvorsen, K.T.; Kleiven, A.R.; Jiao, L.; Goodwin, M. Temperate fish detection and classification: A deep learning based approach. *Appl. Intell.* **2021**. Available online: <https://link.springer.com/article/10.1007/s10489-020-02154-9#citeas> (accessed on 1 October 2021). [CrossRef]
24. Su, H.; Kong, W.; Jiang, K.; Liu, D.; Gong, X.; Lin, B.; Li, J.; Wang, H.; Xu, C. Gold crucian carp identification based on Siamese network. In Proceedings of the International Conference on Image Processing and Intelligent Control (IPIC 2021); SPIE: Bellingham, WA, USA, 2021; pp. 191–194.
25. Kong, W.; Li, D.; Li, J.; Liu, D.; Liu, Q.; Lin, B.; Su, H.; Wang, H.; Xu, C. Detection of golden crucian carp based on YOLOV5. In Proceedings of the 2021 2nd International Conference on Artificial Intelligence and Education (ICAIE), Dali, China, 18–20 June 2021; pp. 283–286.
26. Zheng, L.; Zhang, H.; Sun, S.; Chandraker, M.; Yang, Y.; Tian, Q. Person re-identification in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1367–1376.
27. Zheng, L.; Shen, L.; Tian, L.; Wang, S.; Wang, J.; Tian, Q. Scalable Person Re-identification: A Benchmark. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015.
28. Salman, A.; Siddiqui, S.A.; Shafait, F.; Mian, A.; Shortis, M.R.; Khurshid, K.; Ulges, A.; Schwanecke, U. Automatic fish detection in underwater videos by a deep neural network-based hybrid motion learning system. *ICES J. Mar. Sci.* **2019**, *77*, 1295–1307.
29. ultralytics/yolov5. Available online: <https://github.com/ultralytics/yolov5> (accessed on 7 May 2020).
30. Boyat, A.K.; Joshi, B.K. A Review Paper: Noise Models in Digital Image Processing. *Signal Image Process. Int. J.* **2015**, *6*, 63–75.
31. Kong, Z.; Yang, X.; He, L. A Comprehensive Comparison of Multi-Dimensional Image Denoising Methods. *arXiv* **2020**, arXiv:2011.03462.
32. Toshev, A.; Szegedy, C.D. Human pose estimation via deep neural networks. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 1653–1660.
33. Hsiao, Y.H.; Chen, C.C.; Lin, S.I.; Lin, F.P. Real-world underwater fish recognition and identification, using sparse representation. *Ecol. Inform.* **2014**, *23*, 13–21.
34. Cutter, G.; Stierhoff, K.; Zeng, J. Automated Detection of Rockfish in Unconstrained Underwater Videos Using Haar Cascades. In Proceedings of the Applications and Computer Vision Workshops (WACVW), 2015 IEEE Winter, Waikoloa, HI, USA, 6 January 2015.
35. Xiu, L.; Tang, Y.; Gao, T. Deep but lightweight neural networks for fish detection. In Proceedings of the OCEANS 2017-Aberdeen, Aberdeen, UK, 19–22 June 2017.