Mathematical Biosciences and Engineering

MBE, 21(2): 1765–1790.
DOI: 10.3934/mbe.2024076
Received: 12 November 2023
Revised: 11 December 2023
Accepted: 13 December 2023
Published: 02 January 2024

http://www.aimspress.com/journal/MBE

Research article

An efficient detection model based on improved YOLOv5s for abnormal surface features of fish

Zheng Zhang*, Xiang Lu and Shouqi Cao

College of Engineering Science and Technology, Shanghai Ocean University, Shanghai 201306, China

* **Correspondence:** Email: z-zhang@shou.edu.cn; Tel: +8602161900812.

Abstract: Detecting abnormal surface features is an important method for identifying abnormal fish. However, existing methods face challenges in excessive subjectivity, limited accuracy, and poor real-time performance. To solve these challenges, a real-time and accurate detection model of abnormal surface features of in-water fish is proposed, based on improved YOLOv5s. The specific enhancements include: 1) We optimize the complete intersection over union and non-maximum suppression through the normalized Gaussian Wasserstein distance metric to improve the model's ability to detect tiny targets. 2) We design the DenseOne module to enhance the reusability of abnormal surface features, and introduce MobileViTv2 to improve detection speed, which are integrated into the feature extraction network. 3) According to the ACmix principle, we fuse the omni-dimensional dynamic convolution and convolutional block attention module to solve the challenge of extracting deep features within complex backgrounds. We carried out comparative experiments on 160 validation sets of in-water abnormal fish, achieving precision, recall, mAP_{50}, $mAP_{50:95}$ and frames per second (FPS) of 99.5, 99.1, 99.1, 73.9% and 88 FPS, respectively. The results of our model surpass the baseline by 1.4, 1.2, 3.2, 8.2% and 1 FPS. Moreover, the improved model outperforms other state-of-the-art models regarding comprehensive evaluation indexes.

Keywords: abnormal surface features of fish; YOLOv5s; normalized Gaussian Wasserstein distance metric; MobileViTv2 module; Densone module; ACmix; ODC-CBAM

1. Introduction

Aquaculture provides humans with a wealth of nutrients and has become an important part of the global agricultural economy. According to statistics, 88% of global annual fishery production is directly consumed by humans [1,2]. With population growth and economic development, the demand for fish continues to increase, and the scale of aquaculture is gradually expanding, which brings huge challenges to fish farming [3]. During fish farming, abnormalities such as diseases and parasites may occur in fish, resulting in a decrease in fish attributes, quality and fish welfare. Fish abnormality detection helps farmers adjust breeding strategies in a timely manner, prevent disease outbreaks and improve breeding efficiency [4,5]. In the past, manual visual inspection was the primary method for abnormal fish detection. However, this method has problems such as low efficiency, high missed detection rate and strong subjectivity. Detecting abnormal features on the surface of fish is an important basis for distinguishing abnormal fish. Therefore, rapid and accurate detecting of abnormal surface features of fish has become a hot issue in aquaculture.

Computer vision technology is an effective, cost-efficient and non-invasive detection technique, carrying substantial significance in driving the automation and intelligence of aquaculture [6]. It has great potential for abnormal fish detection in aquaculture [7]. With the development of artificial intelligence, such as computer vision and deep learning, especially in object detection, image classification and image segmentation, researchers have begun to detect abnormal fish surface features by applying neural networks to distinguish abnormal fish.

Yasruddin et al. [8] used computer vision and deep convolutional neural networks to detect fish diseases and used Faster-RCNN to train the surface features of diseased fishes. The results showed that the recognition accuracy was satisfactory. Ashraf and Atia [9] used a transfer learning model to learn two different shrimp disease signatures and detect diseased shrimps from normal shrimps. Wang et al. [10] proposed a computer vision-based detection method for abnormal surface features of Penaeus vannamei. Rapid detection of Penaeus vannamei diseases is achieved through image enhancement methods such as denoising and feature enhancement, as well as the LeNet model. The accuracy of the deep learning model used reaches approximately 96.1%. Chen et al. [11] proposed a two-stage ImageNet deep learning model with a convolutional neural network structure. The model was able to classify three abnormal appearances of grouper, achieving a high average accuracy of 98.94%. Gupta et al. [12] used a convolutional neural network based on VGG19 for fish wound detection, which can classify normal fish and abnormal fish, and the recognition accuracy reached 96.7%. In this body of research researches, although deep learning techniques have shown promising results for fish behavior detection, there are still certain limitations: 1) The detection of abnormal fish in complex backgrounds presents challenges of missing and inaccurate detection. 2) The fish abnormal surface feature data sets used were collected on the workbench and cannot be suitable for abnormal fish detection in underwater scenes. 3) Previous enhancements made by convolutional neural networks have some drawbacks such as insufficient feature extraction and complex model network structure, resulting in an inability to maintain a balance between model complexity, detection speed and detection accuracy.

You only look once (YOLO) [13–18] is an advanced single-stage object detection algorithm, which can be used, for example, small target detection in aquaculture, detection of key components of power transmission lines and detection of cigarette appearance defects, etc. Due to its exceptional performance, it has found extensive applications in land-based recirculating aquaculture systems. Yu et al. [19] proposed a fish skin disease detection model based on the YOLOv4

model, combined with depth-separable convolution and optimized feature extraction network and activation function. The proposed model has high learning ability and the model is lightweight. Compared with the baseline, its mean average precision (mAP) and detection speed are increased by 12.39% and 19.31 FPS, respectively. Wang et al. [20] proposed a diseased fish detection model based on improved YOLOv5s, using the C3 structure instead of the cross-stage partial (CSP) structure, and replacing all 3 × 3 convolutions in the backbone network with parallel 3 × 3, 1 × 3 and 3 × 1 convolutions. The convolution kernel group composed of kernels and the introduction of convolutional block attention module (CBAM) attention mechanism achieved an average accuracy of 99.38%. Prasetyo et al. [21] enhanced the YOLOv4-tiny model for the determination of fish freshness, species classification, and biomass estimation. Their approach involved the integration of novel techniques such as the wing convolutional layer (WCL) and tiny spatial pyramid pooling (Tiny-SPP) to refine and balance diverse feature representations. They effectively optimized computational resources by employing bottleneck and expansion convolution (BEC) for feature fusion. To further improve the model's detection accuracy, they introduced an additional small object detector. Zhao et al. [22] proposed a high-precision lightweight model that uses an improved YOLOv4 to detect dead fish, significantly reducing the number of model parameters and computational amounts. Li et al. [23] introduced a real-time detection approach for identifying abnormal fish behaviors, which combines images of mosaic pixel points with an enhanced version of YOLOv5s, referred to as BCS-YOLOv5. Their proposed method not only improves the extraction of positional information for abnormal fish, but also enables quantitative detection of similar abnormal behaviors. Based on image fusion, BCS-YOLOv5 achieved an impressive inference accuracy of 96.69% on the dataset. The majority of the aforementioned studies have focused on enhancing YOLOv5 for specific detection tasks, resulting in notable improvements and achieving commendable evaluation metrics.

The above-mentioned studies show that good accuracy has been achieved in detecting obvious abnormal fish surface features. However, there are certain limitations in extracting abnormal surface features for small targets and complex scenes. Therefore, this study presents an enhanced YOLOv5-based detection model designed for abnormal surface features. Several novel improvements are introduced in our method, distinguishing it from prior research, as outlined below:

• We introduce the normalized Gaussian Wasserstein distance (NWD) metric to optimize the loss function and non-maximum suppression (NMS) of YOLOv5s to enhance the model's ability to detect small targets and speed up the model's convergence speed.

• We introduce the lightweight MobileViTv2 module and designed DenseOne module. These enhancements improve detection accuracy, while reducing the model size and parameters for resource-constrained edge devices.

• According to the ACmix principle, we obtain the ODC-CBAM module by fusing omni-dimensional dynamic convolution (ODConv) and CBAM, and further integrate it into the feature extraction network, which reduces the missed detection rate and false detection rate of abnormal surface features located in complex scenes.

The rest of this article is as follows. Section II proposes methods for the problem and improves the detailed description of the structure of YOLOv5s and the improvement point. Section III describes experimental data collection, data set construction and some experimental details. Section IV analyzes the experimental results. Section V summarizes the work of this article.

2. Methodology

2.1. Detection methods for abnormal fish

After reviewing the literature, research and interviews with relevant breeders, we identified the following challenges in discerning abnormal fish by the detection of surface features: 1) Since abnormal surface features of fish are an occasional phenomenon in aquaculture, this creates a problem of data scarcity. Moreover, annotating the abnormal surface features requires a lot of time and resources. Therefore, it is difficult to construct a data set of abnormal fish surface features. 2) As shown in the Figure 1, the abnormal surface features of longsnout catfish are clearly visible on the workbench. In-water environments differ from those on the table and often exhibit phenomena such as reflectivity, which can result in unclear fish images. Longsnout catfish in the water may also exhibit such as pixel blur, small size due to variations in distance and serious overlap. 3) Although the current convolutional neural networks exhibit good detection accuracy, they have shortcomings such as weak learning ability for abnormal surface features of small targets and slow detection speed.

Figure 1. Images of abnormal surface features of fish photographed at a table.

Inspired by the current challenges and previous detection of abnormal fish surface features, this study proposes an object detection model based on improved YOLOv5s. Input the data set into the backbone with MobileViTv2 and ODC-CBAM as the main body and extract the abnormal surface features of abnormal longsnout catfish in the complex background. Then, use the designed DenseOne module to improve the reusability of the features and reduce the overall network complexity. Finally, the NWD metric is introduced to optimize the loss function and NMS of the baseline to enhance the model's ability to detect small targets and accelerate the convergence of the model. The method flow chart is shown in the Figure 2.

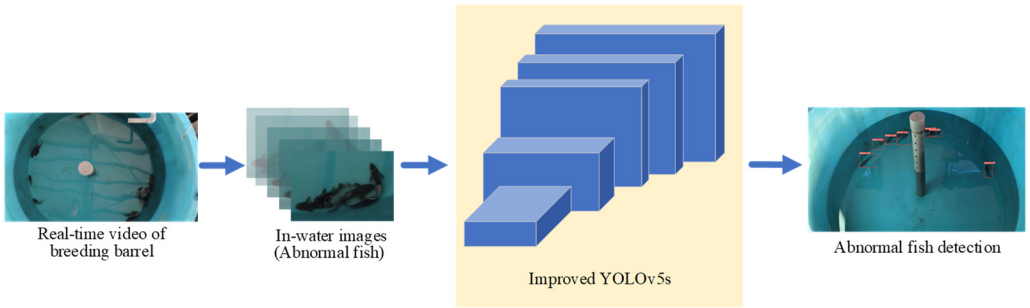

Figure 2. Abnormal fish detection method based on improved YOLOv5s.

2.2. Improved YOLOv5

YOLOv5 represents a notable enhancement over the YOLOv4 introduced in 2020. YOLOv5 has five different versions: YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. The difference between the five models is the depth and width of the network [24]. To select a suitable baseline from five different versions of the YOLOv5 model, we trained them on 1280 training sets of in-water abnormal fish. The training results are shown in Table 1, and the detection accuracy of YOLOv5s, YOLOv5m, YOLOv5l and YOLOv5x are nearly consistent. To balance the detection precision and model size for edge devices in actual scenarios, YOLOv5s is selected as the baseline to detect the abnormal surface features of fish.

Table 1. Comparison of evaluation metrics for the five YOLOv5 models.

Models	P (%)	R (%)	mAP_{50} (%)	$mAP_{50:95}$ (%)	Model Size (MB)	FLOPs (G)	FPS
YOLOv5n	0.911	0.917	0.922	0.574	3.8	4.1	101
YOLOv5s	0.949	0.945	0.964	0.661	14.3	15.8	87
YOLOv5m	0.949	0.948	0.962	0.679	42.1	47.9	76
YOLOv5l	0.948	0.946	0.964	0.675	92.7	107.6	68
YOLOv5x	0.951	0.944	0.959	0.688	173	203.8	61

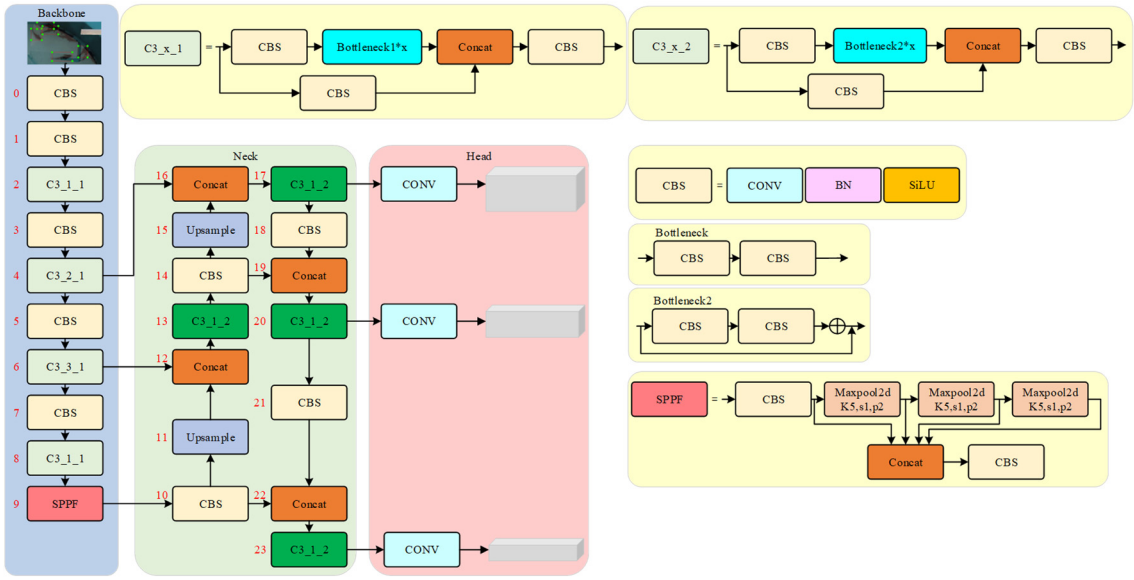

Figure 3. YOLOv5s network structure.

The network structure of YOLOv5s is illustrated in Figure 3. YOLOv5s encompassed four main parts: Input, Backbone, Neck and Head. Regarding the Input, YOLOv5s retains the mosaic data augmentation technique and adaptive image scaling in YOLOv4. Furthermore, YOLOv5s integrates the adaptive anchor boxes calculation into the program, enabling the selection of optimal anchor box values for different data sets. Compared to YOLOv4, several enhancements were introduced to the YOLOv5s Backbone, including the Focus module, CSPDarkNet53 module, and spatial pyramid pooling fast (SPPF) module. These additional modules expand the network's receptive field and further enhance its feature extraction capabilities. The CSPDarkNet53 module includes the CSPNet module, the Bottleneck

module and the C3 module. The Neck of YOLOv5s adopts the feature pyramid network (FPN) and path aggregation network (PAN) structures. FPN employed an up-bottom side connection to extract multi-scale features and construct the structure of feature pyramids. PAN added a bottom-up route and facilitated dense localization of high-level features. FPN and PAN aggregate parameters across different layers, raising the accuracy of object detection. At the Head, YOLOv5 generates multi-scale prediction results based on the outputs of the different Necks. The bounding box loss function is a complete intersection over union (CIoU) loss [25], which builds upon the generalized intersection over union (GIoU) loss by considering information about the position, scale and shape of the target boxes.

2.3. Improvements on YOLOv5s

The YOLOv5s is the smaller network structure among YOLOv5 family. It has obvious advantages in model size and detection speed compared with wider and deeper networks, but inevitably sacrifices detection accuracy. According to the definition of small targets in the COCO data set, small targets are with resolutions less than 32 (pixels) × 32 (pixels). The data set of this study are collected from the abnormal surface features of longsnout catfish in a recirculating aquaculture laboratory. In the process of data collection, certain inherent challenges such as blurred images, complex backgrounds and small targets have been identified. The performance of YOLOv5s is notably inadequate when applied to downstream tasks in this domain. As a result, several advanced methods were proposed for the vanilla YOLOv5s model, which consisted of four main parts:

1) Introduce a new NWD metric [26] and replace the CIoU loss function and NMS of YOLOv5s. We model the bounding boxes as 2D Gaussian distributions and compute their similarity using the NWD between the two distributions. NWD could enhance the detection ability of the model for small targets, optimize the convergence speed of the model and reduced the occurrence of false positives (FP).

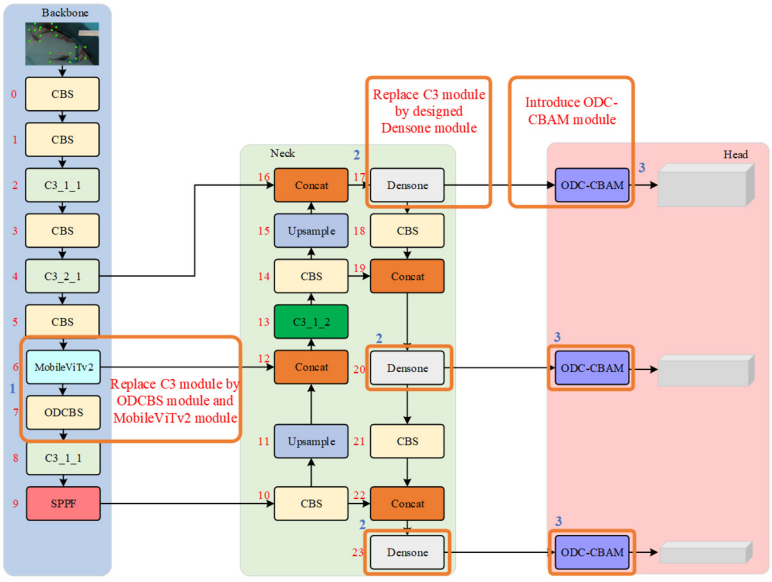

Figure 4. Improved YOLOv5s network structure.

2) The MobileViTv2 [27] module is utilized to replace the 6th layer network of the Backbone. This substitution elevates the features representation ability and computing efficiency of the model.

3) Supplant the C3 module of the PAN part with the designed DenseOne module. The DenseOne module is derived from DenseNet [28] and incorporates three additional 1 × 1 convolution operations. By establishing shortcut connections to reuse features, this reduces the model size and number of parameters.

4) The ODC-CBAM module is embedded into both the Backbone and PAN. By incorporating the principles of ACmix [29], the ODConv and CBAM are fused. The module incorporates ODConv [30], which adopts the convolutional kernel according to the shape and scale of the longsnout catfish dynamically. Simultaneously, the CBAM [31] assists the model in emphasizing the abnormal surface features of longsnout catfish while suppressing interference from complex backgrounds.

The improved YOLOv5s structure is shown in Figure 4.

2.3.1. NWD metric

In YOLOv5s, intersection over union (IoU) and its extensions are employed as evaluation metrics for the loss function and NMS. Nonetheless, IoU exhibits certain limitations and disadvantages in these applications, as outlined below:

1) The original model employs the CIoU loss function to compute the bounding box localization loss. This loss function extends the concept of IoU and incorporates three geometric properties: bounding box overlap area, centroid distance and aspect ratio. The calculation of the CIoU loss formulas is as follows:

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

$$\alpha = \frac{v}{(1-IoU+v)} \tag{2}$$

$$v = \frac{4}{\pi}\left(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h}\right) \tag{3}$$

where, IoU means the ratio of the intersection and union of the prediction bounding box and the actual bounding box. $\rho^2(b,b^{gt})$ represents the Euclidean distance between the centroids of the true and predicted boxes and c represents the diagonal distance of the minimum closed area that can cover both boxes. α is weight factor, v is a similarity ratio of length to width.

Equations (1)–(3) indicate that the CIoU loss function enhances the detection accuracy of the model by incorporating a penalty term based on the aspect ratio while calculating the predicted bounding boxes. Nevertheless, the CIoU loss function may exhibit reduced sensitivity when dealing with targets that possess extremely large or small aspect ratios. Consequently, this can lead to poor detection performance for small targets as the loss function may not adequately provide the necessary gradients for optimizing the network in these scenarios.

2) NMS is a widely employed post-processing technique in object detection. Its primary purpose is to suppress redundant predicted bounding boxes, ensuring that each object is associated with only the most accurate and optimal predicted bounding box. However, the selection of the IoU threshold greatly impacts the final detection result. If the threshold is set too high, there is a risk of erroneously rejecting small targets.

Hence, this investigation introduces the NWD metric and incorporates it into YOLOv5s by

replacing the CIoU loss function and NMS. To address the concentration of foreground and background pixels of the small longsnout catfish within the center and boundary of the bounding box, this study models the bounding box as a two-dimensional Gaussian distribution. The highest weight is assigned to the center pixel of the bounding box, gradually decreasing towards the border. The NWD metric is employed to assess the similarity between the modeled distribution and the actual pixel distribution, enabling a comprehensive evaluation of their likeness.

The object bounding box $R = (c_x, c_y, w, h)$ can be modeled as a two-dimensional Gaussian distribution $N(\mu, \Sigma)$. The NWD equation is shown in Eq (5). The NWD metric is shown in Eq (6).

$$\mu = \begin{bmatrix} c_x \\ c_y \end{bmatrix}, \Sigma = \begin{bmatrix} \frac{w^2}{4} & 0 \\ 0 & \frac{h^2}{4} \end{bmatrix} \tag{4}$$

$$W_2^2(N_a, N_b) = \left\| \left(\left[c_{x_a}\ c_{y_a}\ \frac{w_a}{2}\ \frac{h_a}{2} \right]^{\mathrm{T}}, \left[c_{x_b}\ c_{y_b}\ \frac{w_b}{2}\ \frac{h_b}{2} \right]^{\mathrm{T}} \right) \right\|_2^2 \tag{5}$$

$$NWD(N_a, N_b) = exp\left(-\frac{\sqrt{W_2^2(N_a, N_b)}}{C} \right) \tag{6}$$

where μ and Σ denote the mean vector and the co-variance matrix of the Gaussian distribution, and (c_x, c_y), w and h denote the center coordinates, width and height, respectively. $\|\cdot\|$ represents the Frobenius norm. C is a constant closely related to the data set, and we set C to 5 (the average absolute of our data set). In the detection of small target longsnout catfish, the NWD metric offers several advantages over the IoU:

1) Modeling the target bounding box as a two-dimensional Gaussian distribution presents a more effective approach for capturing the continuous and variable position deviation within the bounding box. Furthermore, by assigning weights and normalizing the pixels in different regions of the bounding box, we achieve improved performance. In comparison to IoU and its extension, the NWD method offers substantial advantages in terms of scale invariance and smoothness in handling position deviations.

2) By employing the NWD to measure the similarity between the predicted bounding box and the ground truth box, we can effectively address the issue of sensitivity in CIoU to small position deviations of the target. This approach proves beneficial even when there is no overlap or containment relationship between the two bounding boxes.

2.3.2. MobileViTv2

The C3 module, integrated into the backbone of YOLOv5s, serves as a convolutional neural network module specifically designed for feature extraction. It employs multiple convolutional kernels with varying scales to extract a more comprehensive range of feature information, thereby enhancing the model's ability to accurately detect objects of different sizes. However, the incorporation of the C3 module expands both the depth (number of convolutional layers) and width (number of channels) of the backbone network. Consequently, the model experiences a substantial increase in computational requirements due to the presence of multiple convolutional layers, resulting in higher latency during deployment on resource-constrained edge devices.

Mehta and Rastegari [27] proposed a light-weight and mobile-friendly hybrid network called

MobileViTv2. MobileViTv2 replaces the multi-head self-attention (MHA) mechanism utilized in MobileViTv1 [32] with a separable attention method. MobileViTv2 initially applies depth-wise separable convolution and a 1 × 1 convolutional layer to process the input feature map, facilitating the extraction of local information. It then employs a transformer module with separable attention to extract global information. The separable attention method computes the context score of the latent token L in the local features of the input. These scores are then reweighted for the input tokens and generate global information. The transformer module with separable self-attention is implemented by element-wise operation, which reduces the computational complexity. Lastly, the module incorporates a 1 × 1 convolutional layer to integrate local information, perform dimensional transformation, and merge features. The utilization of depth-wise separable convolutions allows for efficient capture of spatial information within the input feature maps while maintaining computational efficiency. The transformer module with separable self-attention is implemented by element-wise operation, which reduces the computational complexity. Therefore, this study utilizes the MobileViTv2 module to replace the C3 module in the sixth layer of the original model. This substitution aims to enhance the model's reasoning speed and alleviate the computational complexity resulting from feature extraction: Refer to Figure 5 for details.

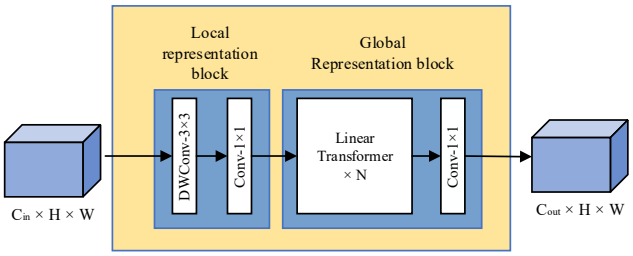

Figure 5. MobileViTv2 schematic.

2.3.3. DenseOne

Traditional convolutional networks with L layers have L connections (one connection between each layer and subsequent layer). DenseNet contains shortcut connections between input layers and output layers. DenseNet has $L(L+1)/2$ direct connections. The output of traditional convolutional networks at the L_{th} layer is shown in Eq (7). The output of DenseNet at the L_{th} layer is shown in Eq (8).

$$x_L = H_L(x_{L-1}) \tag{7}$$

$$x_L = H_L([x_0, x_{1,} \cdots, x_{L-1}]) \tag{8}$$

where $H(\bullet)$ is a non-linear transformation function and x_L is the L_{th} layer of the networks.

DenseNet enhances feature maps propagation by short connections, establishing direct connections between each layer and subsequent layers. This approach effectively improves the model's detection capability by encouraging the reuse of feature maps. Additionally, the input feature maps undergo processing through the transition layer, which includes a batch normalization layer, a 1 × 1 convolution layer and a 2 × 2 average pooling layer. Furthermore, a 1 × 1 convolution is applied in the bottleneck layer to reduce the dimensionality of the input feature map, resulting in a significant decrease in the number of parameters. Alongside improved parameter efficiency, DenseNet offers several advantages, including enhanced information flow and gradient propagation throughout the

entire network. Moreover, it serves as a regularization technique to address overfitting problems in downstream tasks, particularly when dealing with data sets that have limited samples. Figure 6(a) illustrates the details of the DenseNet.

We designed DenseOne based on the CSPNet [33] and DenseNet. First, feature extraction operations are performed on the input feature maps through two 1 × 1 convolutions. These operations help increase the gradient path of networks. Because of the CSP strategy, one could alleviate the disadvantages caused by using explicit feature map copy for concatenation. To balance the computation of the DenseNet, when the dimensionally reduced feature map is input to the subsequent DenseNet module, since the number of channels of the feature map becomes half of the original feature map, the computational bottleneck can be effectively reduced by nearly half. Then, the features of the first branch are reused in DenseB so that each layer in the network shares the global information in the feature map. In addition, the concat operation is used to perform channel merging of the feature maps of the two branches. Finally, a 1 × 1 convolution is used to recombine the connected features. Compared with DenseNet, DenseOne not only increases the gradient path, reducing the computational amount, but also shows more important feature expression capabilities. The structure diagram of DenseOne is shown in Figure 6(b).

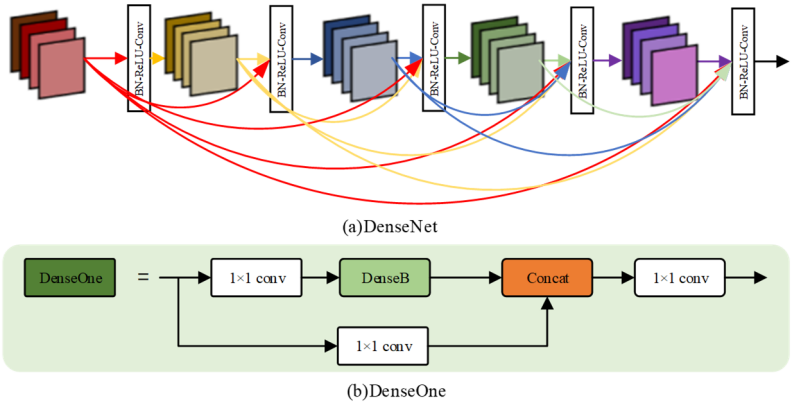

Figure 6. Schematic diagram of the DenseNet and DenseOne modules.

2.3.4. ODC-CBAM

Convolution and self-attention could enable the model to make more precise predictions, and they are usually considered as two peer approaches that are distinct from each other. ACmix is a mixed feature extractor that enjoys the benefit of both self-attention and convolution. For a detailed representation of the ACmix approach, refer to Figure 7.

ACmix could be divided into two stages. At stage I, ACmix implements 1 × 1 convolution operations on the input feature map, obtaining a rich set of intermediate features containing 3 × C feature maps (H × W × C → H × W × 3C, C stands for the number of channels, and H × W stands for the feature size.). At stage II, the intermediate feature maps are used through the convolution path and self-attention path. Because the convolution kernel size is k, the convolution path first utilizes a fully-connected (FC) layer to transform the number of channels to equal the number of all shift directions. Subsequently, features are generated via shifting and aggregation. In the self-attention path, they represent the features obtained in the stage I that are equally divided into queries, keys and values,

following the traditional multiheaded self-attention module. Finally, outputs from the convolution path and self-attention path are added together, the strengths are controlled by two learnable weights.

ACmix introduces 1 × 1 convolutions for the weight mapping part of the convolution and self-attention mechanisms to achieve correlation between the two at the underlying level. ACmix integrates the respective characteristics of convolution operations and self-attention while reducing computational overhead. However, the convolution in ACmix ignores the spatial information and channel information of the convolution kernel, making it difficult for the model to accurately fit features. Moreover, self-attention causes the model to converge too slowly and cannot quickly locate the regional location of useful features. Therefore, we employed the fusion of ODConv and CBAM to enhance the network's ability to learn deep features within complex factory farming environments.

The Dynamic Convolution achieves attention-based dynamic weighting of M parallel convolution kernels. The parallelism of M kernels not only maintains the network's width and depth, but also enhances its representation capability. However, existing dynamic convolutions overlook the other three dimensions of the convolutional kernel space: the size of each kernel's spatial dimension, the input channel number and the output channel number. To enable the model to learn more complex features, ODConv utilizes a novel multi-dimensional attention mechanism and parallel strategy to learn complementary attention for convolution kernels across all four dimensions of the kernel space.

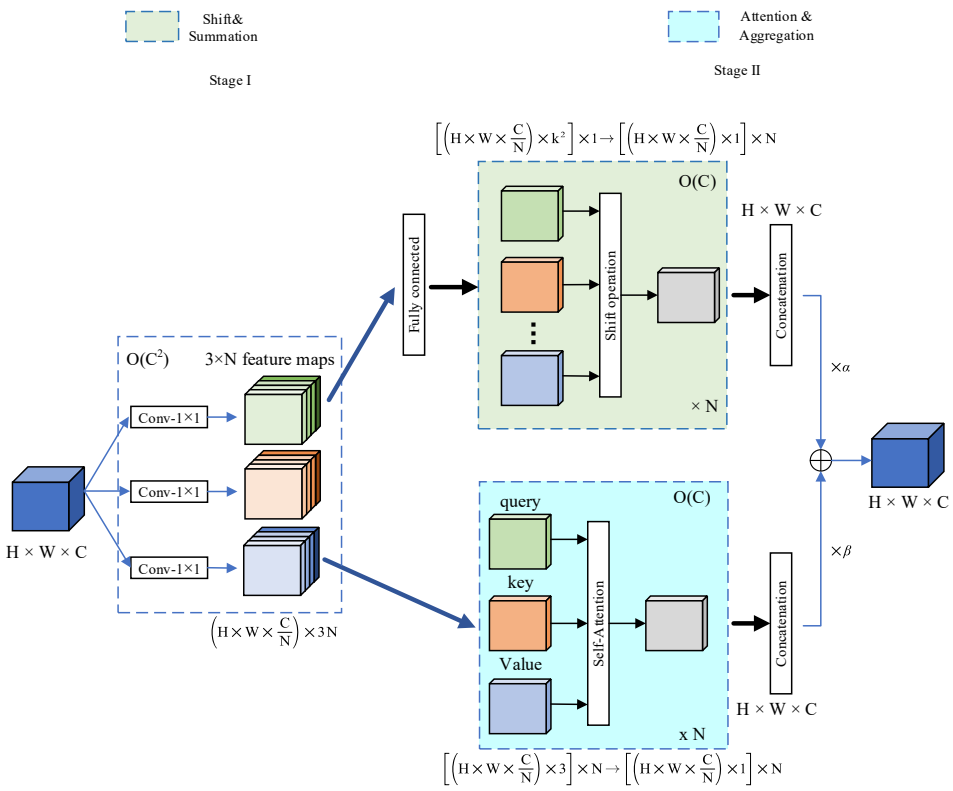

Figure 7. ACmix schematic.

The CBAM is a lightweight attention module that finds extensive usage in various convolutional neural networks. It comprises two components: The channel attention module (CAM) [34] and the spatial attention module (SAM) [35]. The CAM gathers valuable spatial information from feature maps through

average pooling and maximum pooling operations. It produces average pooling and maximum pooling features, which are then fed into a multi-layer perceptron (MLP) to generate key features consisting of multiple perceptual layers. Ultimately, channel attention maps are obtained. On the other hand, the SAM generates 2D spatial attention maps by applying average pooling and max pooling operations across channels. By leveraging the channel attention sub-module and the spatial self-attention module, CBAM dynamically learns and adjusts the weight distribution of channels and spatial dimensions in the feature map. This adaptive learning enhances the network's ability to express distinctive features effectively.

While ODConv and the CBAM attention module are often regarded as distinct paradigms, it has been demonstrated that the extensive calculations involved in both paradigms are essentially accomplished through the same operations. An ODConv with a kernel size of $k \times k$ can be devided into two stages: the first stage, where the ODConv is decomposed into k^2 individual 1×1 convolutions, and the second stage, which performs shifting and summation operations. Similarly, the CBAM attention module is also accomplished in two stages: the first stage projects the queries, keys, and values in the attention module into different 1×1 convolutions kernels, and the second stage calculates attention weights aggregately. Consequently, the first stages of both paradigms involve similar computational operations.

Therefore, there is a possibility of fusing the two paradigms: the ODConv module and the CBAM attention module. In this study, the fusion weight is 0.5 for both the ODConv and CBAM attention modules. The ODC-CBAM module is less computationally complex than the pure convolution or attention mechanism, and it can obtain better performance than both paradigms. Therefore, we used the ODC-CBAM module before embedding it into the output of YOLOv5s to improve the model's ability to identify difficult samples and enhance the detection accuracy. In this study, the ODC-CBAM module was used to replace the conventional convolution module in the original model, and the results are shown in Table 2. The best evaluation metric achieved was with Replacement4 (Replacement4 is the ODC-CBAM the replacement layer 7 of the backbone and the regular convolution in front of the Head).

Table 2. Experimental results of the ODC-CBAM model at different locations.

Models	P (%)	R (%)	mAP_{50} (%)	$mAP_{50:95}$ (%)	Model Size(MB)
Replacement1	0.986	0.984	0.993	0.73	15.1
Replacement2	0.982	0.986	0.991	0.733	14.7
Replacement3	0.977	0.995	0.989	0.721	14.5
Replacement4	0.989	0.990	0.993	0.741	14.2

Note: Replacement 1 is ODC-CBAM replacing the regular convolution of layers 1, 3, 5 and 7 of the Backbone and the front of the Head. Replacement 2 is ODC-CBAM replacing the regular convolution of layers 3, 5 and 7 of the Backbone and the front of the Head. Replacement 3 is ODC-CBAM replacing the regular convolution of layers 5 and 7 of the Backbone and the front of the Head. Replacement 4 is ODC-CBAM replacing layer 7 of the Backbone and the regular convolution in front of the Head.

3. Datasets and experiments details

3.1. Datasets

3.1.1. Data acquisition

The experimental data were collected from December 15th to December 22nd, 2022, at the Genetic Breeding Center for Longsnout Catfish of the Agriculture and Rural Ministry Affairs in Pudong New Area,

Shanghai. The experimental fish species used in this study are diseased longsnout catfish, provided by the College of Fisheries and Life Science at Shanghai Ocean University. The experimental fish comprise 20 individuals with a weight range of 50–100g and a body length of 10–15 cm. The water temperature in the aquaculture environment is maintained at (25 ± 1) ℃, with a dissolved oxygen level of (5 ± 0.3) mg/L and pH value of (7 ± 0.5). When the disease occurs in the longsnout catfish, there are fewer white spots on the surface of the longsnout catfish in the early stages of the disease. Over time, the longsnout catfish develop large areas of abnormal surface features. A fish image acquisition system is developed to obtain raw data more efficiently, which simultaneously captured in-water images. As shown in Figure 8, this system consists of a BARLUS camera (S97K8F-8D6X10), a support bracket, and a circular fish-rearing tank. The rearing tank has a radius of 76cm, a height of 85.5 cm and a water depth of 40 cm. The BARLUS camera captures 24-bit RGB true-color images with a resolution of 3840 (pixels) × 2160 (pixels) and a frame rate of 60 FPS.

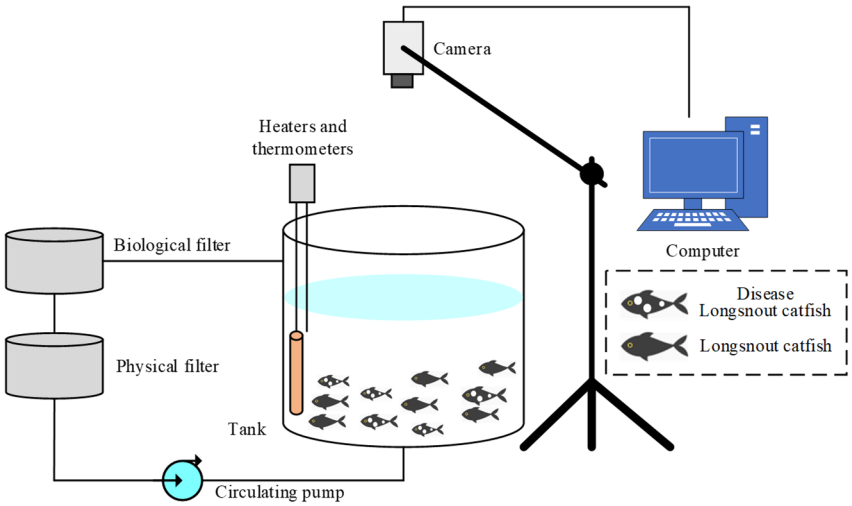

Figure 8. Schematic diagram of the fish image acquisition system.

3.1.2. Dataset for improved YOLOv5s

The College of Fisheries and Life Science at Shanghai Ocean University manually screened the videos, resulting in 38 segments containing abnormal surface features of longsnout catfish. The average duration of each segment is approximately 9 seconds. The initial step of this study involves reading each frame from the 38 video segments collected at the Genetic Breeding Center for Longsnout Catfish of the Agriculture and Rural Ministry Affairs in Pudong New Area, Shanghai. Then, every five frames are sampled to extract one frame, yielding 4104 images of abnormal longsnout catfish. We employ the structural similarity index (SSIM) algorithm [36] to further screen the original dataset obtained from the video streams. This screening aimed to eliminate redundant and noisy images. The SSIM algorithm assesses the similarity of a pair of images based on three main image features: luminance, contrast and structure. It computes a comprehensive SSIM index by weighting and summing these similarity measurements. One of the advantages of SSIM is its consideration of structural information in images, making it robust against lighting variations, noise and distortions. The computation formula for SSIM is shown in Eq (9).

$$SSIM(X,Y) = \frac{(2\mu_X\mu_Y+C_1)(2\sigma_{XY}+C_2)}{(\mu_X^2+\mu_Y^2+C_1)(\sigma_X^2+\sigma_Y^2+C_2)} \tag{9}$$

where $SSIM(X,Y)$ is a metric used to measure the similarity between images X and Y. μ_X and μ_Y denote the mean values of images X and Y, respectively, and their standard deviations are represented by σ_X and σ_Y. The covariance of X and Y is represented by σ_{XY}. The values of C_1 and C_2 are constants that can be arbitrarily set.

After applying the SSIM algorithm for screening, the final dataset consists of 1600 surface images of longsnout catfish (in-water) images from video streams. 1600 surface images of longsnout catfish (in-water) images in the video stream were manually annotated using LabelImg. Our labeling principle for the in-water data set is: annotate the entire fish. The annotations generated XML files containing coordinate information, image size, 27200 annotated bounding boxes and label name (disease). These XML files were saved in the VOC2007 dataset format, creating the abnormal longsnout catfish in-water dataset, which was utilized in this experiment.

To split the datasets for training, validation and testing, we followed an 8:1:1 ratio, and the training set and test set cannot come from the same video sequence. Thus, each dataset was divided into 1280 images for the training set and 160 for the validation and testing sets. Some sample images from this study dataset are shown in Figure 9. As observed from Figure 9, the abnormal longsnout catfish in-water dataset presents some challenging samples which pose some difficulties in detecting abnormal fish. The specific challenges are as follows:

1) Pixel blur: The fast swimming speed of longsnout catfish poses a challenge of accurate target capture.

2) Serious overlap: Due to the biological characteristic of longsnout catfish liking to gather in groups, the targets being tested in the collected images are heavily overlapped.

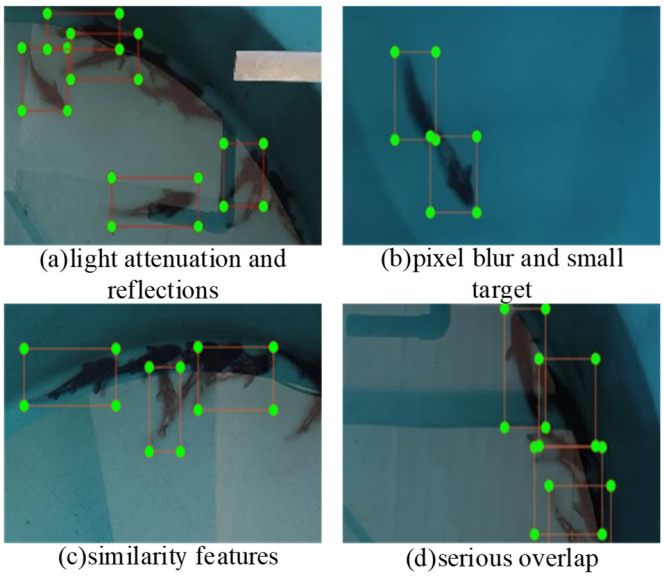

Figure 9. Some example images from this study dataset.

3) Similarity between features: In the early stages, the features of abnormal longsnout catfish are similar to those of healthy longsnout catfish, making it necessary for the model to have a strong feature learning ability.

4) Small target: Most abnormal longsnout catfish are young fish that are far away from the image collection system, making them small or tiny targets with few pixels and insufficient features. Therefore, the model's ability to detect small targets needs improvement.

5) Light attenuation: When longsnout catfish are disturbed, the mucous cells on their surface secrete a large amount of acidic mucus, causing the aquaculture water to become a gel-like substance, which leads to a certain attenuation of the light reflected to the camera. This makes it more difficult to identify surface features.

6) External interference: There are uncontrollable factors, such as the operation of motors in the circulating water aquaculture laboratory where longsnout catfish are raised, causing slight water surface fluctuations in the aquaculture tanks. As a result, the collected images may contain phenomena such as reflections and inverted images.

3.2. Experiment platform and training hyperparameters

In this paper, the improved model is experimented on a deep learning server with the configuration shown in Table 3.

Table 3. Deep learning server configuration.

Configuration	Parameter
CPU	Inter(R) Xeon(R) W-2223
GPU	Nvidia GeForce RTX 2080ti
Operating system	Windows10
Accelerated environment	CUDA11.7 and Cudnn8.0.5
Interpreter setting	Python3.8 and torch1.13.1

Some of the training hyperparameters of the improved model are: the input image size is 640 × 640, the optimizer is SGD with decay and momentum of 0.937, Warming-up strategy, learning rate decay, L2 regularization and data preprocessing techniques are used in the training process. The maximum learning rate is 0.01 and gradually decreases. The batch size is 16 in order to reduce the computing pressure, with a total of 500 epochs of training. The training hyperparameters are shown in the Table 4.

Table 4. Training parameter.

Parameter	Value
Image size	640 × 640
Optimizer	SGD
Learning rate	0.01
Momentum	0.937
Epoch	500

3.3. Model evaluation

The aim of this study is to develop an abnormal longsnout catfish surface feature detection model

that balances both detection accuracy and speed. Mean average precision (mAP) is a commonly used evaluation metric in object detection models. It is calculated based on the Precision-Recall (PR) curve, which is composed of precision and recall [37]. mAP_{50} and $mAP_{50:95}$ can comprehensively evaluate the model's ability to detect targets of different sizes and shapes and more objectively reflect the accuracy of the model. Correspondingly, four indexes are used to evaluate the accuracy of the model: precision, recall, mAP_{50}, and $mAP_{50:95}$.

FPS is the number of detected frames per second, and an FPS of 30 is sufficient for real-time detection. For practical applications in the field of aquaculture, real-time detection of abnormal fish is very important. FPS, as one of the performance evaluation indicators, can show the advantage of the model in processing speed. The formulae for precision (P), recall (R), mAP_{50} and $mAP_{50:95}$ are shown as:

$$P = \frac{TP}{TP+FP} \tag{10}$$

$$R = \frac{TP}{FP+FN} \tag{11}$$

$$AP = \int_0^1 P(R)\, dR \tag{12}$$

$$\text{mAP} = \sum_{i=1}^{k} \frac{AP}{k} \tag{13}$$

where TP is the number of true positive samples; FP is the number of false positive samples; FN is the number of false negative samples; AP is the average precision of a category; and k is the number of categories. The difference between mAP_{50} and $mAP_{50:95}$: mAP_{50} refers to the average AP at an IoU threshold of 0.5, and $mAP_{50:95}$ refers to the average AP over a range of IoU thresholds, typically from 0.5 to 0.95, in steps of 0.05.

4. Experimental results and analysis

4.1. Training result analysis

This study's training results are shown in Figure 10. From Figure 10(a), it can be seen that the precision of the model rose rapidly to 95.53% within the initial 70 training epochs. Subsequently, the precision reached a stable level of approximately 99.5% as the training progressed. Examining Figure 10(b), it can be observed that the recall of the model demonstrated a swift increase of 0.976 within the first 50 epochs of training. With further training, the model's recall stabilized at around 99.3%. Figure 10(c) displays a notable trend in the loss value, wherein a significant decrease occurred within the initial 50 epochs of training, followed by a stabilized pattern after 300 epochs. The training results from Figure 10 demonstrate that the enhanced model performs well in abnormal surface features of longsnout catfish detection. The decreasing loss function indicates that the model has reached a state of convergence. After calculating the timestamp function, our model trains for about 80 seconds for 1 epoch, and the training time for 500 rounds is about 11.1 hours, which allows us to optimize the training time of the model based on the specific needs of actual applications. This time frame may vary depending on the complexity of the model, the size of the data set and the computing resources used.

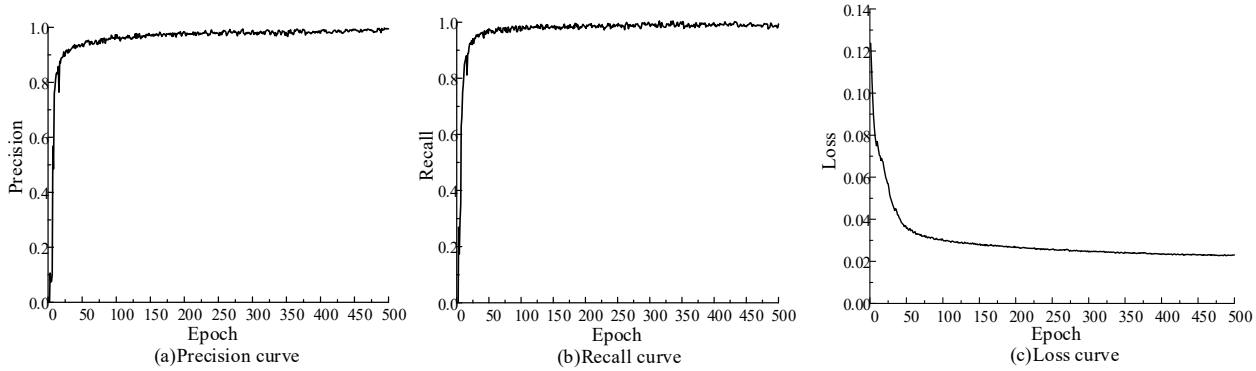

Figure 10. Training results graph (abnormal surface of longsnout catfish in-water dataset).

4.2. Algorithm performance evaluation

4.2.1. Ablation study

To assess the overall performance of the enhanced model, this study conducts specific ablation experiments on each component of the improvement and analyzes their respective effects. It is crucial to ensure that the ablation experiments are conducted using the same data set and hyperparameters. The training results are presented in Table 5. From the table, it can be observed that the model's performance can be improved by employing the NWD metric, DenseOne module, ODC-CBAM module and MobileViTv2 module individually. Notably, the ODC-CBAM module exhibits the most significant impact, surpassing the baseline mAP_{50} and $mAP_{50:95}$ by 2.1 and 3.2%, respectively, with minimal increase in parameters. This can be attributed to the ODC-CBAM module's integration of ODConv and CBAM modules based on the ACmix principle, which leverages the strengths of both modules. As a result, it can effectively learn useful features from complex backgrounds and suppress irrelevant background features, thereby enhancing the model's capability to represent features, especially for challenging samples. The DenseOne module achieves parameter reduction while improving the detection accuracy of the model. Compared to the baseline (Model 1), our proposed model (Model 8) achieves a maximum increase of 2.9% in mAP_{50} on the abnormal longsnout catfish dataset, with a remarkable growth of 12.25% in $mAP_{50:95}$. However, it is important to note that the FPS significantly decreases after reaching the highest $mAP_{50:95}$ value. In the detection of abnormal longsnout catfish, faster reasoning speed facilitates timely identification of affected specimens, reducing unnecessary economic losses and environmental pollution. MobileViTv2 utilizes depth-separable convolution, separable self-attention and element-wise operation to improve the inference speed of the model. Notably, our method maintains nearly the same detection speed as the baseline while comprehensively enhancing the detection accuracy of abnormal longsnout catfish, albeit with a slight increase in parameters (parameters are usually included as part of the memory access and do not affect the inference speed of the model), model size and GFLOPs.

Table 5. Comparison of model training evaluation metrics.

Models	**1**	**2**	**3**	**4**	Parameters	mAP_{50} (%)	$mAP_{50:95}$ (%)	Models Size (MB)	FLOPs (G)	FPS
1	×	×	×	×	7022326	0.964	0.661	14.3	15.8	87
2	√	×	×	×	7022326	0.987	0.676	14.0	14.9	85
3	×	√	×	×	6965935	0.985	0.680	14.3	15.4	67
4	×	×	√	×	7074005	0.987	0.693	14.5	15.1	41
5	×	×	×	√	7361878	0.980	0.685	15.1	16.9	77
6	√	√	×	×	6965935	0.992	0.707	14.3	15.3	66
7	√	√	√	×	7021301	0.993	0.742	15.0	16.6	39
8	√	√	√	√	7369077	0.993	0.741	15.2	16.2	88

Note: "**1**" represents NWD improvement. "**2**" represents DenseOne improvement. "**3**" represents ODC-CBAM improvement. "**4**" represents MobileViTv2 improvement. "×" representatives do not introduce this improvement strategy. "√" representatives introduce this improvement strategy.

4.2.2. Algorithm performance

To verify the detection performance of the enhanced model for abnormal longsnout catfish, a predetermined test set was utilized to input both the pre-improvement and post-improvement models. The visualization of model detection results, as depicted in Figure 10, provides insights into the comparison. Observing Figure 11(a) and (b), the baseline model is prone to missed detection. In contrast, the improved model tackles this issue by substituting the original model's NMS and CIoU loss functions with the NWD metric. By employing two-dimensional Gaussian modeling on the target bounding box and utilizing the normalized Wasserstein distance, the NWD metric calculates similarity and effectively eliminates sensitivity to small deviations in object position based on IoU and its extensions. Figure 11(c) and (d) visually demonstrate the superiority of the improved model over YOLOv5s. This notable improvement can be attributed to the challenges posed by uncontrollable factors such as pixel blur resulting from the high-speed motion of the longsnout catfish and complex backgrounds involving lighting and reflection. Introducing the ODC-CBAM at the backbone network and the front end of Head allows for the extraction of valuable abnormal surface features from complex backgrounds and suppresses the interference of useless features, such as the background. Moreover, replacing the C3 module in PAN with DenseOne enhances feature reuse, facilitates feature propagation and augments feature expression capability. Consequently, our proposed method not only enhances the detection confidence score for abnormal longsnout catfish, but also effectively mitigates false detection issues.

4.2.3. Model evaluation on validation set

In order to better evaluate the feasibility of the improved model in the field of aquaculture, we performed statistical analysis on the validation results. Confidence threshold (Conf_thres) is 0.001, IoU threshold (IoU_thres) is 0.6 and batch-size is 32. The results are shown in Table 6. It is worth noting that the improved model precision, recall, mAP_{50} and $mAP_{50:95}$ have increased by 1.4, 1.2, 3.2 and 8.2%, respectively. Our method has extremely low missed detection rate and false detection rate compared with the baseline. Our inference speed increased by 1 FPS compared with the baseline, while

the model weight size only increased by 0.9M. The model we proposed is suitable for the detection of fish with abnormal surface features in real aquaculture and is easy to deploy to edge devices and web interface development.

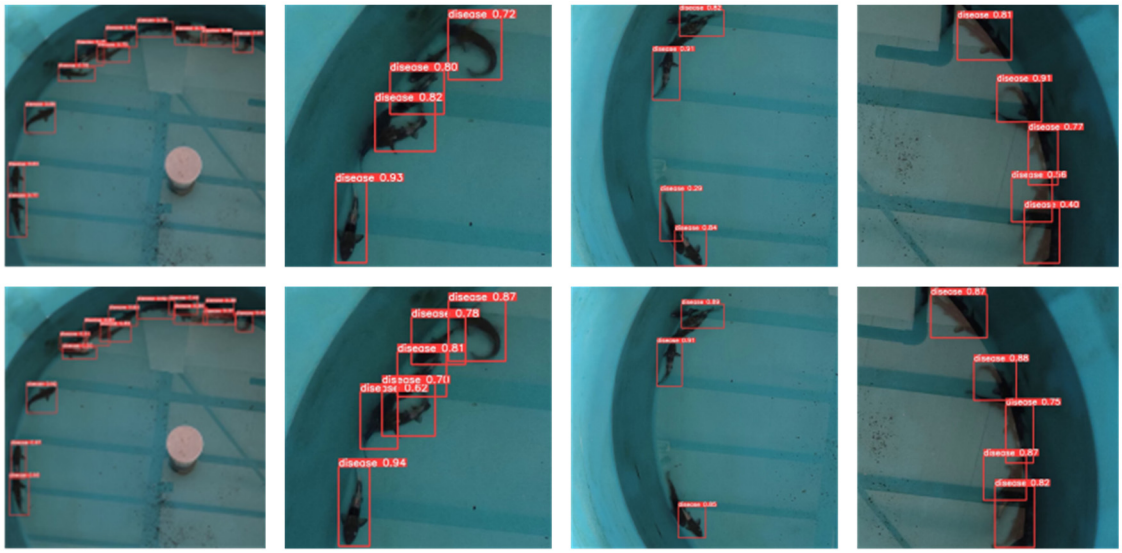

Figure 11. Model inference results before and after improvement. The first line is the inference result of YOLOv5s; The second line is the inference result of our method.

Table 6. Validation set experimental results.

Models	P (%)	R (%)	mAP_{50} (%)	$mAP_{50:95}$ (%)	Model Size (MB)	FPS
YOLOv5s	98.1	97.9	95.9	65.7	14.3	87
improved YOLOv5s	99.5	99.1	99.1	73.9	15.2	88

As can be seen from Figure 12, the improved model result in FP when facing extremely small targets. This is because the IoU between the predicted box and the real box is less than the threshold set by the model, which leads to missed detection. In addition, our model is prone to false positive problems when faced with severe occlusion between targets.

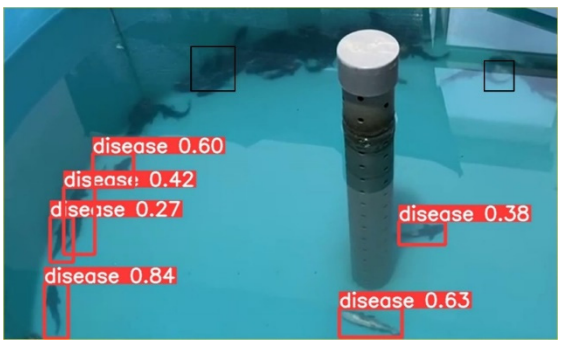

Figure 12. Examples of missed detections and false detections in the validation set.

4.2.4. Grad-CAM visualization results

Gradient-weighted class activation mapping (Grad-CAM) [38] is a technique that enhances model interpretability by visualizing the input regions crucial for predictions, providing visual explanations without requiring architectural modifications or retraining. In this study, two random images of abnormal longsnout catfish from the test set were selected to generate visualized heat maps using the Grad-CAM method for the YOLOv5s before and after the enhancement. The results are presented in Figure 13 (In the color spectrum, regions closer to blue indicate a lower proportion of features, while redder regions denote a higher proportion. A higher feature proportion implies greater importance in detecting abnormal longsnout catfish.). Figure 13 reveals that the red areas in the original model mainly correspond to the healthy parts and the background of the abnormal fish, whereas the improved model precisely identifies the abnormal surface features of the longsnout catfish.

The introduction of the ODC-CBAM in the backbone allowed for the extraction of important features related to the abnormal surface features of longsnout catfish in terms of the convolutional kernel space, input or output channels and more, inhibiting the learning of background and healthy part features. Additionally, the integration of MobileViTv2 into the Backbone facilitated the extraction and integration of local and global information from the features of the abnormal longsnout catfish, resulting in more comprehensive feature extraction. In the PAN part, the C3 module was replaced with the DenseOne module, enhancing feature reuse through short connections and enabling the model to learn a more complete information flow. As a result, the improved model exhibits improved accuracy in detecting abnormal longsnout catfish and mitigates the interference caused by complex backgrounds and other uncontrollable factors.

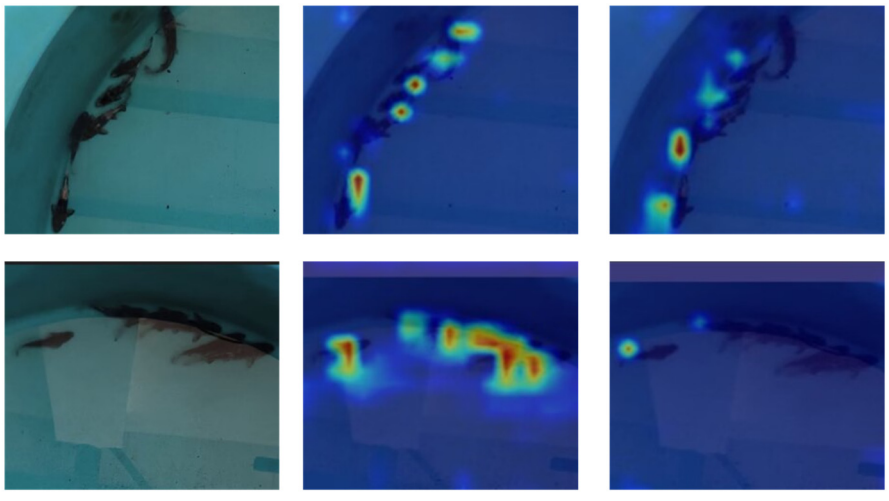

Figure 13. Grad-CAM visualization results. The first column is two randomly selected test set images; the second column is the visualization result of the improved model; the third column is the visualization result of the baseline.

4.2.5. State-of-the-art models' performance comparison

To validate the superior performance of our proposed method on the abnormal longsnout catfish

dataset, we conducted comparative experiments with the state-of-the-art methods, including mainstream one-stage object detection methods: YOLOv4, YOLOv5, YOLOv7, YOLOv8, SSD and the mainstream two-stage object detection algorithm Faster R-CNN. These comparative experiments were meticulously carried out under identical hardware environments, datasets, hyperparameters and training epochs. The results of the comparison are presented in Table 7, while Figure 14 illustrates the comparison of the PR curves of the seven algorithms.

The area surrounded by the PR curve reflects the algorithm's performance. From Figure 14 and Table 7, it is evident that our proposed method surpasses other models in the downstream task of abnormal longsnout catfish surface features detection. First, Table 7 reveals that the evaluation metrics of YOLOv4 and SSD are not excellent. Compared with the one-stage object detection methods SSD and YOLOv4, Faster R-CNN has a certain increase in mAP_{50} and $mAP_{50:95}$, but it is not good with detection speed. Because the one-stage target detection algorithm has a faster detection speed, its detection accuracy is inferior to the two-stage target detection algorithm. YOLOv7 and YOLOv8 fail to exhibit superior performance compared to the baseline, and their models' large number of parameters, big model size and GFLOPs make them unsuitable for deployment on resource-constrained IoT devices. Notably, the parameters, model size and FLOPs of our proposed model are 7.36M, 15.2MB and 16.4G, respectively, ranking second only to the baseline. This proves that our method does not require expensive hardware device support. In the comparative experiments, our method outperforms other algorithms in terms of detection accuracy, with an impressive mAP_{50} reaching 99.3%. Finally, in terms of FPS, the improved algorithm achieves a remarkable 88 FPS, meeting the real-time detection needs in factory farming and outperforming the baseline by 1 FPS, thus surpassing other algorithms in detection speed.

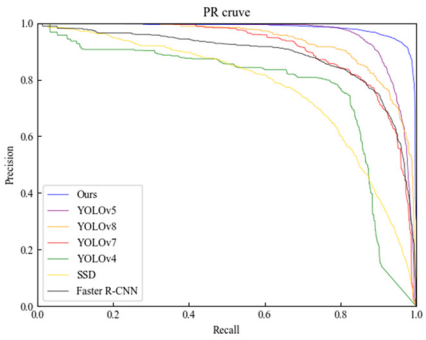

Figure 14. PR cruve graph.

Table 7. Experimental results of the comparison of the seven models.

Models	Parameters	mAP_{50} (%)	$mAP_{50:95}$ (%)	Model Size (MB)	FLOPs (G)	FPS
YOLOv4	52.49	0.776	0.571	105.4	118.9	16
YOLOv7	37.19	0.914	0.636	291.4	103.2	34
YOLOv8	11.13	0.921	0.642	22.5	28.4	43
YOLOv5s	7.02	0.964	0.661	14.3	15.8	87
Ours	7.36	0.993	0.741	15.2	16.4	88
SSD	23.61	0.770	0.556	90.6	273.74	12
Faster R-CNN	28.05	0.881	0.597	108.0	947.28	8

We visually compare the proposed method with the six mainstream target detection models mentioned above, as shown in the Figure 15. As can be seen from Figure 13, YOLOv8 and YOLOv7 have good detection capabilities for individuals with abnormal surface features. However, there is one false positive, two missed detections and one false detection in the detection of aggregated fish with abnormal surface features. This shows that the ability of the two to distinguish background interference is weak, and they cannot accurately allocate prediction frames to the aggregated abnormal surface features of longsnout catfish. As a two-stage target detection algorithm, Faster R-CNN has higher detection accuracy than the one-stage algorithms SSD and YOLOv4, but these three have poor recognition capabilities for fish with abnormal surface features in complex scenes. Our proposed method not only leads other models in confidence scores and has extremely low miss and false detection rates, but it also has higher precision and recall for dense small targets in low-light and background distractor environments.

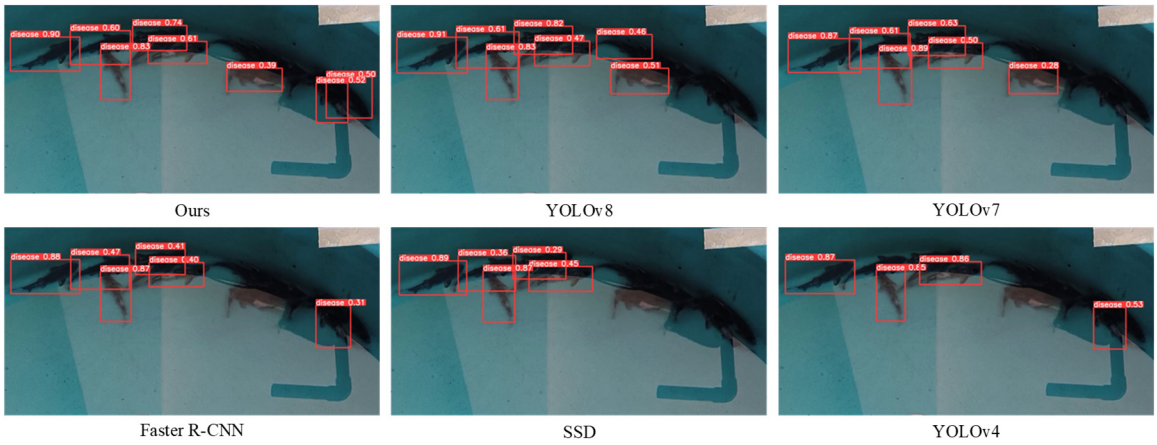

Figure 15. Comparison of different model inference results.

5. Conclusions

This study proposes an improved YOLOv5s target detection model for the automatic monitoring of abnormal surface features of fish. Compared with previous manual detection methods, our model is not affected by factors such as emotion, fatigue or subjectivity. It avoids the impact of individual differences or supervisor bias on detection results, can process large amounts of data in a shorter time and provides more consistent detection results. In the model, we introduce a notable enhancement by substituting the CIoU loss function and NMS with the NWD metric. This improvement aims to enhance the model's ability to detect small targets and speed up convergence speed of the model. The MobileViTv2 module is added to the Backbone to improve the feature representation ability and computing efficiency of the model. In addition, we design the DenseOne module to improve detection accuracy while reducing the model size and parameters for edge devices. Based on the above improvements, the ODC-CBAM modules are integrated into the Backbone and the PAN part of the network, which reduces the missed detection rate and false detection rate of abnormal surface features located in complex scenes. The improved model was evaluated on the validation set, with a precision of 99.5% and a recall of 99.3%, which are 1.4 and 1.2% higher than the baseline respectively. While the inference speed is increased by 1 FPS, the model size is only increased by 0.9M, achieving a

balance between model detection speed, model size and detection accuracy.

The experimental results show that the proposed model can be quickly and effectively used to detect abnormal surface features of fish, but it also has certain limitations: 1) Single data type: The object of study in this study is only one kind of fish with abnormal surface features, and no related experiments were carried out on other fish with abnormal surface feature. The model effect still needs to be verified in future work. 2) Fish density is fixed: The paper did not verify the model effect in a high-density breeding scenario. 3) Schools of fish lack minimal targets: In scenarios where there is an extreme lack of abnormal surface feature information, model performance may be poor. Therefore, in future work we will conduct in-depth research on the problem of abnormal fish detection for different fish or aquaculture scenarios. Moreover, we will combine multi-modality and transfer learning and construct different abnormal fish surface feature datasets to solve various downstream tasks.

Although our model has certain shortcomings, we can quickly and accurately detect abnormal surface features of longsnout catfish in the water, and the model size and parameters are relatively lightweight. We can deploy this model to embedded devices and web platforms. Therefore, this study provides new ideas for the realization of smart aquaculture.

6. Declaration of ethical considerations of computer vision in aquaculture

Fish Welfare: We recognize that any technology that affects fish health and welfare needs to be treated with caution. Our research aims to help monitor abnormal fish species to improve farming efficiency, but we will also highlight the need to ensure the impact of these techniques on fish species is minimized.

Privacy Issues: We take the protection of fish privacy seriously when it comes to data collection and image processing. We try to minimize disruption and impact on individual fish and take steps to ensure data security and privacy protection.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work was supported in part by the National Key R&D Program of China (grant number: 2023YFD2401304), Shanghai Collaborative Innovation Center for Cultivating Elite Breeds and Green-culture of Aquaculture animals (grant number: 2021-KJ-02-12) and the Shanghai Chongming Agricultural Science and Technology Innovation Project (grant number: 2021CNKC-05-06).

Conflict of interest

The authors declare there is no conflict of interest.

References

1. E. A. O'Neil, N. J. Rowan, A. M. Fogarty, Novel use of the alga Pseudokirchneriella subcapitata, as an early-warning indicator to identify climate change ambiguity in aquatic environments using freshwater finfish farming as a case study, *Sci. Total Environ.*, **692** (2019), 209–218. https://doi.org/10.1016/j.scitotenv.2019.07.243
2. Y. Wei, Q. Wei, D. An, Intelligent monitoring and control technologies of open sea cage culture: A review, *Comput. Electron. Agric.*, **169** (2020), 105119. https://doi.org/10.1016/j.compag.2019.105119
3. S. Zhao, S. Zhang, J. Liu, H. Wang, D. Li, R. Zhao, Application of machine learning in intelligent fish aquaculture: A review, *Aquaculture*, **540** (2021), 736724. https://doi.org/10.1016/j.aquaculture.2021.736724
4. C. Liu, Z. Wang, Y. Li, Z. Zhang, J. Li, C. Xu, et al., Research progress of computer vision technology in abnormal fish detection, *Aquacultural Eng.*, **103** (2023), 102350. https://doi.org/10.1016/j.aquaeng.2023.102350
5. Y. Zhou, J. Yang, A. Tolba, F. Alqahtani, X. Qi, Y. Shen, A data-driven intelligent management scheme for digital industrial aquaculture based on multi-object deep neural network, *Math. Biosci. Eng.*, **20** (2023), 10428–10443. https://doi.org/10.3934/mbe.2023458
6. L. Zhang, B. Li, X. Sun, Q. Hong, Q. L. Duan, Intelligent fish feeding based on machine vision: A review, *Biosyst. Eng.*, **231** (2023), 133–164. https://doi.org/10.1016/j.biosystemseng.2023.05.010
7. B. Zion, The use of computer vision technologies in aquaculture-A review, *Comput. Electron. Agric.*, **88** (2012), 125–132. https://doi.org/10.1016/j.compag.2012.07.010
8. M. L. Yasruddin, M. A. H. Ismail, Z. Husin, W. K. Tan, Feasibility study of fish disease detection using computer vision and deep convolutional neural network (DCNN) algorithm, in *2022 IEEE 18th International Colloquium on Signal Processing & Applications (CSPA)*, (2022), 272–276. https://doi.org/10.1109/CSPA55076.2022.9782020
9. A. Ashraf, A. Atia, Comparative study between transfer learning models to detect shrimp diseases, in *2021 16th International Conference on Computer Engineering and Systems (ICCES)*, (2021), 1–6. https://doi.org/10.1109/ICCES54031.2021.9686116
10. Q. Wang, C. Qian, P. Nie, M. Ye, Rapid detection of Penaeus vannamei diseases via an improved LeNet, *Aquacultural Eng.*, **100** (2023), 102296. https://doi.org/10.1016/j.aquaeng.2022.102296
11. J. C. Chen, T. Chen, H. Wang, P. Chang, Underwater abnormal classification system based on deep learning: A case study on aquaculture fish farm in Taiwan, **99** (2022), 102290. https://doi.org/10.1016/j.aquaeng.2022.102290
12. A. Gupta, E. Bringsdal, K. M. Knausgard, M. Goodwin, Accurate wound and lice detection in atlantic salmon fish using a convolutional neural network, *Fishes*, **7** (2022), 345. https://doi.org/10.3390/fishes7060345
13. J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2016), 779–778. https://doi.org/10.1109/CVPR.2016.91
14. C. Chen, G. Yuan, H. Zhou, Y. Ma, Improved YOLOv5s model for key components detection of power transmission lines, *Math. Biosci. Eng.*, **20** (2023), 7738–7760. https://doi.org/10.3934/mbe.2023334

15. Y. Ma, G. Yuan, K. Yue, H. Zhou, CJS-YOLOv5n: A high-performance detection model for cigarette appearance defects, *Math. Biosci. Eng.*, **20** (2023), 17886–17904. https://doi.org/10.3934/mbe.2023795
16. A. Bochkovskiy, C. Wang, H. M. Liao, YOLOv4: Optimal speed and accuracy of object detection, preprint, arXiv:2004.10934.
17. C. Li, L. Li, H. Jiang, K. Weng. Y. Geng, L. Li, et al., YOLOv6: A single-stage object detection framework for industrial applications, preprint, arXiv:2209.02976.
18. C. Wang, A. Bochkovskiy, H. M. Liao, Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2023), 7464–7475. https://doi.org/10.1109/CVPR52729.2023.00721
19. G. Yu, J. Zhang, A. Chen, R. Wan, Detection and identification of fish skin health status referring to four common diseases based on improved YOLOv4 model, *Fishes*, **8** (2023), 186. https://doi.org/10.3390/fishes8040186
20. Z. Wang, H. Liu, G. Zhang, X. Yang, L. Wen, W. Zhao, Diseased fish detection in the underwater environment using an improved YOLOV5 network for intensive aquaculture, *Fishes*, **8** (2023), 169. https://doi.org/10.3390/fishes8030169
21. E. Prasetyo, N. Suciati, C. Fatichah, Yolov4-tiny with wing convolution layer for detecting fish body part, *Comput. Electron. Agric.*, **198** (2022), 107023. https://doi.org/10.1016/j.compag.2022.107023
22. S. Zhao, S. Zhang, J. Lu, H. Wang, Y. Feng, C. Shi, et al., A lightweight dead fish detection method based on deformable convolution and YOLOV4, *Comput. Electron. Agric.*, **198** (2022), 107098. https://doi.org/10.1016/j.compag.2022.107098
23. X. Li, Y. Hao, P. Zhang, M. Akhter, D. Li, A novel automatic detection method for abnormal behavior of single fish using image fusion, *Comput. Electron. Agric.*, **203** (2022), 107435. https://doi.org/10.1016/j.compag.2022.107435
24. P. Jiang, D. Ergu, F. Liu, Y. Cai, B. Ma, A Review of Yolo algorithm developments, *Proc. Comput. Sci.*, **199** (2022), 1066–1073. https://doi.org/10.1016/j.procs.2022.01.135
25. Z. Zheng, P. Wang, D. Ren, W. Liu, R. Ye, Q. Hu, et al., Enhancing geometric factors in model learning and inference for object detection and instance segmentation, *IEEE Trans. Cybern.*, **52** (2022), 8574–8586. https://doi.org/10.1109/TCYB.2021.3095305
26. J. Wang, C. Xu, W. Yang, L. Yu, A normalized gaussian wasserstein distance for tiny object detection, preprint, arXiv:2110.13389.
27. S. Mehta, M. Rastegari, Separable self-attention for mobile vision transformers, preprint, arXiv: 2206.02680.
28. G. Huang, Z. Liu, L. Van Der Maaten, K. Q. Weinberger, Densely connected convolutional networks, in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2017), 2261–2269. https://doi.org/10.1109/CVPR.2017.243
29. X. Pan, C. Ge, R. Lu, S. Song, G. Chen, Z. Huang, et al., On the integration of self-attention and convolution, in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2022), 815–825. https://doi.org/10.1109/CVPR52688.2022.00089
30. C. Li, A. Zhou, A. Yao, Omni-dimensional dynamic convolution, preprint, arXiv: 2209.07947.
31. S. Woo, J. Park, J. Lee, I. S. Kweon, CBAM: convolution block attention module, preprint, arXiv:1807.06521.

32. S. Mehta, M. Rastegari, MobileViT: Light-weight, general-purpose, and mobile-friendly vision transformer, preprint, arXiv: 2110.02178.
33. C. Wang, H. M. Liao, Y. Wu, P. Chen, J. Hsieh, I. Yeh, CSPNet: A new backbone that can enhance learning capability of CNN, in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, (2020), 1571–1580. https://doi.org/10.1109/CVPRW50498.2020.00203
34. J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2018), 7132–7141. https://doi.org/10.1109/CVPR.2018.00745
35. J. Fu, H. Zheng, T. Mei, Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition, in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2017), 4476–4484. https://doi.org/10.1109/CVPR.2017.476
36. Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image quality assessment: from error visibility to structural similarity, *IEEE Trans. Image Process.*, **13** (2004), 600–612. https://doi.org/10.1109/TIP.2003.819861
37. X. Li, Z. Yang, H. Wu, Face detection based on receptive field enhanced multi-task cascaded convolutional neural networks, *IEEE Access*, **8** (2020), 174922–174930. https://doi.org/10.1109/ACCESS.2020.3023782
38. R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-CAM: Visual explanations from deep networks via gradient-based localization, in *2017 IEEE International Conference on Computer Vision (ICCV)*, (2017), 618–626. https://doi.org/10.1109/ICCV.2017.74

AIMS Press