Computer Fraud and Security
ISSN (online): 1873-7056

An Improved YOLOv8 Fish Identification Based on PConv and Attention Mechanism

Aimin Lu[1], Guoyan Yu[1,2], Yiheng Xian[3], Liwen Wu[3], Zhao Li[3,4,*]

[1]School of Mechanical Engineering, Guangdong Ocean University, Zhanjiang 524000, Guangdong, China
[2]Guangdong Provincial Marine Equipment and Manufacturing Engineering Technology Research Center, Zhanjiang 524000, Guangdong, China
[3]College of Mathematics and Computer, Guangdong Ocean University, Zhanjiang 524000, Guangdong, China
[4]Southern Marine Science and Engineering Guangdong Laboratory (Zhanjiang), Zhanjiang 524000, Guangdong, China
**Corresponding Author.*

Abstract:
Accurate and effective identification of fish species is crucial for the processing and transportation of fish products. It enables efficient tracking and management of fish products, making it an essential aspect of the industry. Traditional methods of fish identification usually require expert knowledge within a particular domain. However, these methods can struggle to capture complex fish characteristics, particularly under varying lighting and angle conditions. In the article, an effective recognition algorithm named YOLO-FD is proposed. First, a novel feature extraction module has replaced the C2f module in YOLOv8, which effectively reduces the amount of model parameter calculations. The Efficient Multi-Scale Attention (EMA) applied in this study adeptly preserves spatial and channel information, fostering inter-regional interaction and enhancing feature extraction within the backbone network. The loss function in YOLOv8 was improved to address the sample imbalance problem. The YOLO-FD as an effective fish recognition algorithm addresses challenges faced by traditional methods simultaneously enhancing identification accuracy and offering lightweight improvement.

Keywords: fish identification, YOLOv8, PConv, attention mechanism, loss function

INTRODUCTION

Identification of fish species is essential for the processing of fish products. In the past, fish recognition was mainly based on the extraction of fish morphological features by classifiers based on support vector machines (SVM) [1]. These methods are costly, unable to extract complex features, and have poor robustness. In the last few years, the swift advancement of computer hardware has provided the fundamental computing power for the operation of deep learning. The convolution neural networks have made outstanding contributions to the field of computer vision [2-4]. Therefore, neural networks are an excellent tool for fish identification, providing accurate and reliable results. Excitingly, there are a variety of representative algorithms based on convolutional neural networks to choose from, including the powerful R-CNN [5-7], the efficient YOLO [8-11], and the speedy SSD [12]. The application of convolutional neural networks significantly enhances the ability to extract fish features and exhibits excellent robustness. The R-CNN series algorithm is not suitable for real-time detection at fish processing and production sites because of its slow operation speed due to the characteristics of a two-stage algorithm. The YOLO series of algorithms are frequently employed for their exceptional target detection capabilities, serving as a renowned one-stage method. Yihunie et al. [13] constructed a classification detection model using MobileNetv3 and VGG16 as the backbone network and an SSD detection head for fish recognition. Cai et al. [14] replaced the backbone network of YOLOv3 with MobileNetv1 to successfully identify fish in aquaculture plants, but the generalization of the algorithm has not been verified. Li et al. [15] suggested a method for recognizing fish faces that employs rotating target detection, leading to enhanced precision. However, the method necessitates high hardware requirements and may not be suitable for practical detection. While numerous methods are available for target detection, the majority concentrate on enhancing detection accuracy. This, in turn, results in greater network complexity and increased hardware requirements. At fish processing and production sites, it is difficult to meet the hardware requirements described above, which makes it difficult to migrate mature identification methods directly. Thus it can be seen it is crucial to explore a target detection technique for edge devices.

To facilitate target detection on edge devices, this paper presents an optimized YOLOv8-based algorithm for detecting fish targets. The algorithm is evaluated using a fish image dataset in a real-world setting, demonstrating its effectiveness. The main contributions of this paper are in the following way:

1. The partial convolution is introduced to improve the backbone network of YOLOv8 by drastically reducing the parameters and computational complexity.

2. EMA was added to help the backbone focus on the key area and thus improve detection accuracy.

3. The Slide weighting function has been subject to improve the loss function to address the sample imbalance problem.

The remainder of this paper is structured in the following way: Section 2 mainly examines the methods utilized in the paper. Section 3 mainly introduces the details of the fish target identification algorithm. Section 4 mainly introduces the experimental findings and evaluation. Section 5 concludes the paper with a summary

RELATED WORK

YOLO

YOLO (You Look Only Once) stands out as a paradigmatic one-stage object detection algorithm, approaching object detection as a regression problem. The method achieves concurrent localization and classification of objects by dividing the images into a mesh and making predictions within each mesh unit. The YOLO series distinguishes itself in real-time applications, courtesy of its singular forward propagation design, rendering it well-suited for scenarios requiring rapid object detection. YOLOv8 builds on the triumphs of its forerunners, introducing novel features and refinement to further enhance performance and portability.

Partial Convolution

Traditional convolution operates on all input tensor channels simultaneously via the kernel, which uniformly affects the feature maps of each channel. This concurrent convolution application may cause computational redundancy, especially when the information is redundant across multiple channels. The high computational complexity is a result of traditional convolution requiring consideration of all channels in the entire input during computation.

The fundamental idea behind partial convolution [16] is to strategically involve a subset of the input tensor's channels in convolution operations while leaving the remaining channels untouched. Additionally, the computational complexity of partial convolution is relatively lower since it only works on a portion of the channels. This approach effectively minimizes computational redundancy, especially in cases where there is a high channel correlation.

Attention Mechanism

The attention mechanism [17, 18] is a pivotal technology in the domain of deep learning that takes inspiration from the intricate workings of human vision and perception. It enables neural networks to prioritize essential elements during data processing, thereby bolstering the overall effectiveness and generalization capabilities of the algorithm. In contrast to uniformly processing all inputs, the attention mechanism allows the algorithm to selectively concentrate on the input sequence's particular sections. By implementing a weight allocation mechanism, the significance of each input position can be dynamically learned.

Efficient Multi-Scale Attention (EMA) [19] emerges as an efficient attention mechanism across space, ingeniously transforming certain channel dimensions into batch dimensions. This innovative approach preserves spatial information while concurrently mitigating computational overhead. The model can be made not only more efficient in weight computation but also able to maintain the same detection performance.

Loss Function

The issue of sample imbalance is a significant challenge in multi-class object detection tasks. In most cases, the quantity of hard samples is relatively less than the quantity of easy samples, contributing limited information to

the model training. By increasing the focus on hard samples during training, and elevating their contribution to the model training process, there can be a notable enhancement in the identification reliability of the model.

YOLOv8 employs a binary cross-entropy loss (BCELoss) function for classification loss. However, this function assigns equal weights to all classes, which is not favorable for handling hard samples. To work around this issue, the improved weight function is utilized to enhance the binary cross-entropy loss function. The improved weight function increases the weight assigned to hard samples, enhancing their contribution to the model, and alleviating the issue of sample imbalance.

Fish Detection

Traditional fish detection [1, 20] relies on experts manually extracting features and then using support vector machines or decision trees to classify fish based on these features. This method is costly, unable to extract complex features, and has poor robustness. In the last couple of years, the swift advancement of computer hardware has provided the fundamental computing power for the operation of deep learning. CNNs have also been applied to fish identification. While numerous methods based on CNNs are available for identification, the majority concentrate on enhancing detection accuracy. This, in turn, results in greater network complexity and increased hardware requirements. At fish processing and production sites, it is difficult to meet the hardware requirements described above, which makes it difficult to migrate mature identification methods directly. Deep learning-based fish recognition has improved detection accuracy, but it also requires high hardware requirements, which makes it difficult to be directly applied to real-world detection scenarios. In this paper, a lightweight fish detection and recognition algorithm is proposed, which greatly reduces the computational burden.

METHODS

The Overall Structure of YOLO-FD

In this study, a lightweight fish identification algorithm YOLO-FD has been designed. Firstly, a novel lightweight module, called CPE, for the extraction of features has been designed. The overall architecture of YOLOv8 is retained while replacing the C2f module in the backbone and neck networks with the CPE module. The overall architecture of YOLO-FD can be seen in Figure 1. The loss function is replaced with an improved version by incorporating a dynamic weighting parameter. The CBS module is the fundamental module for carrying out convolutions. The CPE module is the proposed new feature extraction module. The full name of SPPF is Spatial Pyramid Pooling-Fast. This module can effectively avoid the issue of picture warping caused by clipping and scaling. Concat: The Feature fusion module. Upsample: The upsampling module.

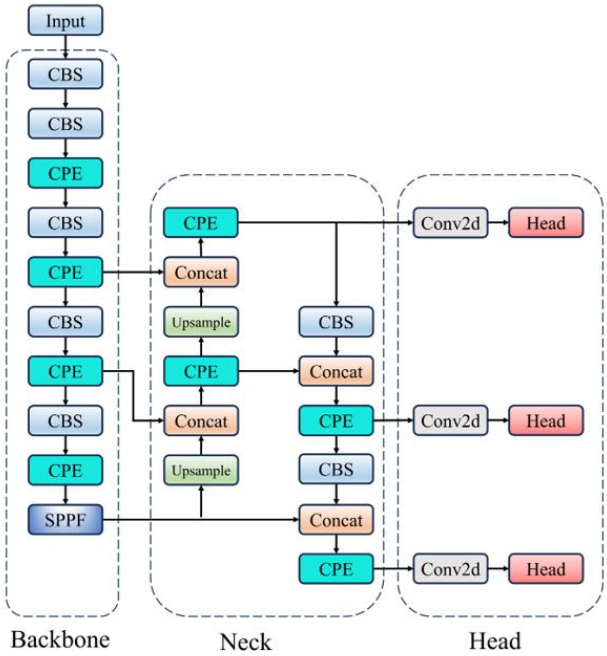


Figure 1. The structure of YOLO-FD

Improved Feature Extraction Module Design

Lightweight convolution module

The feature extraction module C2f in YOLOv8 includes two ordinary convolution operations with a kernel size of 3 ×3. While this structure allows for the extraction of richer features, it significantly increases computational overhead. The introduction of partial convolution (PConv) in this paper, as depicted in Figure 2, replaces the ordinary convolutions in the C2f module. For a feature map with input channels C, where C_p channels undergo convolution with a $k \times k$ kernel, and the remaining $C - C_p$ channels undergo convolution with a 1×1 kernel. This substitution leads to a substantial reduction in the computational cost of the algorithm.

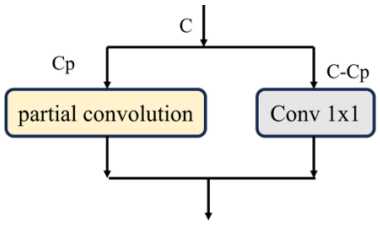


Figure 2. The structure of PConv

Taking the example of the first C2f module, let's calculate the FLOPs. The input is a feature map of size 128 × 160 × 160, where 128 is the number of channels, and 160x160 is the spatial dimension. Ordinary convolutions use 128 filters of size 3x3 to compute an output feature map of size 128 × 160 × 160.

FLOPs for ordinary convolution are calculated as follows:

$160 \times 160 \times 128^2 \times 3^2 = 3.7 \times 10^9$

The number of channels for PConv is taken as 32, that is, 32 channels are used for 3×3 convolution, and the remaining 96 channels are only used for 1×1 convolution. FLOPs for PConv is calculated as follows:

$160 \times 160 \times 128 \times (3^2 \times 32 + 128 - 32) = 1.3 \times 10^9$

The FLOPs of PConv are reduced by about 65% compared to that of ordinary convolution.

Efficient multi-scale attention

In this paper, the introduction of the EMA aims to decrease the computational overhead while retaining the function of every channel. By transforming some of the channel dimensions into bulk dimensions and avoiding a form of dimensionality reduction through generic convolution, the model can be made not only more efficient in weight computation but also able to maintain the same detection performance. To enhance the algorithm's feature extraction capabilities, the EMA module has been introduced within the Bottleneck module.

The revised Bottleneck module takes the place of the original C2f module, resulting in the development of our proposed enhanced feature extraction module, referred to as CPE, as depicted in Figure 3.

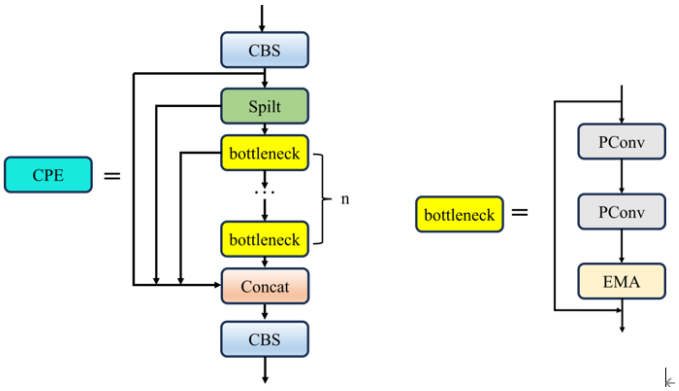


Figure 3. The structure of CPE

The reinforced loss function

YOLOv8 employs a BCELoss function for classification loss. The Eq. (1) shows the binary cross-entropy loss as follows:

$$L(p,t) = -[t \cdot logp + (1 - t) \cdot log(1 - p)] \quad (1)$$

Parameter p represents the predicted probability of a sample, and t denotes the sample label, with values 0, 1. When t = 1, it signifies that the sample is classified as positive, and when t = 0, it indicates that the sample is classified as negative. In this paper, a dynamic weighting parameter α is introduced on top of the binary cross-entropy loss. The parameter α, by assigning different weight coefficients to samples, enhances the contribution of hard samples to the model training. The design of the weight coefficients is inspired by the Slide weighting function and can be expressed by Eq. (2):

$$\alpha = \begin{cases} 1, p < \mu \\ e^{1-\mu}, \mu - 0.1 < p < \mu \\ e^{1-p}, p > \mu \end{cases} \quad (2)$$

The parameter μ represents the average Intersection over Union (IoU) value of all bounding boxes and serves as the threshold for determining positive and negative samples. Samples with IoU values greater than the threshold are considered positive, while those with values below the threshold are considered negative. By increasing the weight of negative samples near the threshold, the model pays more attention to this subset of samples, maximizing their utilization in training the model.

The final improved loss function is expressed as Eq. (3):

$$Loss(p,t) = \alpha L(p,t) \quad (3)$$

EXPERIMENT

Experiment Settings

Experimental environment

To guarantee the consistency and robustness of the experimental results, all experiments were carried out in the same environment. All the experiments were carried out on a server equipped with one Intel(R) Core(R) i9-10980XE CPU @ 1.70 GHz, two NVIDIA(R) GeForce RTX 4090 GPUs with 24 GB of video memory each, and 128 GB of RAM. The operating system utilized was Ubuntu 20.04, with CUDA version 12.2 and CUDNN version 8.9.2. Open-source machine learning libraries employed include PyTorch 2.0.1 and torchvision version 0.15.2, while the Python version used was 3.8.0. The hyperparameters for the specific training were confirmed as follows: The input image dimensions for the experiment were set to 640 × 640, the entire training process comprised 200 epochs, and a batch size of 32 was used during training.

Datasets

Figure 4. Instances in the dataset

The dataset of this study includes 99 common economic fish species found in the South China Sea. Image samples were collected at the market using a variety of cell phones to match the real-world scenario better. All images are stored in RGB color space and JPG format. The same kind of fish was photographed from multiple angles to enhance its generalizability. The dataset is a collection of 19,680 images, divided into 99 distinct classes. An example of some of the fish in the dataset is shown in Figure 4.

The LabelImg annotation tool was used to label the location of each fish in every image with a bounding box. The objective of this labeling technique is to label the fish class and its position in the image.

To experiment, the dataset has been split into three sections: 80% to train, 10% to validate, and 10% to test.

Evaluation metrics

In this study, parameter size, FLOPs, AP_{50}, AP_{75}, and AP are employed as metrics to assess the model's performance. Table 1 displays the available metrics.

Table 1. Evaluation metrics

Evaluation Metric	Meaning
AP	AP at *IoU* = .50 : .05 : .95
AP_{50}	AP at *IoU* = .50
AP_{75}	AP at *IoU* = .75
Parameter size	the cumulative sum of parameters across each layer in the model
FLOPs	floating point of operations

Baseline

The five network models that Ultralytics formally offers are YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x. To identify the detection of fish accurately and effectively, five kinds of network models of YOLOv8 are trained respectively, and the corresponding training results can be seen in Table 2.

Table 2. Comparison of YOLO8 models with different scale sizes

Model	*AP*/%	AP_{50}/%	AP_{75}/%	*Param/M*	*FLOPs/G*
YOLOv8n	82.1	89.8	87.6	3.2	8.7
YOLOv8s	81.9	90.6	88.9	11.2	28.6
YOLOv8m	82.5	**90.7**	**89.1**	25.9	78.9
YOLOv8l	**83.1**	90.5	88.8	43.7	165.2
YOLOv8x	82.0	90.6	88.9	68.2	257.8

According to Table 2, it is apparent that YOLOv8l achieves the highest AP accuracy of 83.1%, followed by YOLOv8m at 82.5%. YOLOv8m also attains the highest AP_{50} and AP_{75} values of 90.7% and 88.9% respectively. In comparison, YOLOv8n, YOLOv8s, and YOLOv8x have lower AP, AP_{50}, and AP_{75} accuracy scores than YOLOv8m. When comparing YOLOv8m with YOLOv8l, it has been discovered that the latter has approximately 1.7 times the number of parameters as the former, and about 2.1 times the number of FLOPs. Nevertheless, the improvement in AP accuracy achieved by YOLOv8l in comparison to YOLOv8m is only 0.6% while its AP_{50} and AP_{75} accuracy rates are lower than YOLOv8m.

By analyzing the network structures, variations in depth and width can be observed. YOLOv8n exhibits the smallest network, whereas the other four networks have expanded the network on this basis. Combined with the analysis in Table 2, it becomes evident that, as the depth and width of the network are increased, the model's complexity and performance are not directly proportional. Furthermore, solely increasing the depth and width of the network does not lead to enhanced accuracy of fish detection. Instead, it increases the number of model parameters and computations required. In summary, considering the accuracy of detection and model size, this paper has opted to utilize YOLOv8m as the baseline model for the study.

Ablation Experiment

To demonstrate the varying impacts of the partial convolution, the EMA attention mechanism, and the improving loss function on the YOLOv8m model through data, the experiments were executed on the fish dataset. YOLOv8m was utilized as the baseline model, and the partial convolution, the EMA attention mechanism, and the improving loss function were gradually integrated into the model to create different enhanced models, which were then compared in experiments. The visualized training process of YOLO-FD can be seen in Figure 5. In the training process of YOLO-FD, it is evident that the loss function converges quickly, ensuring a smooth training trajectory. In the final 10 epochs, the curve experiences a sudden change due to the closing of data augmentation but ultimately achieves convergence.

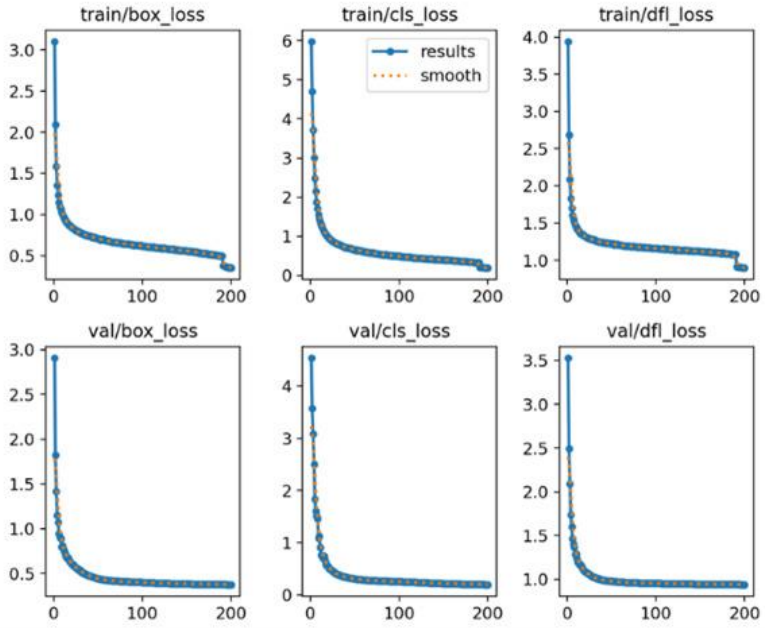


Figure 5. Training process

The experimental findings are illustrated in Table 3 and Figure 6. By analyzing Table 3, it is evident that the addition of the partial convolution, the EMA attention mechanism, and improving loss function in sequence to the base network has led to a substantial advancement in the identification accuracies of the models. The YOLOv8m-PConv has a large decrease in both the quantities of parameters and FLOPs compared to the baseline network.

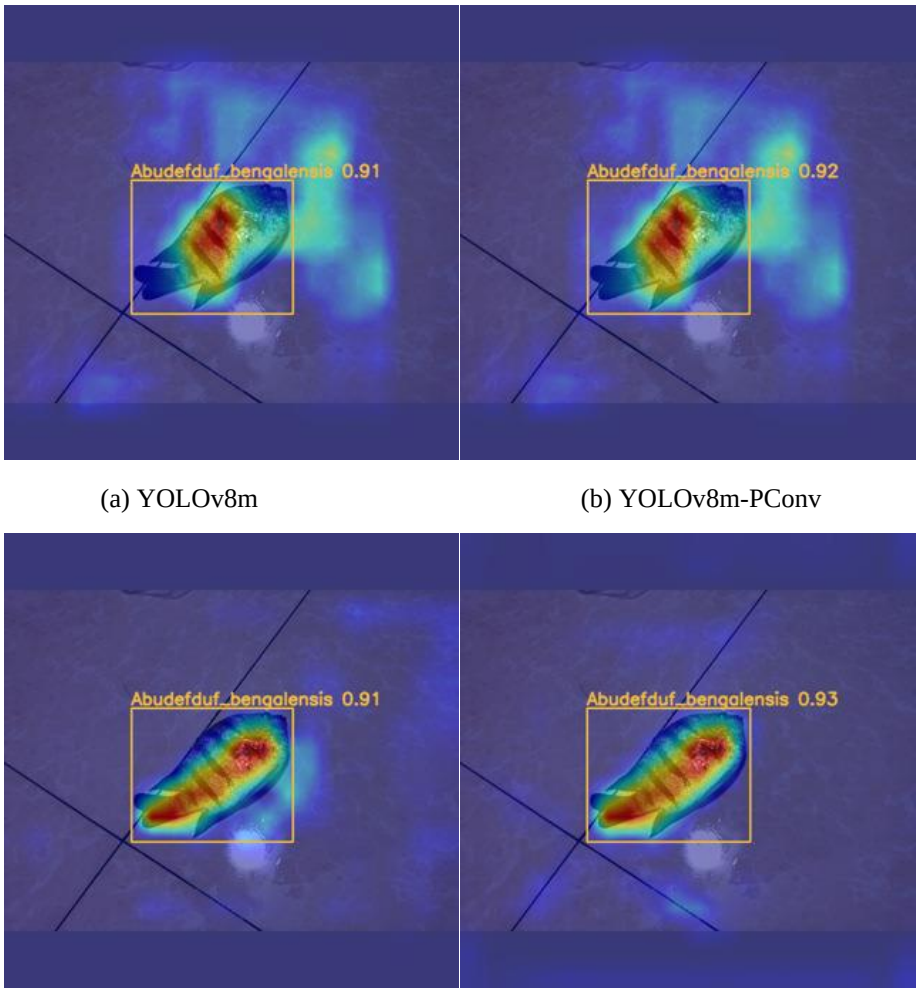
Table 3. Ablation results of different enhanced models

Model	*AP*/%	AP_{50}/%	AP_{75}/%	*Param/M*	*FLOPs/G*
YOLOv8m	82.5	90.7	89.1	25.9	78.9
YOLOv8m-PConv	82.6	90.9	89.3	16.7	50.1
YOLOv8m-PConv-EMA	83.8	91.9	90.4	16.7	**50.1**
YOLO-FD	**84.1**	**92.4**	**90.8**	**16.7**	52.8

The reduction in parameters and FLOPs amounts to 35.5% and 36.5%, respectively. Instead, it is a marginal increase in accuracy. AP, AP_{50}, and AP_{75} improved by 0.1%, 0.2%, and 0.2% respectively over the baseline network. The partial convolution diminishes the quantities of parameters in the model to a greater extent whilst maintaining accuracy and enhancing the efficiency of model execution. YOLOv8m-PConv-EMA incorporates the EMA attention mechanism, resulting in a small increase in FLOPs compared to YOLOv8m-PConv. However, the computational complexity remains clearly below that of the benchmark network. Meanwhile, the accuracy of YOLOv8m-PConv -EMA has improved significantly. This entails a 1.3%, 1.2%, and1.3% enhancement in comparison to the AP, AP_{50}, and AP_{75} of the baseline network, and 1.2%, 1.0%, and 1.1% advancement concerning the AP, AP_{50}, and AP_{75} of the YOLO8-PConv, correspondingly. The Slide weighting function is utilized to enhance the loss function of YOLOv8m-PConv-EMA, thereby resulting in the final YOLO-FD model proposed

within this paper. YOLO-FD shows no increase in model parameters or complexity compared to YOLOv8m-PConv-EMA, yet boasts improved accuracy. The AP, AP_{50}, and AP_{75} of YOLO-FD have improved by 0.3%, 0.5%, and 0.4%, respectively, compared to YOLOv8m-PConv-EMA; and by 1.6%, 1.7%, and 1.7%, respectively, when compared with the baseline network. The Slide weight function improves the imbalance of the samples without increasing the model complexity or quantities of model parameters, enhancing the algorithm's accuracy.

Figure 6 illustrates the heatmap of the detection results. YOLOv8m-PConv-EMA and YOLO-FD due to the added attention mechanism, the model is more focused on detecting the target and less focused on the environment. The detection confidence of YOLO-FD is also slightly higher than the other three models.



(a) YOLOv8m (b) YOLOv8m-PConv

(c) YOLOv8m-PConv-EMA (d) YOLO-FD

Figure 6. Detection result heatmap

To summarize, the YOLOv8m model incorporates the PConv, the EMA attention mechanism, and the Slide weighting function, which results in the proposed YOLO-FD model achieving enhanced identification accuracy while reducing the model's parameters and computation volume, diminishing the hardware requirements, and improving model portability.

Comparison with Other Identification Models

To further verify the YOLO-FD model's identification efficiency, this section compares it with the mainstream YOLOv5, YOLOv6, YOLOv7, YOLOv8, and Faster RCNN target detection model in the current mainstream. Table 4 displays the identification results of all models on the same conditional.

By analyzing Table 4, it is apparent that the YOLO-FD model exhibits noteworthy benefits regarding precision, quantity of model parameters, and FLOPs. Faster RCNN has a significant number of model parameters and

computations, but its detection accuracy is the lowest. Additionally, the AP, AP_{50}, and AP_{75} of Faster RCNN are lower than those of YOLO-FD by 3.9%, 4.0%, and 4.0%, respectively. The model parameters and computational complexity of YOLOv6m and YOLOv7 are both lower than those of RCNN. Furthermore, they achieve overall detection accuracies that are better than RCNN. YOLOv5m compared to YOLOv6m and YOLOv7 achieves higher detection accuracy with fewer model parameters. Compared to the YOLO-FD model, YOLOv5m exhibits a decrease of 1.7%, 2.0%, and 2.0% in AP, AP_{50}, and AP_{75}, respectively, in terms of detection accuracy. Correspondingly, YOLOv5m has 26.9%more parameters, but 7.2% less computation, regarding model parameters and computation.

Table 4. Comparison of different detection techniques

Model	*AP*/%	AP_{50}/%	AP_{75}/%	*Param/M*	*FLOPs/G*
Faster RCNN	80.2	88.4	86.8	41.3	251.4
YOLOv5m	82.4	90.4	88.8	21.2	**49.0**
YOLOv6m	81.1	89.2	87.7	34.9	85.5
YOLOv7	81.3	89.3	87.9	36.9	104.7
YOLOv8m	82.5	90.7	89.1	25.9	78.9
YOLO-FD	**84.1**	**92.4**	**90.8**	**16.7**	52.8

In summary, when comparing the Faster RCNN, YOLOv5, YOLOv6, YOLOv7, and YOLOv8 models, the YOLO-FD model described in this paper enhances computational efficiency and maintains high detection accuracy. This suggests that the model focuses on effective target features, reduces redundant or ineffective features, and thus enhances network performance.

CONCLUSION

This study presents an efficient fish identification model on the basis of YOLOv8.By improving the backbone structure, the addition of an attention mechanism, and refining the loss function, the model achieved high prediction accuracy in experiments on the fish dataset with fewer parameters and computations than the other target detection algorithms, to demonstrate that it performs better than most.

Our dataset consists mainly of clear and complete fish images, but unfortunately, there is a lack of images of occluded or mutilated fish. This lack of data may lead to a bias in the actual identification process of our algorithm.

Therefore, to further refine the study, the plan is to add critical data images in future work. By introducing more occluded or mutilated fish images, more accurate simulation of recognition scenarios in the real world can be achieved, providing more challenging situations for the algorithm. Simultaneously, focus will be placed on improving the algorithm to enhance the recognition ability for these incomplete cases. Through these efforts, it is expected to enhance the practicality and robustness of the algorithm so that it can perform well in various complex situations.

ACKNOWLEDGMENTS

This work is supported by the Program for Scientific Research Start-up Funds of Guangdong Ocean University (No. R20079, 060302102302), Key Laboratory of Modern Marine Fishery Equipment in Zhanjiang (2021A05023), Guangdong Postgraduate Education Innovation Programme (2023JGXM_75).

REFERENCES

[1] O. Ulucan, D. Karakaya, and M. Turkan, A Large-Scale Dataset for Fish Segmentation and Classification, 2020 Innovations in Intelligent Systems and Applications Conference (ASYU), Istanbul, Turkey, 2020, pp. 1-5.

[2] T. T. Khoei, H, O. Slimane and N. Kaabouch, Deep learning: systematic review, models, challenges, and research directions, Neural Comput & Applic 35, 23103–23124 (2023). https://doi.org/10.1007/s00521-023-08957-4.

[3] L. Alzubaidi, J, Bai, A. Al-Sabaawi, J. Santamar´ıa. et al, A survey on deep learning tools dealing with data scarcity: definitions, challenges, solutions, tips, and applications, J BigData 10, 46 (2023). https://doi.org/10.1186/s40537-023-00727-2.

[4] K. He, N. Pu, M. Lao and M. S. Lew, Few-shot and meta-learning methods for image understanding: a survey, Int J Multimed Info Retr 12, 14 (2023). https://doi.org/10.1007/s13735-023-00279-4.

[5] R. Girshick, J. Donahue, T. Darrell and J. Malik, Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 2014, pp. 580-587, doi:10.1109/CVPR.2014.81.

[6] R. Girshick, Fast R-CNN, 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 2015, pp. 1440-1448, doi: 10.1109/ICCV.2015.169.

[7] S. Ren, K. He, R. Girshick and J. Sun, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, 1 June 2017, doi: 10.1109/TPAMI.2016.2577031.

[8] J. Redmon, S. Divvala. R. Girshick, and A. Farhadi, You Only Look Once: Unified, Real-Time Object Detection,2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.

[9] J. Redmon and A. Farhadi, YOLO9000: Better, Faster, Stronger, 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017, pp.6517-6525, doi: 10.1109/CVPR.2017.690.

[10] A. Nazir and M. A. Wani, You Only Look Once - Object Detection Models: A Review, 2023 10th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2023, pp. 1088-1095.

[11] C. -Y. Wang, A. Bochkovskiy and H. -Y.M. Liao, YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors, 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 2023, pp.7464-7475, doi: 10.1109/CVPR52729.2023.00721.

[12] W. Liu, D. Anguelov, D. Erhan and C. Szegedy, SSD: Single Shot MultiBox Detector, In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds) Computer Vision – ECCV 2016. ECCV 2016. Lecture Notes in Computer Science, vol 9905. Springer, Cham. https://doi.org/10.1007/978-3-319-46448-02

[13] A. S. Yihunie, N. M.M, S. Chiranjibi and P. Jack, Class-Aware Fish Species Recognition Using Deep Learning for an Imbalanced Dataset, SENSORS, 22, 21(2022), doi:10.3390/s22218268

[14] K. W. Cai, X. Y. Miao, W. Wang, H. S. Pang, Y. Liu and J. Y. Song, A modified YOLOv3 model for fish detection based on MobileNetv1 as backbone, 91(2020), doi:10.1016/j.aquaeng.2020.102117

[15] D. Y. Li, H. C. Su, K. L. Jiang, D. Liu, X. L. Duan, Fish Face Identification Based on Rotated Object Detection: Dataset and Exploration, FISHES, 7, 4(2022), doi:10.3390/fishes7050219

[16] J. Chen, S. Kao and H. He, Run, Don't Walk: Chasing Higher FLOPS for Faster Neural Networks, 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 2023, pp. 12021-12031.

[17] G. Brauwers and F. Frasincar, A General Survey on Attention Mechanisms in Deep Learning, in IEEE Transactions on Knowledge and Data Engineering, vol. 35, no. 4, pp. 3279-3298, 1 April 2023, doi: 10.1109/TKDE.2021.3126456.

[18] M. H. Guo, T. X. Xu, J. J. Liu, et al., Attention mechanisms in computer vision: A survey, Comp. Visual Media 8, 331–368 (2022). https://doi.org/10.1007/s41095-022-0271-y

[19] D. L. Ouyang, S. He, G. Z. Zhang et al., Efficient Multi-Scale Attention Module with Cross-Spatial Learning, CASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 2023, pp. 1-5, doi:10.1109/ICASSP49357.2023.10096516.

[20] Z. M. Qian, S. H. Wang, X. E. Chen et al., An effective and robust method for tracking multiple fish in video image based on fish head detection, BMC Bioinformatics 17, 251 (2016). https://doi.org/10.1186/s12859-016-1138-y