frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 09 July 2024
DOI 10.3389/fmars.2024.1424619

OPEN ACCESS

EDITED BY
Chao Zhou,
Beijing Research Center for Information Technology in Agriculture, China

REVIEWED BY
Ruobin Gao,
Nanyang Technological University, Singapore
Rajamanickam Narayanamoorthi,
SRM Institute of Science and Technology, India

*CORRESPONDENCE
Zhuowei Wang
zwwang@gdut.edu.cn

RECEIVED 28 April 2024
ACCEPTED 21 June 2024
PUBLISHED 09 July 2024

CITATION
Ruan Z, Wang Z and He Y (2024) DeformableFishNet: a high-precision lightweight target detector for underwater fish identification. *Front. Mar. Sci.* 11:1424619. doi: 10.3389/fmars.2024.1424619

DeformableFishNet: a high-precision lightweight target detector for underwater fish identification

Zhukang Ruan, Zhuowei Wang* and Yiqing He

School of Computer Science and Technology, Guangdong University of Technology, Guangzhou, China

The application of computer vision in fish identification facilitates researchers and managers to better comprehend and safeguard the aquatic ecological environment. Numerous researchers have harnessed deep learning methodologies for studying fish species identification. Nonetheless, this endeavor still encounters challenges such as high computational costs, a substantial number of parameters, and limited practicality. To address these issues, we propose a lightweight network architecture incorporating deformable convolutions, termed DeformableFishNet. Within DeformableFishNet, an efficient global coordinate attention module (EGCA) is introduced alongside a deformable convolution network (EDCN/EC2f), which is grounded in EGCA, to tackle the deformation of fish bodies induced by swimming motions. Additionally, an EC2f-based feature pyramid network (EDBFPN) and an efficient multi-scale decoupling head (EMSD Head) are proposed to extract multi-scale fish features within a lightweight framework. DeformableFishNet was deployed on our freshwater fish dataset, with experimental outcomes illustrating its efficacy, achieving a mean average precision (mAP) of 96.3%. The model comprises 1.7 million parameters and entails 4.7 billion floating-point operations (FLOPs). Furthermore, we validated DeformableFishNet on three public underwater datasets, yielding respective mAPs of 98%, 99.4%, and 83.6%. The experiments show that DeformableFishNet is suitable for underwater identification of various scenes.

KEYWORDS

fish identification, underwater images, deformable convolution, attention mechanism, deep learning, underwater target detection

1 Introduction

As technology advancements in artificial intelligence, the Internet of Things, and big data continue to flourish, edge computing has emerged as a pivotal paradigm shift within the realm of computational sciences (Deng et al., 2020; Chang et al., 2021). Leveraging the capabilities of edge devices, computer vision-based detection applications have infiltrated a myriad of domains (Li et al., 2022b; Jiang et al., 2023). Among these, the application of

computer vision for fish detection assumes paramount importance in advancing intelligent aquaculture systems. By meticulously analyzing the visual characteristics and anatomical structures of fish, this technology facilitates automatic species identification and categorization. This not only enables real-time surveillance of fish populations' spatial distribution and density in aquatic ecosystems but also furnishes invaluable ecological information. Nevertheless, implementing fish detection and classification in submerged aquatic habitats presents significant challenges, primarily due to the distinct complexities intrinsic to underwater environments.

In recent years, scholarly endeavors have largely focused on leveraging deep learning-driven detection and classification approaches to tackle the intricate challenges of fish detection and classification. Convolutional neural networks (CNNs) have gained remarkable prominence due to their inherent capability to autonomously discern and adaptively learn relevant features, thereby eliminating the need for manual feature engineering and exhaustive multi-stage analyses—for example, Labao and Naval (2019) innovatively combined region-based CNNs with long short-term memory to create a pioneering fish detection system. Similarly, Cai et al. (2020) employed the MobileNetv1 architecture in conjunction with YOLOv3 to develop an effective fish detection model. Furthermore, Prasetyo et al. (2022) designed a multi-stage residual network based on VGGNet to excel in fish classification tasks. It is worth noting that Xu et al. (2021) exploited transfer learning strategies and the advanced SE-ResNet152 model to adeptly address the intricate issue of identifying small-scale, imbalanced fish species with commendable accuracy. Collectively, these studies illustrate the vibrant progress in harnessing AI-powered tools to enhance the monitoring and management of aquatic ecosystems.

Underwater fish detection and recognition also confront the significant hurdle of limited labeled data availability. In addressing this constraint, Allken et al. (2018) innovatively adopted a deep visual image synthesis technique to augment the training dataset, achieving an impressive 94% classification accuracy for cod, Atlantic herring, and Atlantic mackerel. Meanwhile, Banan et al. (2020) leveraged the pretrained VGG16 model and further fine-tuned the expansive ImageNet corpus to boost the recognition capabilities for multiple fish species. This approach yielded significantly improved average classification rates, particularly in distinguishing among four different Asian carp species. Despite such progress, the dearth of freshwater fish datasets that accurately reflect natural environmental conditions persists as a pressing concern in the field.

Within the realm of deep learning-powered object detection, two primary architectural paradigms dominate the landscape: two-stage and one-stage frameworks. The two-stage approach commences with a meticulous region proposal phase, wherein areas suspected to harbor potential objects are pinpointed. Subsequently, these nominated regions undergo scrutiny by a CNN to precisely identify and localize objects within those confines. Esteemed exemplars embodying this methodological route encompass R-CNN, R-FCN, Fast R-CNN, Faster R-CNN, and Mask R-CNN. On the other hand, one-stage algorithms are such as the YOLO series (Redmon et al., 2016; Redmon and Farhadi, 2017, 2018; Mao et al., 2019; Bochkovskiy et al., 2020; Ge et al., 2021), SSD, and RetinaNet. These models ingeniously intertwine the region proposal procedure with the actual object detection task, streamlining the process for both enhanced speed and maintained accuracy. They are lauded for their uncanny ability to swiftly and accurately detect objects without the need for an explicit segmentation step, thereby illustrating the remarkable duality of speed and precision in modern object detection technology.

Recently, transformer-based architectures have risen to prominence, showcasing profound benefits across a wide array of visual tasks. These innovative models uniquely excel at discerning and encapsulating long-range interdependencies between objects, thereby empowering transformer-driven detectors to either match or surpass the performance benchmarks set by their more traditional counterparts. In the domain of object detection, a suite of groundbreaking models has emerged, each capitalizing on transformer-based encoder–decoder designs. Chief among these are Vision Transformer (ViT) (Dosovitskiy et al., 2020), Swin Transformer (Liu et al., 2021), and DETR (end-to-end object detection with transformers) (Carion et al., 2020) as well as the likes of RT-DETR (real-time detection transformer) and DINO (DETR with Improved deNoising anchOr boxes) (Zhang et al., 2022a). Despite these strides, compact yet efficient CNN-based object detection models still maintain their stronghold in striking the critical balance between speed and precision, as exemplified by the likes of YOLOX (Ge et al., 2021) and the subsequent generations of the YOLO family extending from YOLOv6 to YOLOv9 (Li et al., 2022a; Wang et al., 2023). This dynamic underscores the relentless pursuit of excellence in the rapidly evolving space of visual object detection technology.

Our investigation has illuminated several pivotal challenges inherent to contemporary underwater fish detection efforts. While current research extensively employs sophisticated detection models to boost fish detection performance, it often fails to adequately address specific challenges unique to this domain. Fish, being naturally non-rigid organisms, pose a particular challenge; their pose deformations during swimming can drastically undermine detection precision. Furthermore, the similarity in features across different fish species, compounded by notable intra-species variations, introduces an additional layer of intricacy to the task. In the realm of smart fisheries, there is a pressing requirement to optimize fish detection models for edge device implementation without sacrificing operational effectiveness. Lastly, the bulk of existing research centers on marine fish detection, with scant attention given to the distinct challenges and requirements of detecting fish in freshwater environments.

In this paper, in order to solve the challenge of underwater fish detection, we combine the proposed modules to propose a new network structure with lightweight and deformable convolution (DeformableFishNet). In addition, we capture and create an underwater fish dataset by using underwater cameras in natural underwater habitats to verify the performance of DeformableFishNet. The contribution of this work can be summarized as follows:

1. We have designed an efficient global coordinate attention mechanism (EGCA). EGCA enhances the key features of fish targets, leading to improved detection accuracy of fish by the detector. To address the issue of body distortion that fish experience while swimming in water, resulting in various body shapes within the same fish, we employ EGCA attention and deformable convolution to design a deformable convolution network (EDCN/EC2f).
2. For the purpose of lightweighting the model network structure and extract fish features better, we have designed a feature pyramid network structure (EDBFPN) by using EDCN/EC2f. Then, we redesigned an efficient multi-scale decoupled head (EMSD head) to reduce the model's parameter count and FLOPs. The EMSD head can obtain multi-scale feature maps through cheap convolution operations.
3. We combined the proposed modules to propose the DeformableFishNet. We applied DeformableFishNet to our freshwater fish dataset and public underwater datasets. The experiments show that our model not only performs admirably in detecting fish beneath the surface but also exhibits robust capabilities in identifying objects across a diverse array of underwater scenarios, showcasing its broad applicability and robustness in underwater target detection.

2 Related work

2.1 Fish identification and classification

In recent years, extensive research has focused on employing deep learning methodologies for fish detection and classification. DeepFish (Qin et al., 2016), for instance, initially extracts background features through sparse and low-rank matrix factorization, subsequently employing deep learning architectures to discern the key characteristics of frontal fish images. Another study by Zhou et al. (2022) integrates a self-attention mechanism within a tower-like structure, preceding the main CNN with a generative adversarial network (GAN) to enrich data variability. Researchers in the study of Knausgård et al. (2022) utilize a squeeze-and-excitation ResNet (SE-ResNet) augmented with a compressed and encouraged (CE) loss function for individual fish classification per image, and they also implement transfer learning strategies to mitigate the constraints imposed by limited training samples for various fish categories. The work presented by Ben Tamou et al. (2021) comprises a dual faster R-CNN configuration where the models share either a common region proposal network or a unified classifier. AdvFish (Zhang et al., 2022b) introduces a min–max bilevel adversarial optimization framework, enhancing model robustness by training on adversarially perturbed images using an adaptive perturbation methodology. Lastly, Mathur et al. (2020) merges features derived from layers 154 and 157 of ResNet-50 to elevate the model's overall performance.

Most studies have used better neural networks to enhance the performance of the models. However, fish is a non-rigid object. Fish can change their shape and posture due to swimming or bending, which may affect their features and appearance models. In this paper, we propose an efficient global coordinate attention module (EGCA) and EGCA-based deformable convolutional network (EDCN/EC2f) to address this problem.

2.2 Neural network module

The primary objective of convolutional layers is to extract features from input data. Conventionally, these layers are designed with fixed sizes and shapes. A key limitation of such traditional convolutional layers lies in their reduced adaptability to unforeseen variations, leading to a weaker generalization capacity. To address this issue, prevailing strategies often entail utilizing extensive datasets, incorporating more intricate deformable examples, employing diverse data augmentation methodologies, and manually devising customized features and algorithms. Nonetheless, despite these efforts, these conventional approaches continue to face constraints in achieving optimal adaptability and generalization.

Deformable convolution (Dai et al., 2017; Zhu et al., 2019; Wang et al., 2022) introduced offset amounts in the receptive field, which can be learned and adapted to fit the actual shape of objects. This allows the convolutional regions to always cover the surrounding area of the object, regardless of how it deforms. Deformable convolution learns appropriate convolution kernel parameters for each task, dynamically adjusting the shape and weights of the convolutional filters to capture features in the input data more effectively.

Deformable convolutional networks can dynamically adjust the shape and weights of convolutional kernels, but the adjustment of kernel offsets is generated only once through a convolution. Therefore, we propose an EGCA-based deformable convolutional network (EDCN/EC2f).

Traditional feature pyramid networks (FPN) suffer from information loss and redundancy when extracting features at different scales. Bi-directional feature pyramid network (BiFPN) (Tan et al., 2020) addresses this issue by fusing features from different resolutions through lateral and vertical connections, allowing for better integration and utilization of features across different scales.

In comparison to traditional self-attention downward FPN, BiFPN has several advantages. Firstly, it removes nodes with only one input edge. Secondly, it adds an additional edge between original input and output nodes to facilitate merging more features without increasing cost. Finally, it treats each double-directional path as a single feature networking layer and repeats the same layer multiple times to achieve a higher-level feature fusion. BiFPN employs three weighted fusion methods, namely, unbounded fusion, Softmax-based fusion, and fast normalized fusion.

BiFPN enables better feature fusion with fewer parameters and computational resources. However, BiFPN was not specifically designed for underwater fish detection and classification tasks. Therefore, we propose an EC2f-based feature pyramid network (EDBFPN), a dedicated architecture for underwater fish detection and classification tasks.

2.3 Object detection models

YOLOv8 is one of the latest versions of YOLO developed by Ultralytics, which builds upon the success of previous versions while introducing new functionalities and improvements to enhance performance and flexibility. The backbone network and neck utilize a richer gradient flow structure with C2f and adjust different channel numbers for various scale models. The head part separates the classification and detection heads, switching to the current mainstream decoupling head structure, and replaces anchor-based with anchor-free. YOLOv8 uses Task-Aligned Assigner positive and negative sample matching method and introduces distribution focal loss (DFL).

YOLOv8, compared with two-stage model and Vision Transformers model, has less computation and parameters. However, its head still has a large amount of calculation and parameters, and it is not a model designed specifically for underwater fish detection and classification tasks. Therefore, by combining multiple plug-and-play modules proposed by us, we designed a new network structure with lightweight and deformable convolution (DeformableFishNet) specifically for underwater fish detection and classification tasks.

In summary, based on previous research and the characteristics of underwater fish detection and classification tasks, we propose several plug-and-play modules, namely, EGCA, EDCN/EC2f, EDBFPN, and EMSD head. Finally, we combine the proposed modules to form a dedicated network for underwater fish detection and classification tasks called DeformableFishNet.

3 Methodology

The proposed DeformableFishNet network architecture is shown in Figure 1. We aimed at overcoming the unique challenges posed by underwater fish detection. To tackle the inherent issue of fish body distortion, which often compromises accurate feature extraction, we integrated a novel deformable convolution network (EDCN/EC2f) module into the backbone network. This module serves to efficiently and effectively extract crucial fish features such as morphological structures and textural details, thereby ensuring robust representation learning even in the face of complex body movements.

Furthermore, recognizing the importance of multi-scale feature integration in enhancing detection performance, we incorporated an EC2f-based feature pyramid network (EDBRPN) module within the neck part of the network. This EDBRPN module facilitates superior fusion of fish features extracted across various scales, thereby promoting comprehensive and discriminative understanding of the fish within the scene.

Lastly, in the head portion of the network, we adopted an efficient multi-scale decoupled (EMSD) head module. This innovative component empowers the model to procure multi-scale fish features with minimized computational overhead, thus dramatically reducing the parameter count and overall computational complexity of the detection model. As a result, this economization not only enhances the model's speed and accuracy but also significantly eases its deployment onto resource-constrained edge devices.

3.1 Efficient global coordinate attention module

Using attention mechanism can make the detection model automatically find and pay attention to the most relevant areas or features in the input image. It is helpful to strengthen the recognition ability of fish characteristics and focus on key areas, especially in complex and fuzzy underwater environment. When dealing with non-rigid object deformation, illumination change, occlusion, and other situations, attention mechanism helps the detection model to meet the challenges of these fish detection tasks flexibly, helps the model capture subtle local details, improves the recognition of boundaries and small targets, and then improves the overall performance of the model. Inspired by CA (Hou et al., 2021), CBAM (Woo et al., 2018), and SimAM (Yang et al., 2021) attention modules, we designed an efficient global coordinate attention module (EGCA). The structure of EGCA is shown in Figure 2.

Given an input tensor with feature dimensions of $C \times H \times W$, we first use parameter-free SimAM attention to obtain a tensor F1 that reduces redundant information in the input tensor and enhances our ability to perceive and utilize key features related to the input. We then multiply F1 with the input tensor, followed by global average pooling and convolution operations to produce a tensor F2 with global attention information.

We separately apply global average pooling in the width and height directions to the input tensor, resulting in two tensors, F3 and F4, with shapes $C \times H \times 1$ and $C \times 1 \times W$, respectively. We then concatenate these two tensors along the channel dimension, resulting in a tensor F5 with shape $C \times 1 \times (H + W)$. After applying a convolution layer, we obtain two tensors F6 and F7 with shapes $C \times H \times 1$ and $C \times 1 \times W$, respectively, through channel splitting and transpose operations.

To obtain more refined and meaningful information, we perform additional convolution operations on tensor F5. Then, we use Sigmod operation to obtain tensor F8 with weights on both the width and height dimensions. We use channel splitting after transposing the tensor to obtain separate weight matrices F6 weight and F7 weight for the width and height directions, respectively. Tensor F8 takes the mean value and multiplied with tensor F2 to obtain tensor F9.

Finally, we perform Sigmod operations on the products of F6 and F6 weight, F7 and F7 weight, and F9 and their respective weights and then multiply the results with the input tensor to obtain the output tensor of the EGCA attention module.

The EGCA attention mechanism weights the spatial coordinates of the feature map and focuses on the information in the global spatial coordinates, adaptively adjusting the importance of each position, thereby improving the model's ability to perceive and understand space information. This allows the model to better

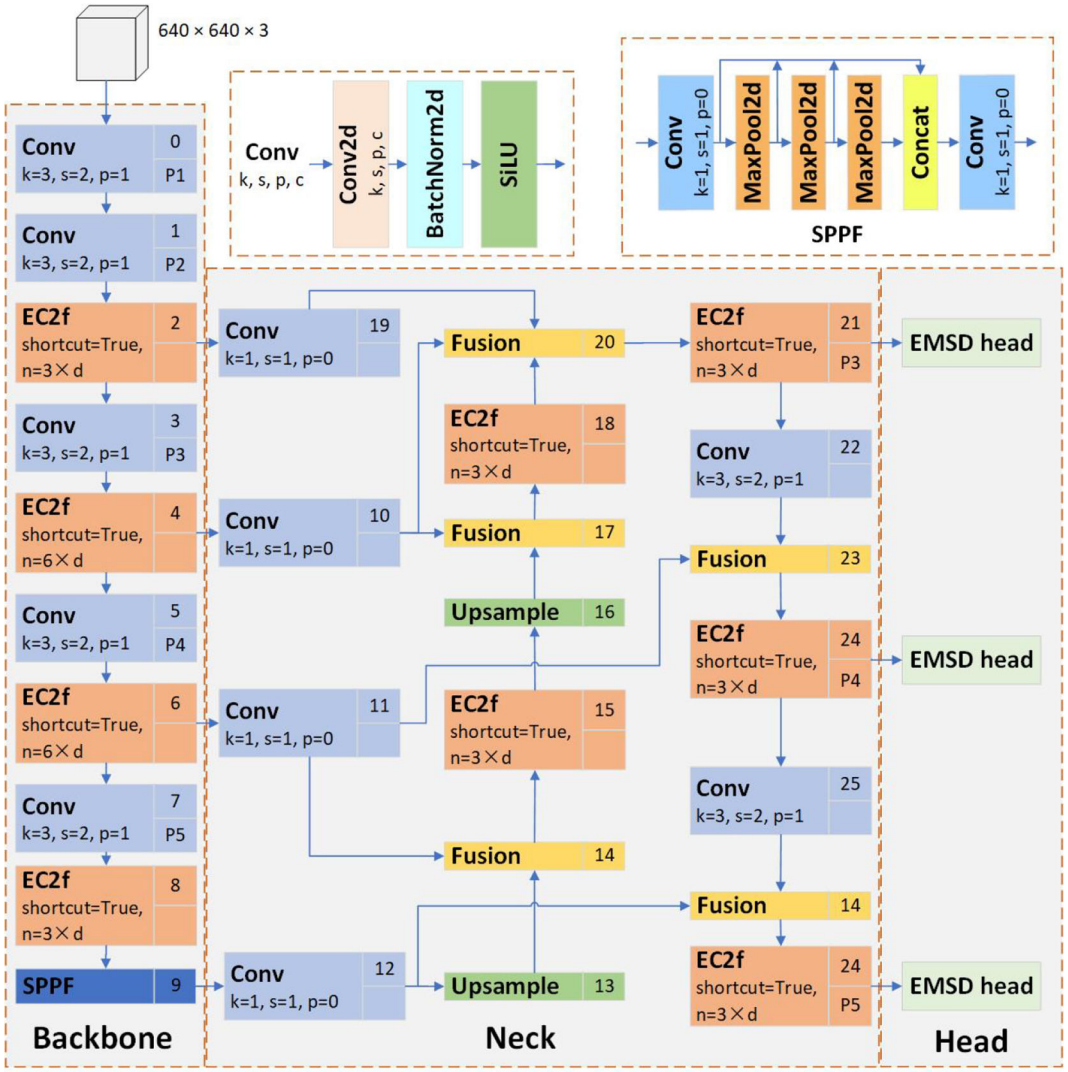

FIGURE 1
Overview of DeformableFishNet. The detailed structure of some modules will be shown below.

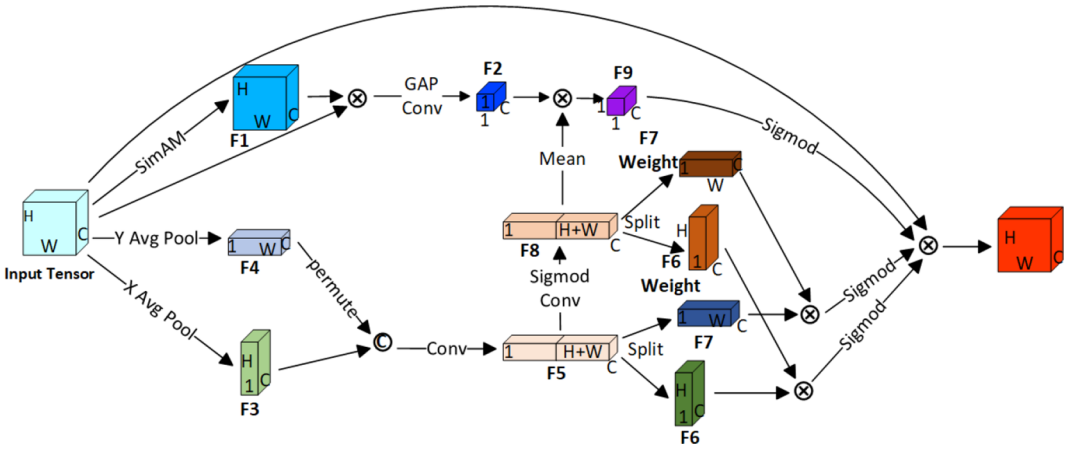

FIGURE 2
Structure of EGCA.

capture the key positions in images, videos, or other spatial data, enhancing the expressiveness of the spatial features.

3.2 EGCA-based deformable convolution (EDCN/EC2f)

As fish swim underwater, their supple bodies undergo a gamut of deformations and contortions. Notably, even a single fish specimen can manifest a myriad of postures, morphing seamlessly from one silhouette to another. Conventional CNNs, with their rigidly fixed convolutional kernel weights, enforce uniform receptive fields across all regions of an image, which can prove inadequate when confronted with the nuanced demands of tasks like fish detection where adaptability is paramount.

Deformable convolution (Dai et al., 2017; Zhu et al., 2019; Wang et al., 2022) is a type of convolutional operation. This technique endows the model with learnable parameters that enable the generation of adaptable convolutional kernels tailored specifically to the idiosyncrasies of the input data. By learning the optimal kernel parameters for each given task instance, deformable convolution deftly adjusts the geometric structure and weighting of the convolutional filters. This dynamism allows for a more incisive capture of the intrinsic features embedded within the input samples, thereby enhancing the model's performance across a broad spectrum of inputs.

However, earlier implementations of deformable convolution derived the offset amounts needed to reshape the kernels by simply applying additional convolutional layers to the same input feature maps. Recognizing the necessity to solve the difficulty of feature extraction caused by fish swimming, we innovatively integrated the EGCA mechanism with deformable convolution, thus giving birth to the EGCA-based deformable convolutional network (EDCN), which is visually illustrated in Figure 3. Our EDCN employs a convolutional operation coupled with an EGCA attention module to compute the finely calibrated offset quantities required for the deformable convolutional kernels.

To escalate the effectiveness of fish feature extraction even further, we built upon the C2f convolutional module of the YOLOv8 model, devising the EC2f module, as showcased in Figure 4. In this evolution, the EDCN substitutes the secondary convolutional layer within the original Bottleneck structure. Culminating our architectural enhancements, we judiciously implemented the EC2f module into the backbone network of YOLOv8, thereby fortifying its capability to extract and interpret the complex and varied features of fish in underwater imagery.

3.3 EC2f-based feature pyramid network

The BiFPN (Tan et al., 2020) is a hierarchical feature network structure that enables bidirectional feature fusion between different levels of feature pyramids. Through top-down and bottom-up feature propagation and fusion, BiFPN can effectively combine low-level detail features and high-level semantic features, providing multi-scale and multi-dimensional feature expression capabilities.

In order to fuse the fish features extracted at various scales, the detection model can promote a comprehensive and differentiated understanding of the fish in the scene. To achieve better performance and accuracy while reducing the model complexity, we referenced BiFPN and designed an EC2f-based feature pyramid

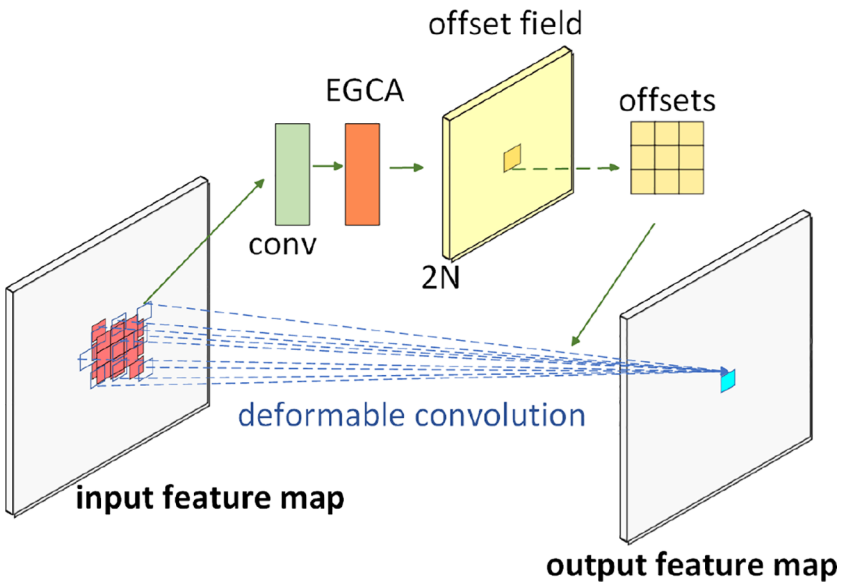

FIGURE 3
Structure of EDCN.

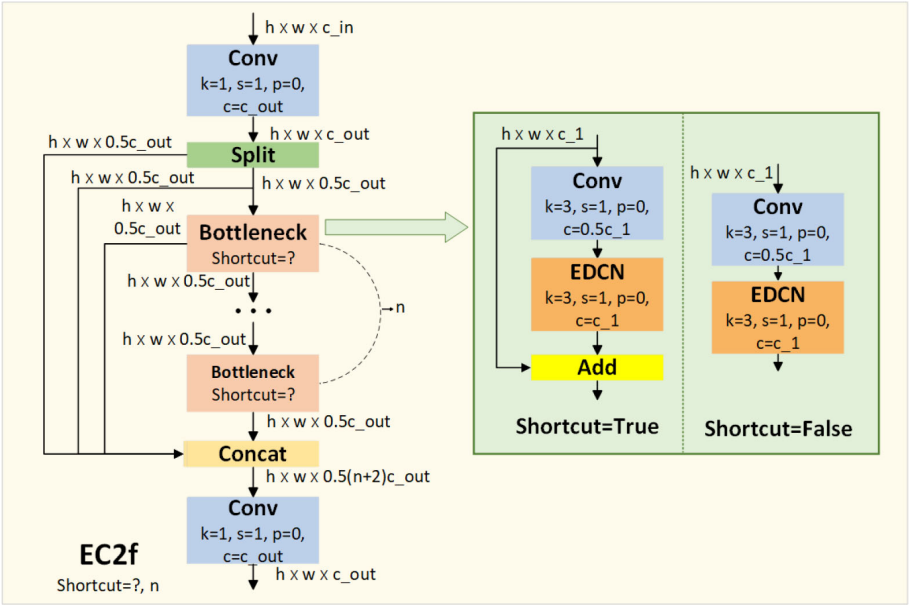

FIGURE 4
Structure of EC2f. The shortcut represents whether to make a residual connection or not. *n* represents the number of stacked Bottleneck modules.

network (EDBFPN), as shown in Figure 5. In EDBFPN, the Fusion module performs a weighted sum of the input features based on learned weights to implement feature fusion. Fusion uses Fast normalized fusion in BiFPN.

3.4 Efficient multi-scale decoupling head

Han et al. (2020) and Tang et al. (2022) found that traditional neural networks often produce excessive and redundant feature numbers, which lead to additional parameters and floating-point operations. Pointwise convolution acts as an upward and downward transformation of feature channels. Howard et al. (2017); Sandler et al. (2018), and Howard et al. (2019) discovered and utilized pointwise convolution to fuse features across different channels. Yu and Koltun (2015) and Yu et al. (2017) pointed out that using dilated convolution can enlarge the receptive field of the feature map without changing its shape and ensure that the input and output feature maps have the same shape. Wang et al. (2018) suggested that using multiple expansion coefficients with different dilation rates leads to better performance when performing dilated convolution. To maintain high accuracy and lightweight models, we redesigned an efficient multi-scale decoupling head (EMSD head) with a new structure, as shown in Figure 6. In the EMSD head, we design an efficient multi-scale convolution module (EMSConv) that can obtain multi-scale features at a low cost. To obtain multi-scale features without increasing the number of parameters and floating-point operations, we first divide the input feature channel into four parts. The first part uses a regular convolution kernel with a size of 1 and no dilation. In the second, third, and fourth parts, we use convolution kernels with the size of 3 and the dilation rates of 1, 2, and 3, respectively, for dilated convolution. We concatenate the four feature maps obtained after each convolution operation and then use pointwise convolution to fuse the features across different channels.

4 Experiment and results

4.1 Underwater image datasets

4.1.1 Freshwater fish dataset

The experimental endeavor of this meticulous study was meticulously conducted employing two distinct species of freshwater fish, thereby substantiating the efficacious application of the pioneering DeformableFishNet framework. Presently, scholarly investigations into the realm of underwater fish identification predominantly revolve around marine dwellers, overlooking the rich biodiversity inhabiting freshwater ecosystems. Simultaneously, there exists a palpable dearth of comprehensive datasets capturing underwater fish imagery within their natural freshwater habitats.

In response to these challenges, this study took action by creating a freshwater fish dataset that authentically reflects the conditions found in real-world environments. Leveraging RGB camera technology submerged within the aqueous environs of the esteemed Guangzhou Pearl River Park located in the southern Chinese province of Guangdong, the research team captured high-quality images of fish populations thriving in their native freshwater milieu. The specimens featured in these photographic records represent two prominent species: the vibrant and ornamental koi fish, known for their striking colors and patterns, and the resilient tilapia, an economically significant fish species. Through such rigorous empirical efforts, this study not only

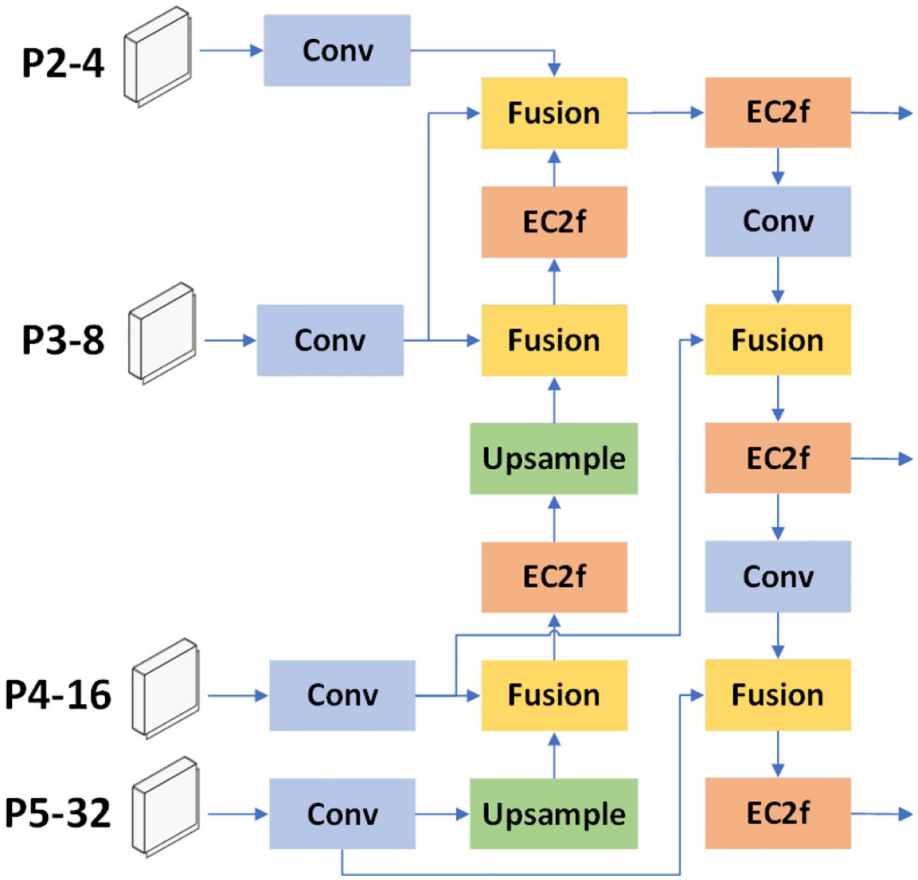

FIGURE 5
Structure of EDBFPN.

addresses a critical gap in the field but also promotes advancements in underwater fish recognition technologies applicable to diverse freshwater settings.

Tilapia and koi fish have different shapes, but they both belong to lateral flat fish, and their bodies have a certain streamlined structure, which helps them to move quickly in the water. The body color and texture of tilapia are relatively monotonous, not as colorful and ornamental as that of koi fish.

Tilapia is tall and flat, with a raised back and a slightly rounded abdomen. The dorsal fin has more than 10 fin spines, the caudal fin is flat or round, and there are obvious longitudinal stripes on the side and caudal fin. Mouth margin is not necessary. The body color of tilapia is usually grayish brown, and the edge of scales may be black. There will be six or seven black horizontal bands on the side of tilapia during the juvenile period, and the body color may become bright in the reproductive season. Koi fish has a typical spindle shape, plump figure, straight back line, slightly rounded abdomen, big mouth crack, a pair of tentacles on both sides of the corner of his mouth, dorsal fin and gluteal fin at the back, and caudal fin in a fan shape or Shuang Ye shape. The biggest feature of koi fish is its rich colors and unique stripes. Common colors are red, white, black, yellow, and blue, and there are various combinations of spots, patches, and lines. The color pattern of each koi fish is unique.

As shown in Figure 7, more than one fish is captured in each photo, and the fish is captured at various angles and brightness. The purpose is to ensure that the dataset is closest to the life state of fish under natural conditions. We used open-source image annotation tool "Labellmg" to create ground truth as shown in Figure 8, selecting 4,691 images for our dataset. Following a ratio of 7:1:2, we divided the images into training sets, validation sets, and testing sets.

4.1.2 Fish4Knowledge23 dataset

In order to verify the performance of DeformableFishNet in underwater fish detection, we apply DeformableFishNet to Fish4Knowledge23 dataset for experimental verification. The Fish4Knowledge23 collection is exclusively composed of underwater images of fish. Originating from a collaborative initiative between Taiwan Power Company, Taiwan Ocean Research Institute, and Kenting National Park, this dataset was meticulously compiled over a period spanning from October 1, 2010 to September 30th, 2013 at underwater monitoring stations strategically positioned in Taiwan's Nanwan Strait, Orchid Island, and the serene waters of Houbihu Lake.

Each and every fish image encapsulated within this dataset is carefully extracted from underwater video footage encompassing visual representations of 23 unique fish species, totaling an

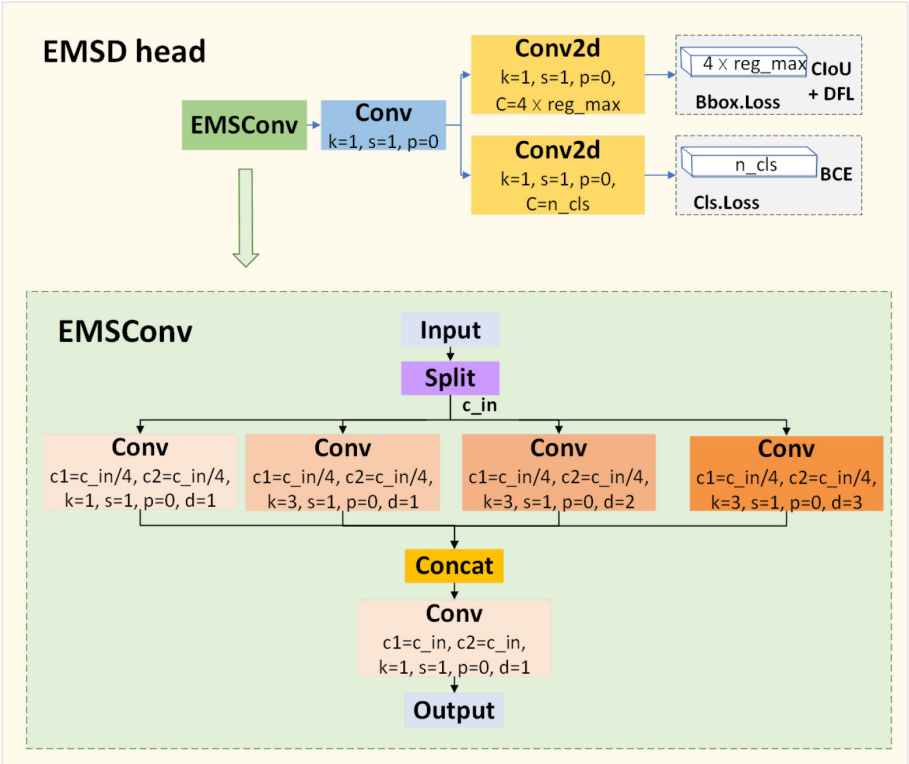

FIGURE 6
Structure of EMSD head.

FIGURE 7
Overview of the freshwater fish dataset. There are photos of tilapia and koi fish with different angles and brightness.

FIGURE 8
Dataset labeling. Tilapia is in the purple box. Koi fish is in the green box.

impressive 27,370 individual frames. However, it is worth noting that the resolution quality of these images is relatively low. The distribution of data is notably skewed, with the prevalence of certain fish species' images being approximately a thousand-fold greater than the rarest ones.

4.1.3 Brackish dataset

In our study, we deploy DeformableFishNet onto the challenging brackish dataset to extensively investigate and validate its prowess in detecting targets amidst the obscure and indistinctive depths of underwater environments. This particular dataset was meticulously gathered in the narrow straits of Northern Denmark, encapsulating a diverse array of marine life forms, including fish, crustaceans such as crabs, and various other aquatic creatures. Each dataset entry is meticulously annotated with precise bounding boxes demarcating the spatial locations of the targets. Comprising a total of 14,518 annotated images hosting a cumulative tally of 28,518 instances distributed across six distinct categories, the brackish dataset primarily emphasizes dimly lit and blurry underwater scenarios.

4.1.4 RUOD dataset

In order to comprehensively assess and affirm the versatile application potential of DeformableFishNet across a multitude of underwater scenarios, we subject it to rigorous testing on the expansive RUOD dataset. This dataset exemplifies a wide array of general underwater landscapes and encapsulates a myriad of underwater detection complexities that pose significant challenges to existing methodologies.

The RUOD dataset boasts a diverse array of target categories, ranging from schooling fish and diving humans to intricate marine flora and fauna such as starfish, vibrant corals, majestic sea turtles, spiny sea urchins, elongated sea cucumbers, bivalve mollusks like scallops, elusive cephalopods like squids, and ethereal jellyfish, cumulatively encompassing 10 distinct classes.

Table 1 presents the statistical data of underwater datasets.

4.2 Implementation details

For model training and inference, we utilized Ubuntu 20.04.6 LTS, an AMD EPYC 7543P 32-Core CPU processor, and CUDA 12.0. The graphics processing unit (GPU) used was NVIDIA RTX A5000 with 24 GB of memory. The network development framework employed was torch-2.0.1+cu117. The integrated development environment (IDE) used was PyCharm. We set the epoch to 400, batch size to 16, and image size to 640 × 640. The optimizer used was stochastic gradient descent (SGD) with an initial learning rate of 0.01 and weight decay of 0.0005.

TABLE 1 Statistical data of datasets.

Dataset	Image (num)	Label (num)	Species	Source	Size (pixel)
Freshwater fish	4,691	More than 4,691	2	Captured from fresh water	1,920 × 1,080
Fish4Knowledge	27,370	More than 27,370	23	Captured from open sea	320 × 320 and 640 × 640
Brackish	14,518	28,518	6	Captured from brackish water	Variable
RUOD	74,903	More than 74,903	10	Consists of multiple datasets, such as URPC, UDD, DUO, and UODD	Variable

4.3 Evaluation metrics

We have chosen precision (*P*), recall (*R*), F1-score (F1), mean average precision (mAP), parameters, floating-point operations (FLOPs), and frames per second (FPS) as the comparative metrics to evaluate the detection performance and determine the strengths and weaknesses of the model. Using IoU = 0.5 as the standard, precision and recall can be calculated using the following formulas Equations 1, 2:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{1}$$

$$\text{Recall} = \frac{TP}{TP + PN} \tag{2}$$

TP represents the number of true positive samples correctly identified as positive, FP represents the number of false positive samples incorrectly classified as positive, and FN represents the number of false negative samples incorrectly classified as negative. F1-score is the harmonic average of precision and recall, which is used to comprehensively consider the performance of classifier. The value range of F1-score is between 0 and 1, where 1 represents the perfect classifier and 0 represents the worst classifier. mAP_{50} is the area under the precision–recall (PR) curve formed by precision and recall. For $mAP_{50:95}$, the area under the PR curve is calculated by dividing it into 10 IoU thresholds ranging from 0.5 to 0.05 to 0.95 and then taking the average of the results. FPS represents the number of images detected by the model per second.

4.4 Experiment results

4.4.1 Ablation experiment

In the YOLO-based object detector, we validated the effectiveness of each proposed module. We conducted ablation experiments on the proposed modules in YOLOv8. Table 2 presents the results of the ablation experiments of the EGCA module. After YOLOv8 added DCNv2 or EDCN, the FPS of the model decreased, but the detection accuracy of the model is improved. Comparing YOLOv8 + EDCN (EGCA + DCNv2) and YOLOv8 + DCNv2, the *R* of YOLOv8 + EDCN (EGCA + DCNv2) is lower, but *P* and F1 are higher. In particular, the $mAP_{50:95}$ of YOLOv8 + EDCN (EGCA + DCNv2) is as high as 79.1%. Experimental results show that the EGCA module is effective.

Table 3 presents the results of the ablation experiments of all modules. The variations in various loss functions during the training process are illustrated in Figure 9. We can find that the proposed modules have played a significant role in improving the performance of the model. When the EC2f module was added to the backbone network of YOLOv8, *P* reached its highest value of 92.6%, mAP_{50} reached its highest value of 96.6%, and $mAP_{50:95}$ reached its highest value of 79.1%. This indicates that the proposed EC2f module adapts well to the changing poses of swimming fish. When only the EDBFPN module was added to the original YOLOv8, all performance metrics of the model improved, and the number of parameters and FLOPs decreased. This suggests that the EDBFPN module effectively combines performance improvement and model lightweighting. When only the EMSD head was added to the original YOLOv8, the number of parameters decreased to 2.6M, and FLOPs decreased to 5.8G. This shows that the EMSD head module effectively reduces the number of parameters and FLOPs in the model.

When the different models are combined together, the improved model shows varying degrees of improvement in all performance metrics compared to the original YOLOv8 while also reducing the number of parameters and FLOPs to varying degrees. When both EDBFPN and EMSD head were added to YOLOv8, the mAP_{50} decreased by 1%, but the parameter count reduced to the optimal value of 1.7M. Finally, when we integrate all our modules into YOLOv8, the model experiences a significant reduction in parameter count and FLOPs while maintaining excellent detection performance. The performance metrics of DeformableFishNet still outperform the original YOLOv8.

4.4.2 Experimental results of different models

In pursuit of a comprehensive and stringent comparison of model performance under authentic circumstances, we employed the widely recognized COCO evaluation metrics to scrutinize the comparative merits and deficiencies across a variety of models. Central to this evaluation is the average precision (AP), a metric derived from the precision–recall curve, harmoniously integrating the dual facets of precision and recall. COCO adopts mean average precision (mAP) as the principal gauge of overall model efficacy, which is achieved by averaging the AP scores across all object categories. It is worth noting that COCO further incorporates the intersection over union (IoU) threshold concept, wherein AP calculations are performed for a range of IoU levels and reported as AP@[0.5:0.05:0.95]. This signifies that the AP is computed at IoU thresholds incrementally progressing from 0.5 to 0.95 in 0.05 intervals before being averaged, thereby furnishing a holistic reflection of model performance across a spectrum of localization difficulties. Given the substantial variation in object sizes within the COCO datasets, the evaluation metrics take into account the diverse dimensions of targets, segmenting them into small, medium, and

TABLE 2 Results of the ablation experiments of the EGCA module.

Method	*P*	*R*	F1	mAP_{50}	mAP50:95	Parameters	FLOPs	FPS
YOLOv8	91.1%	90.4%	91.0%	96.0%	76.6%	3.2M	8.7G	**709**
YOLOv8 + DCNv2	91.0%	**91.5%**	91.0%	96.5%	77.7%	**3.0M**	8.0G	684
YOLOv8 + EDCN (EGCA + DCNv2)	**92.6%**	90.8%	**92.0%**	**96.6%**	**79.1%**	3.1M	**7.9G**	625

The bold values are the optimal values for this column.

TABLE 3 Results of the ablation experiments of all modules.

YOLOv8	EC2f	EDBFPN	EMSD head	*P*	*R*	F1	mAP_{50}	$mAP_{50:95}$	Parameters	FLOPs	FPS
✓				91.1%	90.4%	91.0%	96.0%	76.6%	3.2M	8.7G	**709**
✓	✓			**92.6%**	90.8%	**92.0%**	**96.6%**	**79.1%**	3.1M	7.9G	625
✓		✓		90.4%	92.0%	91.0%	96.3%	77.8%	2.0M	7.1G	434
✓			✓	91.2%	91.1%	91.0%	96.3%	77.6%	2.6M	5.8G	361
✓	✓	✓		91.3%	**92.2%**	**92.0%**	**96.6%**	78.6%	2.1M	6.9G	476
✓	✓		✓	92.0%	91.8%	**92.0%**	**96.6%**	77.6%	2.7M	5.6G	338
✓		✓	✓	90.6%	91.8%	91.0%	95.9%	77.6%	**1.7M**	4.9G	357
✓	✓	✓	✓	92.0%	91.0%	**92.0%**	96.3%	78.1%	**1.7M**	**4.7G**	350

The bold values are the optimal values for this column.

large categories and computing the corresponding AP values. This ensures a balanced assessment of model competence across all target sizes. Moreover, COCO also presents average recall (AR) with a predetermined limit on the number of detections, offering insights into how effectively a model can identify true positives when constrained by a finite number of predicted bounding boxes, thereby providing a complementary perspective on model performance.

Table 4 shows the comparison of our proposed method with recently proposed methods in terms of COCO metrics. Sparse R-CNN is a two-stage algorithm. RTMDet, YOLOv7, YOLOv8, and YOLOv9 are one-stage algorithms, and both DINO and RT-DETR are transformer-based algorithms.

It can be clearly observed that DeformableFishNet achieves the highest average precision (AP) in various scenarios. In the case of an IoU range from 0.50 to 0.95, DeformableFishNet achieves an AP of 75.2%, representing an 11.1% increase over the performance of Sparse R-CNN in the same IoU range and 0.4% higher than the second-best YOLOv8. When considering a maximum detection

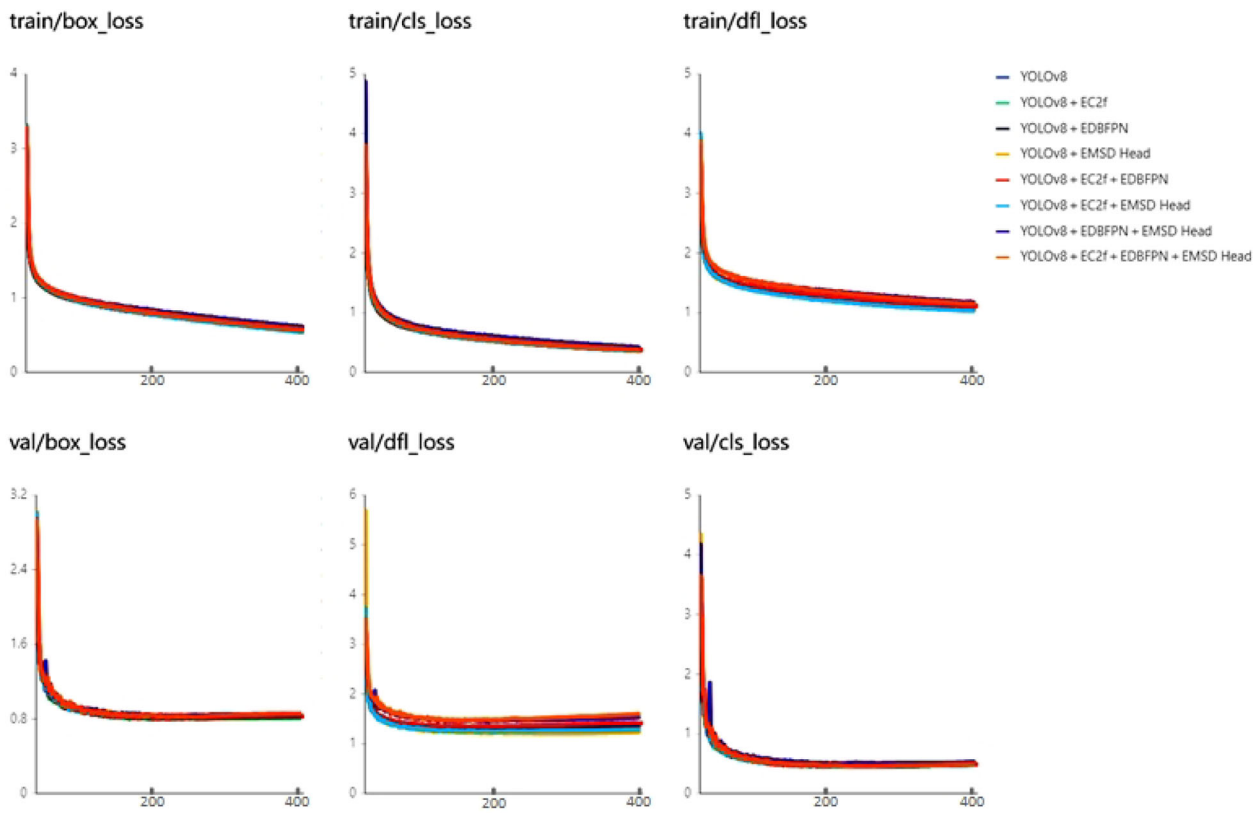

FIGURE 9
Loss function during the training process of the ablation experiment. The horizontal axis in the figure represents the number of training epoch.

TABLE 4 Results of comparative experiments.

Model	$AP_{IoU=0.50}$	$AP_{IoU=0.75}$	$AP_{IoU=0.50:0.95}$	$AR_{maxDets=1}$	$AR_{maxDets=10}$	$AR_{maxDets=100}$	Parameters	FLOPs
Sparse R-CNN-r50	90.2%	73.3%	64.1%	45.8%	75.8%	78.6%	2.5M	7.5G
DINO-4scale	94.6%	86.0%	73.9%	49.5%	80.6%	83.2%	47M	279G
RT-DETR-r18	95.2%	86.7%	74.6%	50.8%	80.1%	**82.8%**	20M	60G
RTMDet-l	94.9%	84.9%	72.9%	48.9%	79.5%	80.3%	52.3M	160.3G
YOLOv7-l	95.2%	86.8%	74.5%	**51.0%**	**81.4%**	81.6%	37.6M	106G
YOLOv8-n	95.1%	87.3%	74.8%	50.6%	80.7%	80.8%	3.2M	8.7G
YOLOv9-c	95.1%	86.2%	73.5%	49.9%	80.2%	80.6%	25.5M	102.8G
DeformableFishNet	**95.4%**	**87.7%**	**75.2%**	**51.0%**	80.6%	80.6%	**1.7M**	**4.7G**

count of 10, DeformableFishNet's Average Recall (AR) is marginally lower compared to YOLOv7. Conversely, with a maximum detection count of 100, DINO surpasses all other algorithms. Nevertheless, across all evaluated metrics, DeformableFishNet has the best detection performance. The parameters of DeformableFishNet is only 1.7M, and the FLOPs is only 4.7G. In general, DeformableFishNet demonstrates superiority over other algorithms.

4.4.3 Experimental results in different underwater datasets

In Figure 10, it can be found that DeformableFishNet is effective on freshwater fish dataset. Table 5 shows the experimental results of our model on the proposed freshwater fish dataset. DeformableFishNet has achieved excellent results in many indicators such as *P*, *R*, and mAP. DeformableFishNet achieved *P* of 92.0%, *R* of 91.0%, mAP_{50} of 96.3%, and $mAP_{50:95}$ of 78.1%. Figure 11 is a normalized confusion matrix. For koi, the model correctly identified them as koi (true positive) in 94% cases. However, in the remaining 6% cases, it mistook koi for other kinds of fish (false negative). For tilapia, the model correctly identified them as tilapia (true positive) in 94% cases, but in 20% cases, it mistook tilapia for koi (false positive). For background, the model successfully identifies it as background (true negative) in 94% cases, but in 5% cases, it mistook the background for tilapia (false positive). DeformableFishNet performs well in identifying koi and tilapia, but there are some errors in identifying the background. This may require further optimization to reduce false positives.

The experimental findings pertain to the comparative detection results for tilapia and koi fish. Upon scrutiny, DeformableFishNet exhibits a higher aptitude for accurately detecting tilapia relative to koi fish. This differential performance could potentially be attributed to the unique physical attributes of these species. Koi fish, renowned for their vivid colors and intricate textures, present a broader and more diverse set of visual features. Each individual koi displays a distinctive pattern and hue arrangement, which might

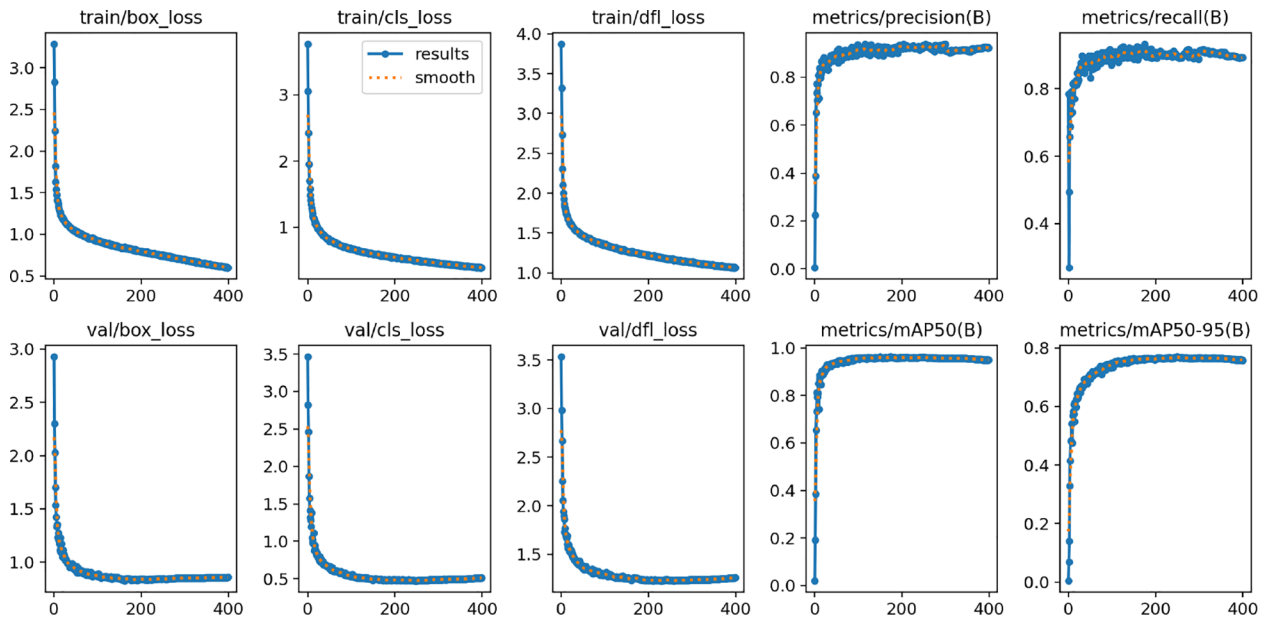

FIGURE 10
Performance values in the freshwater fish dataset.

TABLE 5 Results of DeformableFishNet in the freshwater fish dataset.

Class	*P*	*R*	F1	mAP_{50}	$mAP_{50:95}$
All	92.0%	91.0%	92.0%	96.3%	78.1%
Koi	90.2%	89.3%	89.7%	95.0%	74.6%
Tilapia	93.8%	92.7%	93.2%	97.6%	81.6%

necessitate a more sophisticated recognition process. On the whole, DeformableFishNet can be well applied to the detection of freshwater fish. Figures 12, 13 show the detection results of DeformableFishNet on the freshwater fish dataset.

To substantiate the versatility and robustness of DeformableFishNet in the realm of fish recognition, we applied DeformableFishNet to Fish4Knowledge23 dataset, and the results are shown in Table 6. In Figure 14, it can be found that DeformableFishNet is effective on Fish4Knowledge23 dataset. We can find that *P* reaches 96.7%, *R* reaches 97.6%, mAP_{50} reaches 99.4%, and $mAP_{50:95}$ reaches 85.5%. Displaying DeformableFishNet's consistency and reliability across a broad spectrum of IoU thresholds. DeformableFishNet has achieved excellent results in Fish4Knowledge23 dataset. In all kinds of fish, the index values have reached above 90.0%. The highest mAP_{50} value of 99.5% was achieved in the detection of various fish species such as *Myripristis kuntee*, *Amphiprion clarkia*, and *Plectroglyphidodon dickii*. In the detection of *Neoniphon samara*, the mAP_{50} value reached the lowest at 97.6%.

Figure 15 is a normalized confusion matrix. In the first line (*Dascyllus reticulatus*), the model can almost always correctly identify this fish and only in a few cases misjudged it as other species or backgrounds. In the second row (*Myripristis kuntee*), the model can perfectly identify this kind of fish. In the next few lines, it also shows a similarly high accuracy, such as for *Amphiprion clarkia*, *Plectroglyphidodon dickii*, etc.

However, in line 12 (*Scolopsis bilineata*), although the model can correctly identify in most cases, some are misjudged as other types, such as *Ncoglyphidodon nigroris* and *Zancius cornutus*. The last line represents the background, and it can be seen that the model rarely misjudges any kind of fish as the background, which shows that it has a strong ability to distinguish between fish and background.

DeformableFishNet is excellent in identifying most fish species, but there may be confusion between certain species—for example, *Scolopsis bilineata* is sometimes misjudged as NCglyphodon Niger or *Zancius cornutus*. In addition, the model rarely makes mistakes in judging the background, which shows that it has good background removal ability. Figure 16 shows the detection results of DeformableFishNet on Fish4Knowledge23.

We engaged the DeformableFishNet model in a rigorous evaluation using the brackish dataset, aiming to authenticate its efficacy under the challenging conditions of murky and indistinct underwater scenes. In Figure 17, it can be found that DeformableFishNet is effective on brackish dataset. The experimental results are presented in Table 7. In the brackish

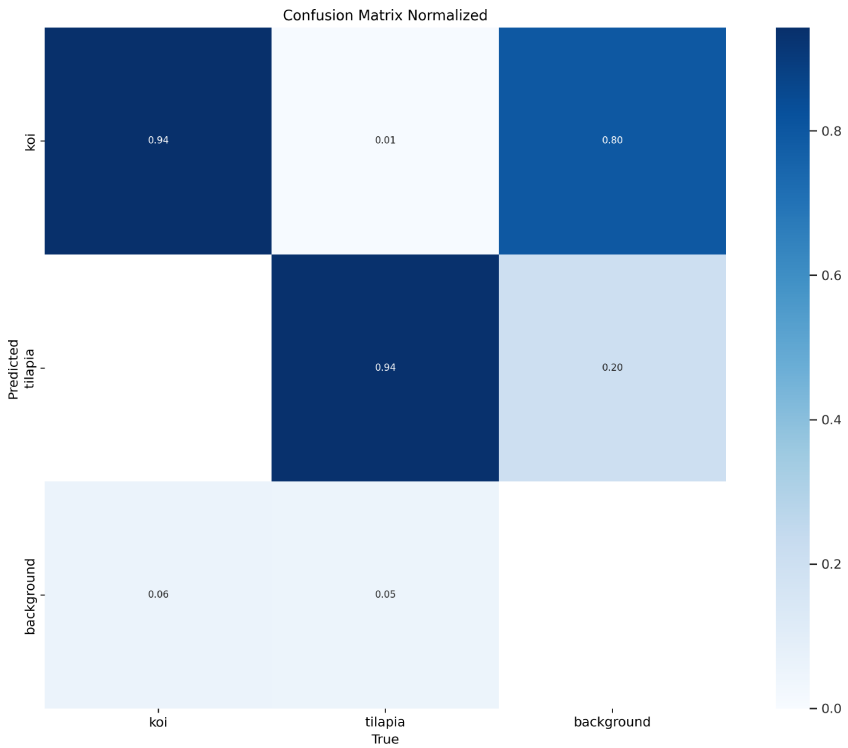

FIGURE 11
Confusion matrix in the freshwater fish dataset.

FIGURE 12
Detection results of DeformableFishNet in the freshwater fish dataset.

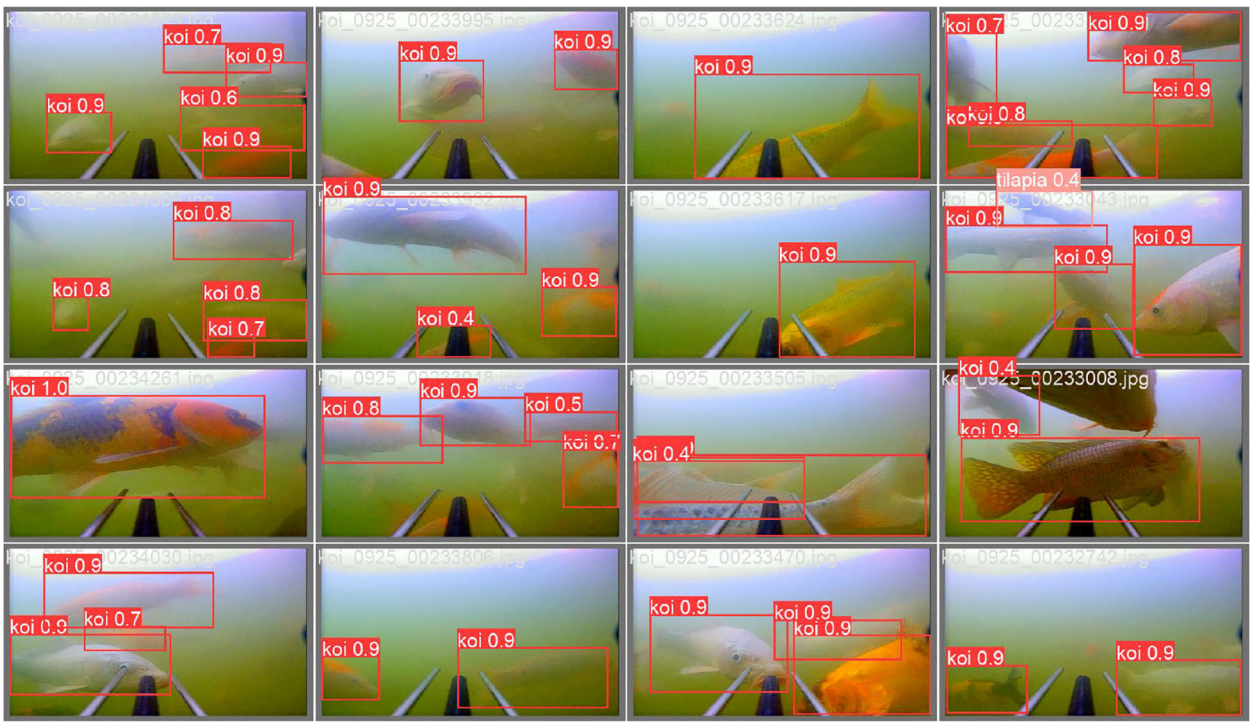

FIGURE 13
Detection results of DeformableFishNet in the freshwater fish dataset.

10.3389/fmars.2024.1424619

TABLE 6 Results of DeformableFishNet in the Fish4Knowledge23 dataset.

Class	*P*	*R*	F1	mAP_{50}	$mAP_{50:95}$
All	96.7%	97.6%	97.1%	99.4%	85.5%
Dascyllus reticulatus	98.2%	96.9%	97.5%	99.2%	86.7%
Myripristis kuntee	99.5%	100%	99.2%	99.5%	85.8%
Amphiprion clarkia	99.8%	100%	99.9%	99.5%	79.5%
Plectroglyphidodon dickii	100%	99.7%	100%	99.5%	79.6%
Chromis chrysura	100%	98.9%	99.4%	99.5%	82.1%
Lutjanus fulvus	95.2%	94.7%	94.9%	99.3%	78.4%
Pomacentrus moluccensis	99.0%	100%	99.2%	99.5%	90.9%
Abudefduf vaigiensis	98.2%	100%	98.6%	99.5%	80.2%
Zebrasoma scopas	97.2%	100%	98.3%	99.5%	84.9%
Chaetodon trifascialis	98%	100%	98.7%	99.5%	85.6%
Acanthurus nigrofuscus	91.8%	97.0%	94.3%	99.2%	88.6%
Siganus fuscescens	86.8%	100%	92.9%	99.5%	89.5%
Canthigaster valentine	96.7%	100%	98.3%	99.5%	77.0%
Balistapus undulates	95.7%	100%	97.8%	99.5%	92.4%
Hemigymnus melapterus	93.6%	100%	96.7%	99.5%	83.3%
Scolopsis bilineata	96.5%	100%	98.2%	99.5%	84.8%
Ncoglyphidodon nigroris	100%	63.8%	78.0%	99.5%	93.1%
Scaridae	95.9%	100%	97.9%	99.5%	88.0%
Hemigymnus fasciatus	98.9%	100%	99.4%	99.5%	85.5%
Chaetodon lunulatus	99.8%	99.0%	99.3%	99.5%	83.9%
Pempheris vanicolensis	87.3%	100%	93.2%	99.5%	99.5%
Neoniphon samara	99.2%	96.8%	98.0%	98.0%	82.0%

dataset, DeformableFishNet achieved *P* of 98.0%, *R* of 96.6%, and mAP_{50} of 98.0% as well as $mAP_{50:95}$ of 81.2%. The mAP_{50} of DeformableFishNet in detecting starfish and crab reached 99.5%, which was better than that of other marine organisms. DeformableFishNet not only performs well in fish detection but also evidences its adaptability and effectiveness in recognizing and detecting non-fish marine species, such as crabs and starfish. Although DeformableFishNet recorded its lowest mAP_{50} score in detecting small fish compared to other creature categories, it still maintained 95.4% mAP_{50} in this regard.

Figure 18 is a normalized confusion matrix. The model correctly identified the fish in 98% of the cases and in only 2% of the cases misjudged the fish as other categories. For small fish, the model correctly identified small fish in 93% of the cases, and in the remaining 7% of the cases, most of them were misjudged as fish and a small part as background. For crab, the model correctly identified the crab in 99% of the cases and only misjudged the crab as other categories in 1% of the cases. For shrimp, the model correctly identified shrimp in 98% cases and misjudged shrimp as other categories in 2% cases. For jellyfish, the model correctly identified jellyfish in 93% cases and misjudged jellyfish as other categories in 7% cases. The model has a good detection effect on starfish. The model correctly identified starfish in 100% cases, and there was no misjudgment. Generally speaking, DeformableFishNet performs well in identifying various aquatic organisms, especially fish, crab, shrimp, and starfish. For small fish and jellyfish, although there is a high accuracy, there are still some misjudgments.

Figure 19 shows the detection outcomes of DeformableFishNet on the brackish dataset. DeformableFishNet excels in detecting objects in dark and blurry underwater environments and demonstrates equally commendable performance in the detection of other marine life forms.

In order to thoroughly assess the adaptability and effectiveness of the DeformableFishNet model across a multitude of underwater settings, we executed a series of rigorous tests harnessing the RUOD dataset. In Figure 20, it can be found that DeformableFishNet is effective on RUOD dataset. The comprehensive experimental outcomes are encapsulated in Table 8, illustrating that DeformableFishNet secured a noteworthy *P* of 86.3%, *R* of 75.6%,

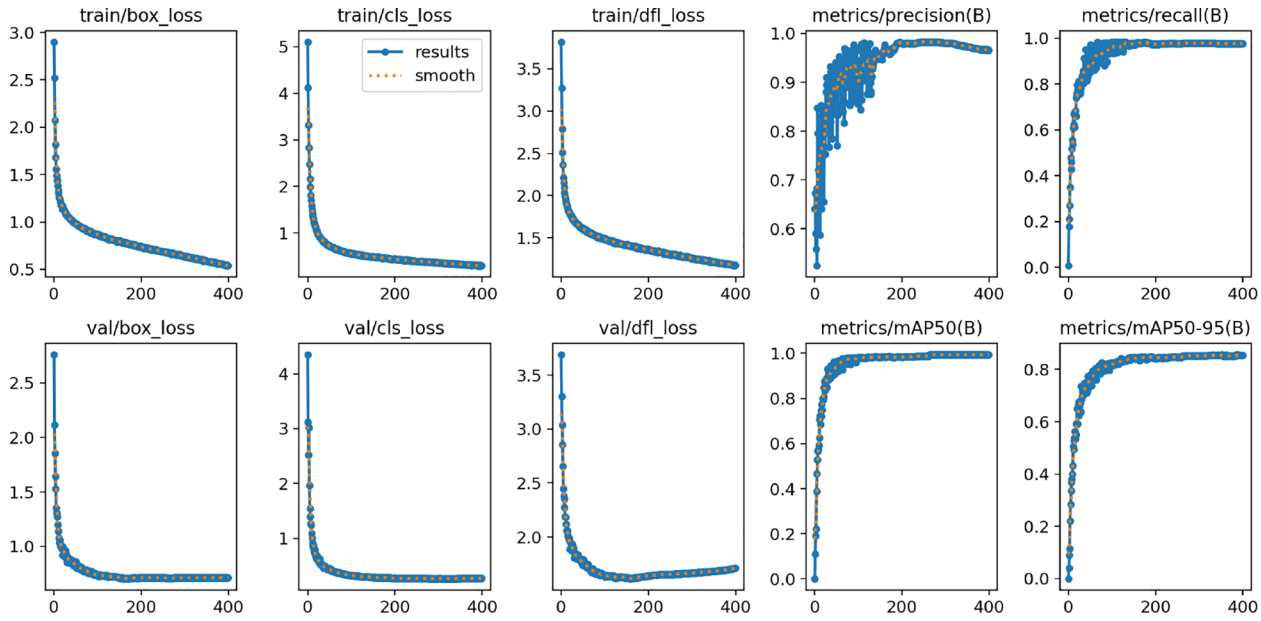

FIGURE 14
Performance values in the Fish4Knowledge23 dataset.

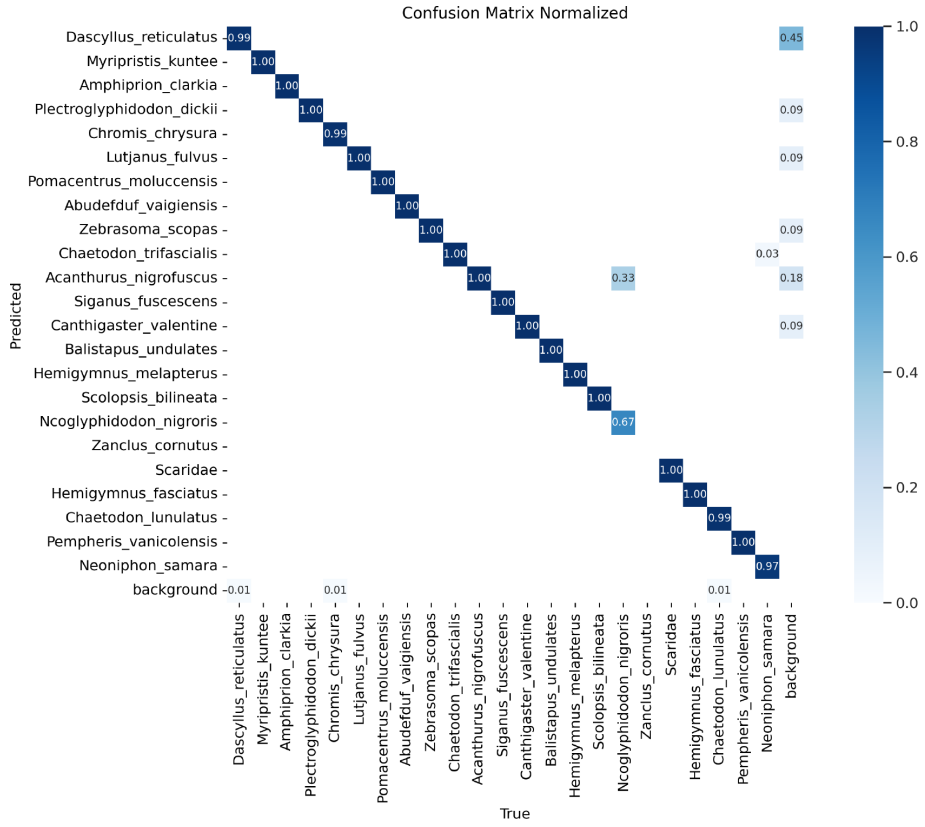

FIGURE 15
Confusion matrix in the Fish4Knowledge23 dataset.

FIGURE 16
Detection results in the Fish4Knowledge23 dataset.

and mAP_{50} of 83.9%. Concurrently, the model achieved $mAP_{50:95}$ of 60.4% on the diverse and challenging RUOD dataset.

A highlight from these results was DeformableFishNet's superior performance in the detection of cuttlefish, where it reached its pinnacle mAP_{50} score of 96.7%. Given the RUOD dataset's absence of granular classifications for fish species, the model faced its most daunting challenge in the "fish" category, recording a minimum mAP_{50} of 66.1%. Nevertheless, this datum underscores the robustness of DeformableFishNet even in the presence of less defined categories.

Figure 21 shows the performance of the model in identifying underwater creatures. The darker the color, the higher the accuracy, while the lighter the color, the lower the accuracy. In the RUOD data, DeformableFishNet performs well in identifying sea cucumbers, sea urchins, scallops, starfish, corals, divers, cuttlefish, and turtles, but there are some misjudgments in identifying fish and jellyfish. At the same time, the model easily misjudges the background as jellyfish and turtles. Figure 22 shows the detection results of DeformableFishNet on the RUOD dataset. DeformableFishNet exhibits performance in the arduous task of underwater object detection under a wide array of environmental conditions.

TABLE 7 Results of DeformableFishNet in the brackish dataset.

Class	*P*	*R*	F1	mAP_{50}	$mAP_{50:95}$
All	98.0%	96.6%	97.0%	98.0%	81.2%
Fish	98.8%	98.0%	98.4%	99.1%	86.6%
Small fish	94.1%	91.5%	92.8%	95.4%	68.8%
Crab	99.2%	99.0%	99.1%	99.5%	88.0%
Shrimp	96.5%	97.8%	97.2%	98.5%	73.7%
Jellyfish	99.6%	93.3%	96.4%	95.9%	72.1%
Starfish	99.9%	99.7%	99.8%	99.5%	98.1%

In this study, DeformableFishNet performed well on freshwater fish dataset. Compared with other target detection algorithms, DeformableFishNet achieves higher detection accuracy with lower parameters and floating-point computation. In the open underwater datasets, although DeformableFishNet has some misjudgments, it also has high accuracy. Overall, DeformableFishNet not only performs well in fish detection but also performs well in identifying various aquatic organisms.

There is still much room for improvement in this research. DeformableFishNet is designed based on the characteristics of freshwater fish dataset. However, the number of samples in freshwater fish dataset is limited, and there are few fish species. DeformableFishNet performs well on several datasets, but the actual underwater environment is complex and changeable, and the diversity of fish posture, size, Color, and background may exceed the coverage of the training data. Future research can further enhance the generalization ability of the model, for example, by introducing more diverse training data, adopting data

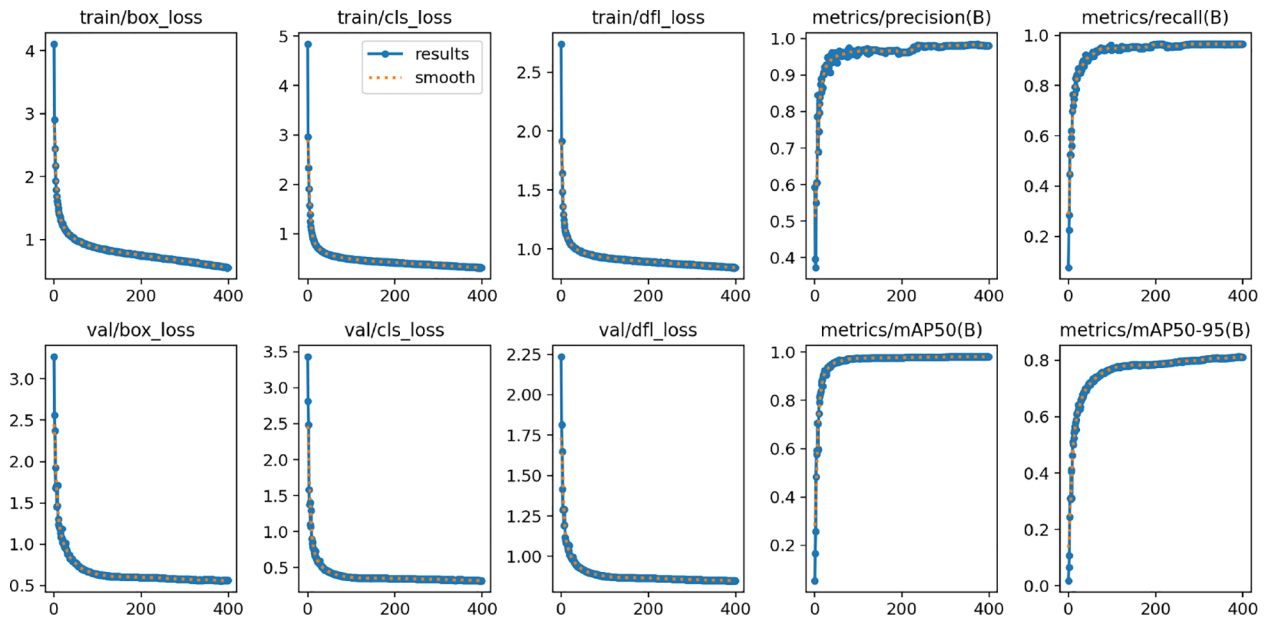

FIGURE 17
Performance values in the brackish dataset.

enhancement technology, or designing more robust feature representation methods.

Underwater images often face problems such as blur, large illumination variation, and turbid water quality, which may affect the recognition accuracy. Developing a feature extraction method that can resist these interference factors or incorporating a specific adaptive mechanism into the model will be the key to improve the recognition robustness.

DeformableFishNet is an improvement on YOLOv8. According to the official test, YOLOv8 can realize real-time detection on high-

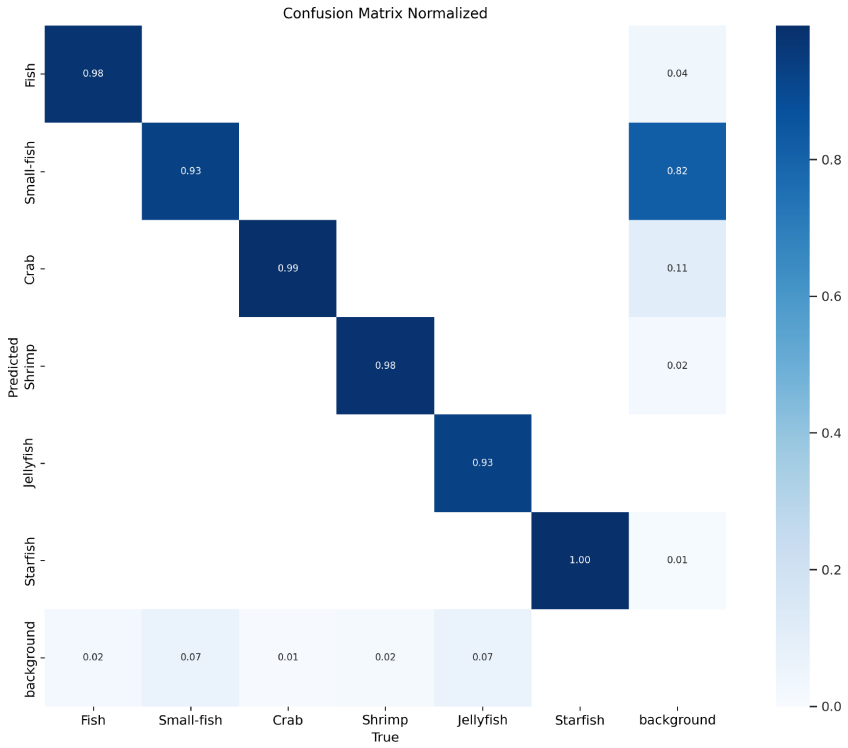

FIGURE 18
Confusion matrix in the brackish dataset.

FIGURE 19
Detection results in the brackish dataset.

definition video stream. Although DeformableFishNet has paid attention to the lightweight of the model, it may be necessary to reduce the computational cost (FLOPs) or improve the reasoning speed in some application scenarios. DeformableFishNet needs to be tested and adjusted according to the actual situation of edge devices. Exploring a more efficient network structure design, model pruning, quantization technology, or hardware acceleration scheme is the future improvement direction.

TABLE 8 Results of DeformableFishNet in the RUOD dataset.

Class	*P*	*R*	F1	mAP_{50}	$mAP_{50:95}$
All	86.3%	75.6%	80.0%	83.9%	60.4%
Holothurian	85.0%	67.1%	75.6%	78.5%	48.5%
Echinus	92.3%	83.2%	87.7%	92.3%	54.5%
Scallop	85.4%	63.2%	73.8%	79.5%	51.3%
Starfish	88.7%	79.7%	84.1%	88.0%	55.3%
Fish	80.2%	56.5%	67.3%	66.1%	46.3%
Corals	78.2%	66.0%	71.9%	72.5%	53.0%
Diver	90.1%	91.4%	90.8%	94.4%	76.2%
Cuttlefish	93.7%	93.9%	93.8%	96.7%	81.5%
Turtle	93.5%	89.3%	91.4%	94.7%	81.2%
Jellyfish	75.6%	65.3%	70.3%	73.1%	56.2%

5 Conclusion

In this paper, we proposed a new network structure named DeformableFishNet. First, we design an EGCA attention module and combine it with deformable convolution to introduce EDCN and EC2f convolution modules as the backbone and neck units of the new network. Second, to better extract features, perform feature fusion, and lightweight the neural network, we propose the EDBFPN. Finally, aiming to maintain high performance and maximize lightweightness for deployment on edge devices for fish detection and classification, we redesign an EMSD head that obtains multi-scale feature maps through inexpensive convolution operations.

The experimental results show that the performance of DeformableFishNet is obviously better than many algorithms. The experimental results show that all the modules we proposed have achieved good results. In our freshwater fish dataset, the mAP_{50} of the DeformableFishNet has achieved 96.3% and the $mAP_{50:95}$ has achieved 78.1%. In three public underwater datasets, the DeformableFishNet got 98%, 99.4%, and 83.6% mAP_{50}, respectively. The parameters of DeformableFishNet is 1.7M, and the FLOPs is 4.7G. Compared with the target detection algorithm YOLOv8, the proposed model parameters are reduced by 1.5M and the FLOPs by 4G. DeformableFishNet is suitable for deployment on edge devices to achieve real-time underwater fish detection and classification. DeformableFishNet not only achieves high accuracy in the recognition task but also is friendly to edge devices. This study is expected to promote the application of fish identification method based on deep learning in production.

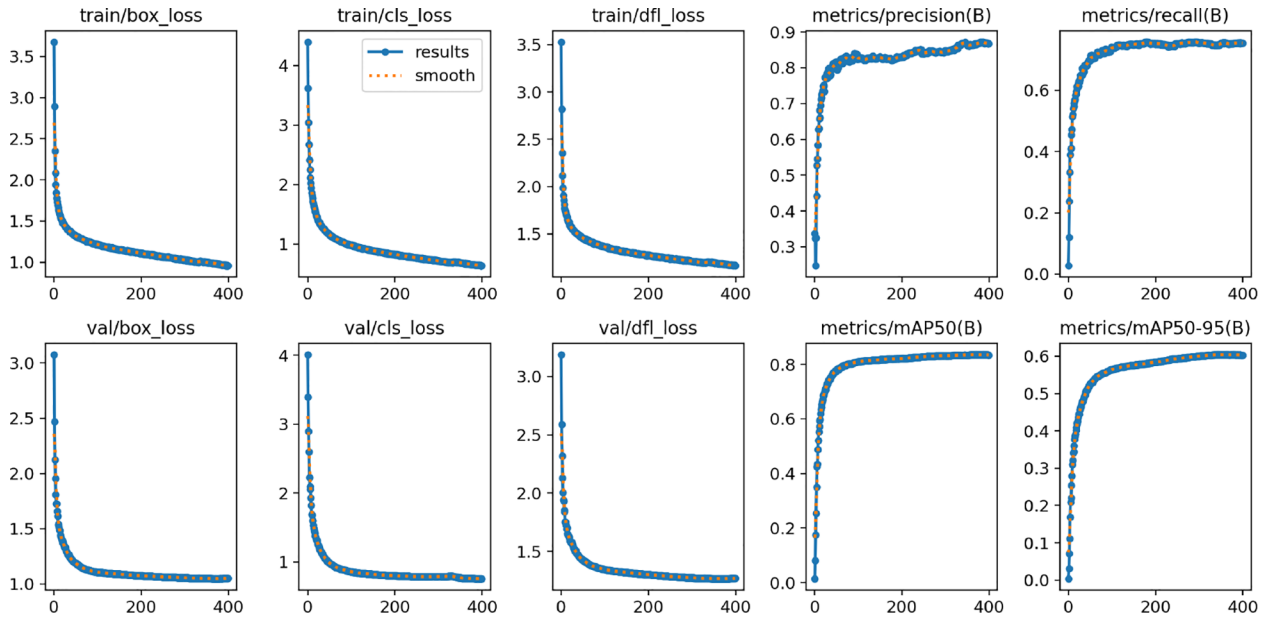

FIGURE 20
Performance values in the RUOD dataset.

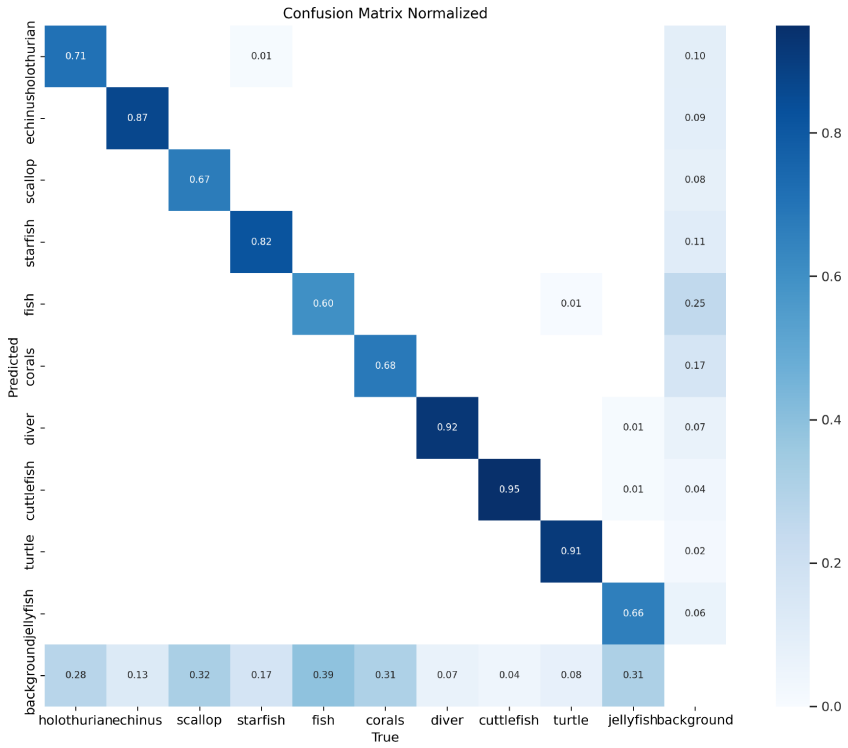

FIGURE 21
Confusion matrix in the RUOD dataset.

FIGURE 22
Detection results in the RUOD dataset.

In the future, we will continue to expand the freshwater fish dataset, planning to include underwater images of freshwater fish from more species and ecological environments to significantly enhance the diversity and comprehensiveness of the dataset. We will further optimize the model according to the resource consumption during the actual deployment of edge devices. In addition, we will devote ourselves to developing a series of advanced technical models, such as real-time fish tracking system, efficient fish quantity estimation method, and methods for fish disease identification, all of which will provide strong data and technical support for fishery resource management, ecological protection, and related scientific research fields.

Data availability statement

The original contributions presented in the study are included in the article/supplementary materials, further inquiries can be directed to the corresponding author/s.

Ethics statement

Ethical approval was not required for the study involving animals in accordance with the local legislation and institutional requirements because in this study, we only need to use underwater cameras to take pictures of fish activities in natural underwater environment. This study does not need to be released directly or indirectly from animals.

Author contributions

ZR: Conceptualization, Methodology, Validation, Writing – original draft, Writing – review & editing. ZW: Writing – review & editing. YH: Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This research was funded by the Guangzhou Science and Technology Plan Project (202201011835).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Allken, V., Handegard, N. O., Rosen, S., Schreyeck, T., Mahiout, T., and Malde, K. (2018). Fish species identification using a convolutional neural network trained on synthetic data. *ICES Journal of Marine Science* 76 (1), 342–349. doi: 10.1093/icesjms/fsy147

Banan, A., Nasiri, A., and Taheri-Garavand, A. (2020). Deep learning-based appearance features extraction for automated carp species identification. *Aquacultural Engineering* 89, 102053. doi: 10.1016/j.aquaeng.2020.102053

Ben Tamou, A., Benzinou, A., and Nasreddine, K. (2021). Multi-stream fish detection in unconstrained underwater videos by the fusion of two convolutional neural network detectors. *Appl. Intell.* 51, 5809–5821. doi: 10.1007/s10489-020-02155-8

Bochkovskiy, A., Wang, C.-Y., and Liao, H.-Y. (2020). Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.

Cai, K., Miao, X., Wang, W., Pang, H., Liu, Y., and Song, J. (2020). A modified yolov3 model for fish detection based on mobilenetv1 as backbone. *Aquacultural Engineering* 91, 102117. doi: 10.1016/j.aquaeng.2020.102117

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. (2020). End-to-End Object Detection with Transformers. *European conference on computer vision*, 213–229. doi: 10.1007/978-3-030-58452-813

Chang, Z., Liu, S., Xiong, X., Cai, Z., and Tu, G. (2021). A survey of recent advances in edge-computing-powered artificial intelligence of things. *IEEE Internet of Things Journal* 8, 13849–13875. doi: 10.1109/jiot.2021.3088875

Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., et al. (2017). "Deformable convolutional networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 764–773 doi: 10.1109/iccv.2017.89

Deng, S., Zhao, H., Fang, W., Yin, J., Dustdar, S., and Zomaya, A. Y. (2020). Edge intelligence: The confluence of edge computing and artificial intelligence. *IEEE Internet Things. J.* 7 (8), 7457–7469. doi: 10.1109/jiot.2020.2984887

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Ge, Z., Liu, S., Wang, F., Li, Z., and Sun, J. (2021). Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*. doi: 10.48550/arXiv.2107.08430

Han, K., Wang, Y., Tian, Q., Guo, J., Xu, C., and Xu, C. (2020). "Ghostnet: More features from cheap operations," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1577-1586. doi: 10.1109/cvpr42600.2020.00165

Hou, Q., Zhou, D., and Feng, J. (2021). "Coordinate attention for efficient mobile network design," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 13708–13717. doi: 10.1109/cvpr46437.2021.01350

Howard, A., Sandler, M., Chen, B., Wang, W., Chen, L.-C., Tan, M., et al. (2019). "Searching for mobilenetv3," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 1314-1324. doi: 10.1109/iccv.2019.00140

Howard, A., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., et al. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*. doi: 10.48550/arXiv.1704.04861

Jiang, N., Sheng, B., Li, P., and Lee, T.-Y. (2023). Photohelper: Portrait photographing guidance via deep feature retrieval and fusion. *IEEE Trans. Multimedia.* 25, 2226–2238. doi: 10.1109/tmm.2022.3144890

Knausgård, K. M., Wiklund, A., Sørdalen, T. K., Halvorsen, K. T., Kleiven, A. R., Jiao, L., et al. (2022). Temperate fish detection and classification: a deep learning based approach. *Applied Intelligence* 52 (6), 6988–7001. doi: 10.1007/s10489-020-02154-9

Labao, A. B., and Naval, P. C. (2019). Cascaded deep network systems with linked ensemble components for underwater fish detection in the wild. *Ecological Informatics* 52, 103–121. doi: 10.1016/j.ecoinf.2019.05.004

Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., et al. (2022a). Yolov6: A single-stage object detection framework for industrial applications. *arXiv preprint arXiv:2209.02976*. doi: 10.48550/arXiv.2209.02976

Li, J., Chen, J., Sheng, B., Li, P., Yang, P., Feng, D. D., et al. (2022b). Automatic detection and classification system of domestic waste via multimodel cascaded convolutional neural network. *IEEE Transactions on Industrial Informatics* 18 (1), 163–173. doi: 10.1109/tii.2021.3085669

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., et al. (2021). "Swin transformer: Hierarchical vision transformer using shifted windows," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 9992-10002. doi: 10.1109/iccv48922.2021.00986

Mao, Q.-C., Sun, H.-M., Liu, Y.-B., and Jia, R.-S. (2019). Mini-yolov3: Real-time object detector for embedded applications. *IEEE Access* 7, 133529–133538. doi: 10.1109/access.2019.2941547

Mathur, M., Vasudev, D., Sahoo, S., Jain, D., and Goel, N. (2020). Crosspooled fishnet: transfer learning based fish species classification model. *Multimedia. Tools Appl.* 79, 31625–31643. doi: 10.1007/s11042-020-09371-x

Prasetyo, E., Suciati, N., and Fatichah, C. (2022). Multi-level residual network vggnet for fish species classification. *Journal of King Saud University-Computer and Information Sciences* 34 (8), 5286–5295. doi: 10.1016/j.jksuci.2021.05.015

Qin, H., Li, X., Liang, J., Peng, Y., and Zhang, C. (2016). Deepfish: Accurate underwater live fish recognition with a deep architecture. *Neurocomputing* 187, 49–58. doi: 10.1016/j.neucom.2015.10.122

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779-788. doi: 10.1109/cvpr.2016.91

Redmon, J., and Farhadi, A. (2017). "Yolo9000: Better, faster, stronger," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6517-6525. doi: 10.1109/cvpr.2017.690

Redmon, J., and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*. doi: 10.48550/arXiv.1804.02767

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). "Mobilenetv2: Inverted residuals and linear bottlenecks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4510–4520. doi: 10.1109/cvpr.2018.00474

Tan, M., Pang, R., and Le, Q. V. (2020). "Efficientdet: Scalable and efficient object detection," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10778–10787. doi: 10.1109/cvpr42600.2020.01079

Tang, Y., Han, K., Guo, J., Xu, C., Xu, C., and Wang, Y. (2022). Ghostnetv2: Enhance cheap operation with long-range attention. *Adv. Neural Inf. Process. Syst.* 35, 9969–9982. Available at: https://proceedings.neurips.cc/paper_files/paper/2022/file/40b60852a4abdaa696b5a1a78da34635-Paper-Conference.pdf.

Wang, C.-Y., Bochkovskiy, A., and Liao, H.-Y. M. (2023). "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 7464–7475.

Wang, P., Chen, P., Yuan, Y., Liu, D., Huang, Z., Hou, X., et al. (2018). "Understanding convolution for semantic segmentation," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1451-1460. doi: 10.1109/wacv.2018.00163

Wang, W., Dai, J., Chen, Z., Huang, Z., Li, Z., Zhu, X., et al. (2022). Internimage: Exploring large-scale vision foundation models with deformable convolutions. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14408–14419.

Woo, S., Park, J., Lee, J.-Y., and Kweon, I. S. (2018). CBAM: Convolutional Block Attention Module. *Proceedings of the European conference on computer vision (ECCV)*, 3–19. doi: 10.1007/978-3-030-01234-21

Xu, X., Li, W., and Duan, Q. (2021). Transfer learning and se-resnet152 networks-based for small-scale unbalanced fish species identification. *Computers and Electronics in Agriculture* (Elsevier) 180, 105878. doi: 10.1016/j.compag.2020.105878

Yang, L., Zhang, R.-Y., Li, L., and Xie, X. (2021). Simam: A simple, parameter-free attention module for convolutional neural networks. *Int. Conf. Mach. Learning, International. Conf. Mach. Learn* (PMLR), 11863–11874. Available at: https://proceedings.mlr.press/v139/yang21o.html.

Yu, F., and Koltun, V. (2015). Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*.

Yu, F., Koltun, V., and Funkhouser, T. (2017). "Dilated residual networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 636–644. doi: 10.1109/cvpr.2017.75

Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., et al. (2022a). Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv. preprint. arXiv:2203.03605*.

Zhang, Z., Du, X., Jin, L., Wang, S., Wang, L., and Liu, X. (2022b). Large-scale underwater fish recognition via deep adversarial learning. *Knowledge. Inf. Syst.* 64, 353–379. doi: 10.1007/s10115-021-01643-8

Zhou, X., Chen, S., Ren, Y., Zhang, Y., Fu, J., Fan, D., et al. (2022). Atrous pyramid gan segmentation network for fish images with high performance. *Electronics* 11, 911. doi: 10.3390/electronics11060911

Zhu, X., Hu, H., Lin, S., and Dai, J. (2019). "Deformable convnets v2: More deformable, better results," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9300-9308. doi: 10.1109/cvpr.2019.00953