Proceedings of the 3rd International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/2025.26117

SUD-YOLO:A Stable Underwater Target Detection Algorithm Based on Sampling Improved YOLOv11

Yujun Cai

University of Reading Reading, Berkshire, England, United Kingdom
caiyujun20000608@163.com

Abstract: In the field of target detection, underwater target detection (UTD) still faces many challenges. Although YOLO11 shows excellent real-time detection performance, its direct application in UTD is not satisfactory because it has not been designed for complex scenarios such as excessive object deformation and blurred lighting in underwater environments, and is unable to fully extract and utilize the effective information in images, resulting in low detection accuracy. To overcome this drawback, we developed a new detection model SUD-YOLO (Stable Underwater Detection) based on YOLOv11 to improve the detection accuracy and stability for underwater objects. Compared with YOLOv11, SUD-YOLO provides SRFD (Shallow Robust Feature Downing-sampling) and DRFD (Deep Robust Feature Downing-sampling) modules, which alleviate the problem of important information loss during the deep propagation process due to sampling (Upsampling and Downsampling) or multi-layer convolution by input feature scaling fusion. At the same time, EfficientHead is adopted instead of the traditional mixed detection head to ensure that the output features are not mutually dependent. Experimental results on the URPC2020, Luderick and Deepfish datasets prove that SUD-YOLO has higher stability and faster convergence during training, demonstrating excellent UTD performance. This research proposes an efficient and reliable method for UTD tasks, providing technical support for underwater exploration and marine resource investigation, and contributing to the development of underwater intelligent detection.

Keywords: Underwater target detection, YOLOv11, SRFD and DRFD, EfficientHead

1. Introduction

For a long time, object detection has been one of the core issues in the field of computer vision, and it is also a challenging research direction. The difficulty of this research lies mainly in that the images of objects are often disturbed by various noise factors such as lighting, background, and occlusion. At the same time, due to the inevitable jitter of camera and movement of object itself, the final obtained images often become blurry and deformed, which will lead to misleading and low usability. Therefore, the main goal of this task is to, through vision algorithm models, accurately select the object regions and their categories of interested images or video frame sequences as much as possible under the influence of various noise factors.

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2025.26117

Currently, the target detection models for images or videos are mainly divided into two categories: single-stage and two-stage. The single-stage target detection algorithms mainly include the YOLO series models suitable for real-time image processing. Compared with multi-stage models, this type of model has higher efficiency in real-time monitoring tasks because it does not need to pre-select candidate regions. Additionally, SSD model [1] can handle features of different image scales. Although it is also a single-stage detection method like YOLO, this model can use features of different depths for target detection task, thus having higher accuracy in detecting targets of different sizes, especially small objects, compared to YOLO. In recent years, methods based on attention mechanisms such as Vit [2] and Swin-Transformer [3] have gradually emerged. This method, compared with the traditional CNN structure, enables these models to pay more attention to important local or global features of images when processing high-resolution images, thereby reducing computation and improving efficiency. Meanwhile, two-stage models such as Faster-RCNN [4] and Mask-RCNN [5] have also received extensive research due to their high detection performance. These models first generate candidate regions through selective search algorithms or region proposal neural networks and then extract features on the candidate regions for further detection. This method can effectively avoid bootstrap characteristics of single-stage models and significantly reduce the false detection rate of the model.

The single-stage object detection box-based YOLO detection model has received extensive attention and research due to its extremely fast detection speed. For instance, the YOLOV3 model [6], compared to the previous two versions, increased the depth of the convolutional layers while introducing residual modules, Batch Normalization layers [7], and LeakyReLU activation functions, making the model have a high cost-performance ratio in terms of both detection speed and accuracy. At the same time, to address the shortcomings of YOLOV1 [8] and YOLOV2 [9] in small target detection, the FPN [10] structure was introduced, which fused multi-level features to improve the detection performance of different-sized targets.

YOLOV5 improved the efficiency of feature extraction by introducing CSP, SPP, and PAN structures in the backbone and neck networks while ensuring effective reduction of model parameters. In subsequent developments, this series of models also continuously evolved in network structure optimization, such as YOLOX [11] discarding previous prior box mechanism and adopting the Anchor-free approach; YOLOV8, based on the SPP structure of YOLOV5, introduced the SPPF structure, further optimizing the model parameters and improving the inference speed. At the same time, both introduced the decoupled detection head method to separate the detection and classification modules of the model, reducing mutual interference during gradient backpropagation.

In recent years, the YOLOv11 model was proposed based on the comprehensive improvements of the previous YOLO series, such as replacing the C2F module of YOLOv8 in the backbone network with the C3K2 module, which combines the flexibility of the C3K module and the efficiency of the C2F module, reducing model parameters while increasing the adaptability of muti-scale features. Additionally, after the traditional SPPF layer, the C2PSA structure was added, introducing the PSA attention mechanism to enhance the feature extraction ability of the backbone network. In terms of detection head improvement, YOLOv11 retained the depth separable convolution in YOLOv10 [12] to replace the traditional convolution, reducing the number of parameters and improving the detection efficiency. Overall, the improvements of YOLOv11 not only enhance the detection accuracy and stability but also raise detection efficiency.

Although these models have demonstrated excellent application value in the field of object detection and have shown good performance on specific datasets, there are still some challenging problems in the field of object detection that need to be solved. For example, the images captured by

DOI: 10.54254/2755-2721/2025.26117

cameras cannot fully reflect the actual deformation, size and texture differences of the target in the real world, which makes the model less generalizable in real applications. Secondly, in video object detection, there are often situations of motion blur or occlusion, which can make it difficult to locate and identify the detected target, and even in environments where the distinction between the background and the target is too low, the detection performance will significantly decline. Thirdly, if the object to be detected has characteristics of scarcity, then there will be an imbalance problem in the dataset. Similarly, overly dense detection objects can lead the model to learn redundant information, all of which will affect the overall training effect of the model. Therefore, in order to improve the detection performance of the model in complex environments, this research will adopt a target detection model that integrates a feature sampling module and detection augmentation methods to address the complex underwater fish target detection problem. The main work of this study is as follows:

SRFD (Shallow Robust Feature Downing-sampling) and DRFD (Deep Robust Feature Downing-sampling) feature sampling modules were introduced, which alleviated the problem that the shallow backbone network often lost important information during feature extraction, and at the same time ensured that the information flow could be stably propagated in the deep layers of network.

EfficientHead decoupled detection head structure was adopted to solve the interference of feature information and improve the detection accuracy of the model.

Through experiments, SUD-YOLO was comprehensively evaluated on three underwater target detection datasets. The model demonstrated excellent generalization performance.

2. Related work

Over the past decade, numerous studies have been devoted to the task of underwater target detection. In 2015, Choi [13] first applied the GoogLeNet deep learning framework based on convolutional neural networks to the fish detection and classification task of LifeCLEF. In the task, he used the foreground selection search algorithm to select the target candidate regions and then classified the regions through the neural network model. This method enabled the model to achieve a fish count score and detection accuracy of over 0.9 and 0.8, respectively, in the video.

In 2016, Zhang et al. [14] adopted an unsupervised learning approach to automatically label fish target samples in their research. This method solved the difficulty of manual annotation of underwater fish datasets by integrating optical flow and selective search algorithms. After obtaining a large number of candidate regions through this method, they also employed an improved NMS method to reduce the model's misclassification rate, ultimately increasing the average precision (AP) by approximately 20% compared to the un-fused model.

Li et al. [15] made lightweight improvements to Faster-RCNN, introducing mainstream methods such as C.Relu architecture, residual connections, and batch normalization to accelerate the network while enhancing performance. They achieved a MAP of 90% on the ImageCLEF fish dataset for detection and multi-classification tasks and significantly improved the inference speed compared to the original model.

In recent years, due to the inability of two-stage detection models to meet the requirements of most real-time underwater detection tasks, single-stage methods with faster detection speeds, such as the fish detection model based on YOLO, have been widely applied. For the improvement of Yolov3, Abdullahz et al. [16] optimized the upsampling stride and added a spatial pyramid structure on its basis, achieving an average accuracy of 75% on the Deepfish and Ozfish datasets. In subsequent research, Long et al. [17] introduced a triple attention module of coordinate attention, channel attention, and spatial attention into the backbone network of the YOLOv5s architecture,

DOI: 10.54254/2755-2721/2025.26117

demonstrating comparable MAP and superior inference speed to the larger YOLOv5L model on the zebrafish dataset with a high density of target quantities.

In the past two years, improvements based on the YOLO series have become increasingly common. Xu et al. [18] proposed a residual enhancement module, a shared parameter detection head, and a lightweight edge enhancement module for underwater object detection tasks to improve the YOLOv8 model. The optimized model achieved MAP of 88.1% and 86.2% on the RUOD and URPC2020 datasets, respectively, which significantly reducing the parameters of model while maintaining improved detection accuracy. Guo et al. [19] also based their research on the YOLOv8 model to address the issues of small underwater targets and object deformation, mainly introducing a transformable convolution structure to optimize the backbone network. This module enhanced the model's recognition of spatial transformations and achieved the best results under high IOU conditions on both the URPC2020 and COCO2017 datasets.

Based on the latest proposed YOLOv11 model structure, Liao et al. [20] made optimizations by introducing the multi-scale expansion attention mechanism (MSDA) to improve the C2PSA module in the original structure. They also introduced a learnable spatial feature fusion module for the detection head, thereby enhancing the model's multi-scale information fusion capability. At the same time, to address the problem of excessive difficult samples caused by class imbalance in the dataset, a sliding weight loss function method was adopted to increase the learning weight of difficult samples. This method demonstrated excellent generalization performance on DUO and RUOD. Luo et al. [21] recognized the importance of global and local context information for detection performance in their experiments. They improved the model's detection efficiency by using various methods, such as replacing the traditional convolutional upsampling method with the ADown module, proposing C2PSA to replace the traditional C2PSF structure in feature fusion, introducing depthwise separable convolution, point-wise affine transformation, and gating mechanisms, which enhanced the ability of model to fuse global and local information and achieved a 1% to 2% improvement in mAP compared to the baseline model on datasets UTDAC2020, DUO, and RUOD.

3. Methodology

3.1. Data preprocessing

In terms of the preprocessing of the dataset, this study first resized all the input images to 640*640 pixels. Then, the dataset was divided into training set and validation set. To verify the generalization performance of the model, this experiment was conducted on three different classic underwater datasets. Among them, the dataset Deepfish was divided into 14,148 and 2,645 images; URPC2020 was divided into 5,351 and 1,534 images; and Luderick was divided into 2,672 and 824 images. Before substituting these datasets into the model for training, these images need to be scaled to 640*640 pixels and undergo certain probabilities of translation, rotation, cropping, mixing, and mosaic enhancement. To increase the stability of model training, it is necessary to pre-cluster the detection boxes to obtain the prior boxes of different scale targets as the benchmark for selecting the target size during model training.

3.2. Improved network architecture

The structure of YOlOv11 [22] is the starting point for this improvement method. Compared with the previous classic YOLOv5 and YOLOv8 structures, this structure introduces the C3k2 module. Different with traditional Bottleneck, C3, and C2f modules, this module adopts dynamically

DOI: 10.54254/2755-2721/2025.26117

adjustable convolution kernels, enabling the model to obtain multi-scale feature information in complex detection tasks. For detection head, depth wise separable convolution structure replaces the ordinary convolution, and this structure uses different independent convolution layers for each passband, significantly reducing interference between channels and enabling better acquisition of important information of different channels.

This study improved the basic network model framework of YOlOv11 to adapt to underwater target detection tasks with variable light conditions, large variations in target scales, and turbid water quality environments. This network structure is mainly composed of Backbone, Neck, and Head.

Backbone part of this study introduces SRFD module to replace the first two convolution layers of the original structure, and all other convolution modules in this part are replaced by the DRFD module [23].

The Neck part is used to connect the Backbone and Head, and mainly used for fusion of multi-scale features. To extract deep features without losing important feature information, DRFD is also used to replace the feature upsampling modules of medium-sized and large-sized targets.

Finally, the Head part is mainly used for the detection and classification of targets based on the features extracted from the Backbone and Neck networks. The improved detection structure is shown in Figure 1.

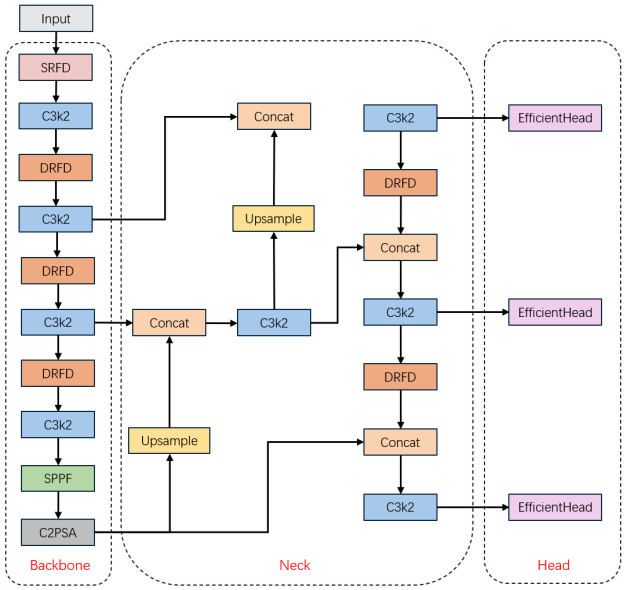

Figure 1. Overall structure of SUD-YOLO

3.3. High-low frequency feature sampling module

The adoption of the high and low frequency feature sampling module (SRFD and DRFD) in this detection structure is, on one hand, to replace the traditional upsampling module while ensuring that the original feature information will not be lost and extracting additional content further. Due to the blurry nature of underwater images, after multiple layers of feature extraction, much noise information may be generated during training, which often leads to instability in the training process of model. This instability will be amplified in inappropriate upsampling and downsampling, resulting in the difficulty of model convergence. For the high-frequency features of underwater images, these features retain the texture details of the objects and often have certain discrimination against different shapes of underwater objects. This information is, on the one hand, dependent on

DOI: 10.54254/2755-2721/2025.26117

the quality of the low-level feature extraction layer in the detection network, and on the other hand, depends on the feature discrimination ability of the subsequent deep structure. At the same time, for the high-frequency features of the image, especially for blurry underwater images, most of these features are generated by noise. These noises may be determined by environmental factors such as light intensity and water quality, and for the targets to be detected, most of these are interference information. To remove these types of noise, for deep layer modules in the network structure, better feature filtering ability is required. To increase stability of sampling, batch normalization is added in SRFD and DRFD, providing a guarantee for the stable forward and backward propagation of the feature flow in this module. The optimized module structure is shown in Figures 2 and 3.

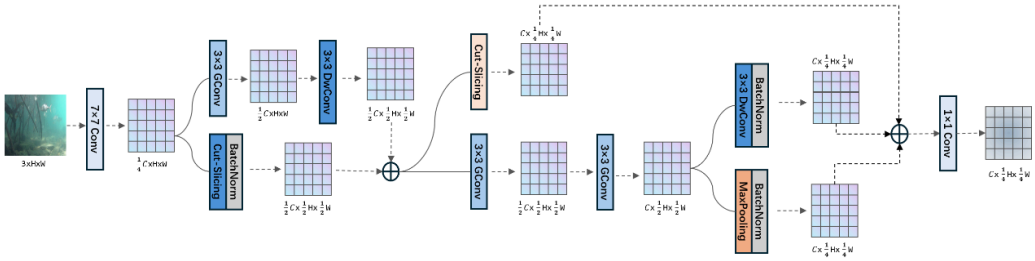

Figure 2. The structure of SRFD. ⊕ represents the concat operation. The elements below each tensor represent the size of channels, height and width respectively

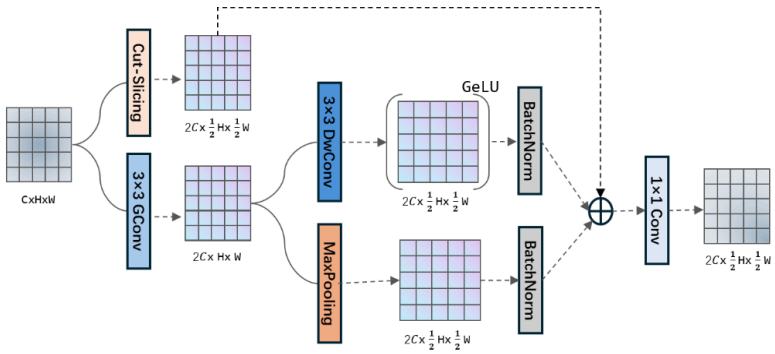

Figure 3. The structure of DRFD. GeLu represents the Gaussian Error Linear Units activation function

3.4. Efficient Detection Head

The hybrid detection head has certain advantages in terms of parameter quantity and efficiency for simple tasks due to its one-step nature. However, for such complex tasks, the hybrid detection structure is prone to cause interference among feature information, thereby directly affecting the final detection performance and accuracy. In other words, for the final detection tasks, which mainly involve the positioning of target boxes and the classification of target categories, the features that are focused on for these two tasks may be different. Therefore, the traditional strategy of sharing a single feature in complex tasks cannot fully exert the advantages of the model itself [24]. For this problem, in this experiment, the decoupled Efficient Detection Head proposed in the YOLOv6 structure [25] was adopted to replace the original hybrid detection head structure (Figure 4).

DOI: 10.54254/2755-2721/2025.26117

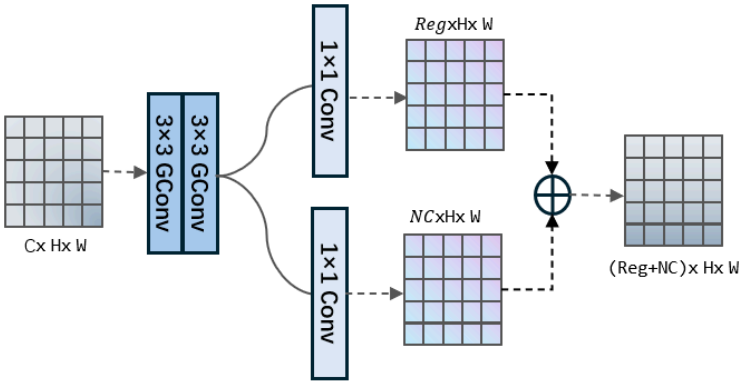

Figure 4. The structure of Efficient Detection Head. Reg represents the number of regression boxes, while NC stands for the number of classification categories

4. Experiments

In this section, we will evaluate the performance of the high and low frequency feature sampling module and efficient head combined with the YOLOv11 basic structure. On one hand, we will verify it through various detection metrics of the Deepfish, Luderick and URPC2020 datasets. On the other hand, we will examine the coupling between the modules in the form of ablation experiments.

4.1. Datasets

Deepfish dataset [26] is a publicly available large-scale benchmark for marine fish image segmentation and detection. This dataset contains over 20 different marine fish habitats and more than 40,000 high-definition underwater images from the subtropical region. In this experiment, 4,405 images were selected as the training and evaluation benchmark for this detection task. The target category was limited to fish, and this sub-dataset also included all different underwater environments in the original dataset. To facilitate subsequent verification and comparison of model performance, it was further divided into a training set of 3,596 images and a validation set of 809 images.

Luderick dataset [27] was proposed to enable training of high-quality automatic annotation models. The images in this dataset come from two river basins in southeastern Australia, the Tweed estuary basin, and the Tallebudgera Creek basin with extensive seagrass coverage. The original data collection was captured by underwater mobile cameras, recording videos of Luderick fish and Australian bream in dense seagrass backgrounds. This study selected 3,496 images from it as the evaluation basis and conducted a unified single-class target detection task for fish.

URPC2020 dataset [28] originated from the Underwater Robot Picking Contest in 2020. The original dataset was collected by underwater remote robots. The purpose of this competition was to combine underwater target detection with robots, providing more data sources that are more in line with the real underwater environment for the target detection field. In this experiment, 5,351 images were selected as the training set and 1,534 as the validation set. It included four different target categories: echinus, holothurian, scallop, and starfish.

DOI: 10.54254/2755-2721/2025.26117

4.2. Experimental setup

4.2.1. Evaluation metrics

This experiment will comprehensively evaluate the performance of the model on different datasets through Precision, Recall, F1-score, AP and MAP. Firstly, the accuracy rate serves as the evaluation metric for the overall performance of the model. This metric represents the proportion of truly correct instances among all the results that are detected as positive examples, reflecting precision of model. It is defined as:

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

where TP represents the number of positive samples in the detection results, and FP is the number of samples that were detected as positive but actually were negative. Secondly, the recall rate indicates the proportion of all Ground Truth samples that were detected by the model, reflecting the model's completeness in detection. The formula is defined as:

$$Recall = \frac{TP}{TP+FN} \tag{2}$$

where FN refers to the number of negative samples that are detected as negative cases.

Since Precision and Recall can change completely opposite ways due to the influence of FP or FN, to better balance the relationship between them, the introduction of the F1-score metric alleviates this problem. It is defined as follows:

$$F1-score = \frac{2*(Precision*Recall)}{Precision+Recall} \tag{3}$$

In the above formula, the definitions of Precision and Recall have been provided by (1) and (2).

Fourthly, the detection accuracy of each category is measured using the AP (Average Precision) metric. In this experiment, the sample selection criteria for each category is IOU of 0.5. This metric can better reflect the performance of the model in detecting specific categories, which is expressed as:

$$AP = \int P(r)dr \tag{4}$$

P(r) refers to the function of Precision with respect to Recall, where r represents the recall rate. In general, this means integrating the PR curve with respect to the recall rate r.

Finally, to evaluate the comprehensive detection performance of the model for all categories, this experiment adopted MAP as the measurement standard. The samples used to calculate this indicator were selected based on the criterion of having an IOU of 0.5 or an IOU between 0.5 and 0.95. The calculation details are as follows:

DOI: 10.54254/2755-2721/2025.26117

$$MAP = \frac{1}{NC} \sum_{i=1}^{NC} AP_i \tag{5}$$

where NC represents the total number of all sample categories in this dataset, and AP_i represents the average accuracy rate calculated for each category.

4.2.2. Implementation detail

The GPU model used in this experiment is Nvidia GeForce 2080Ti, which is deployed on the Windows 10 operating system. The environment used for this model is Python 3.10 and CUDA 12.1 version.

The model structure adopts the YOLOv11s model in the YOLOv11 version, and the configuration is that image input size is 640 and the batch-size is 16. The loss function is chosen as CIOU, the optimizer is selected as SGD, the learning rate and weight decay rate are set to 0.01 and 0.0005 respectively, and the training cycle is set to 100 epochs. Mosaic enhancement is turned off in the last 10 epochs. Additionally, the YOLOv5 and YOLOv3 versions used in the comparative experiments also adopt similar settings.

4.3. Experimental results

Firstly, the detection results of the SUD-YOLOv11 model on the URPC2020, Luderick and Deepfish datasets are shown in Figure 5. Where, the three rows represent the detection results of the model on URPC2020, Luderick and Deepfish respectively. Each row consists of the Ground Truth and prediction labels after model inference for comparison. To comprehensively evaluate the performance of the model on different underwater datasets, Table I. presents the results using three evaluation metrics (Precision, Recall and MAP). Besides the performance differences between different datasets, the comparative experiments can directly reflect the advantages and disadvantages of different models. Therefore, YOLOv3, YOLOv5 and YOLOv11 were also compared.

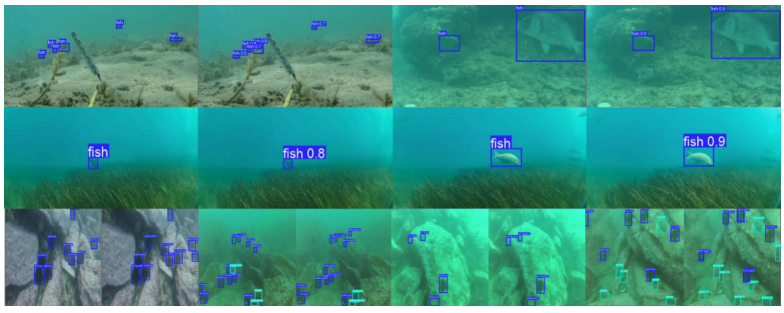

Figure 5. The comparison results of SUD-YOLO on different datasets. The first and third columns are ground truth, the second and fourth columns are the detection results

DOI: 10.54254/2755-2721/2025.26117

Table 1. Performance on datasets

Dataset	Method	F1-score	Recall	MAP_0.5	MAP_0.5:0.95
URPC2020	YOLOv3	0.806	0.777	0.841	0.464
	YOLOv5	0.809	0.782	0.847	0.476
	YOLOv11	0.786	0.748	0.831	0.488
	SUD-YOLO（Ours）	0.810	0.784	0.852	0.504
Luderick	YOLOv3	0.842	0.814	0.883	0.505
	YOLOv5	0.842	0.814	0.883	0.505
	YOLOv11	0.875	0.863	0.911	0.586
	SUD-YOLO（Ours）	0.901	0.891	0.929	0.627
Deepfish	YOLOv3	0.959	0.950	0.981	0.662
	YOLOv5	0.954	0.948	0.981	0.654
	YOLOv11	0.937	0.920	0.969	0.653
	SUD-YOLO（Ours）	0.948	0.944	0.976	0.672

Table 2. Ablation experiment based on YOLOv11

Dataset	Modules	F1-score	Recall	MAP_0.5	MAP_0.5:0.95
URPC2020	YOLO11	0.786	0.748	0.831	0.488
	+SRFD	0.789	0.756	0.841	0.493
	+EfficientHead	0.795	0.769	0.843	0.498
	+all	0.806	0.779	0.852	0.504
Luderick	YOLO11	0.875	0.863	0.911	0.586
	+SRFD	0.880	0.876	0.923	0.608
	+EfficientHead	0.881	0.866	0.921	0.621
	+all	0.893	0.877	0.929	0.627
Deepfish	YOLO11	0.937	0.920	0.969	0.653
	+SRFD	0.940	0.927	0.973	0.659
	+EfficientHead	0.942	0.938	0.975	0.661
	+all	0.941	0.94	0.976	0.672

Overall, from the detection results of the optimized model on the three datasets, for the MAP metric with an intersection-over-union (IOU) of 0.5, it achieved better performance than other models on both URPC2020 and Luderick datasets. Moreover, at higher IOU levels (MAP50:95), it could perform optimally on all three datasets. The F1-score takes into account precision and recall rate of the model, and demonstrates stability of the model in detecting positive and negative samples on URPC and Luderick. From the URPC2020 dataset, compared to the baseline model YOLOv11, there are improvements in all metrics, with the Recall metric showing the most significant improvement, approximately 3% increase. Additionally, compared to YOLOv3 and YOLOv5 models, the MAP metrics have improved by 1%-2% within different IOU ranges of 0.5 and 0.95. For Luderick dataset, this model outperformed other models in all metrics, with a 4% improvement in MAP50:95 compared to the baseline model, and a 12% improvement compared to other models.

DOI: 10.54254/2755-2721/2025.26117

Although the Deepfish dataset did not outperform YOLOv3 in Precision, Recall, and MAP50, it still showed a significant difference in the MAP50:95 metric, demonstrating better recognition performance for positive samples.

From the above results, this model shows significant advantages over the baseline model YOLOv11, as well as models v3 and v5. To further illustrate introduction meaning of the SRFD, DRFD, and Efficient Head modules, this study used ablation experiments to verify this point. This study separately tested high and low frequency sampling modules (SRFD and DRFD) and the Efficient Head module. In Table 2, "+SRFD" represents adding the SRFD and DRFD modules to the Yolov11 structure, "+EfficientHead" represents only adding the Efficient Head detection head structure, and "+all" represents the detection results after combining both modules. For all datasets, the introduction of the high and low frequency sampling modules not only improved the F1-score but also improved the Recall and MAP metrics. This module showed the most significant improvement for the Luderick dataset, averaging an improvement of over 1% in all detection metrics. The introduction of the Efficient Head module improved Recall by approximately 2% for both URPC2020 and Deepfish datasets. The replacement of the detection head, compared to the high and low frequency sampling modules, had a more significant improvement in detection accuracy. Finally, the integration of the two modules overall achieved a good fit with the model structure, reaching the optimal or near-optimal level on all datasets.

From the training process, the introduction of these modules had a significant impact on the stability and convergence of training. Figure 5 shows the update trend graphs of the four important indicators of each dataset over 100 training epochs. The improved model, compared to the baseline model, had smaller fluctuations in precision in the first 50 epochs, a more stable improvement trend, and a smoother update process, which was reflected in all three datasets. From the MAP metric, the training stability of the Deepfish dataset has improved the most significantly. Meanwhile, for the other two datasets, its contribution mainly lies in faster convergence of the model in the first 20 rounds. Similarly, the stable improvement in the recall rate metric is also reflected in the results.

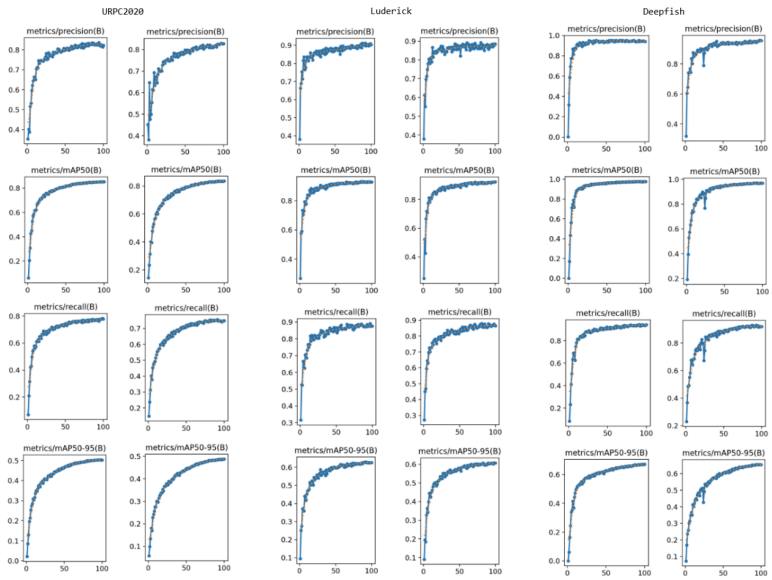

Figure 6. The comparison results between SUD-YOLO and YOLOv11 on different datasets during different traning periods. For each dataset, the first column of figures shows metrics of SUD-YOLO, and the second column is for YOLOv11

DOI: 10.54254/2755-2721/2025.26117

5. Conclusion

This research is based on the structure of YOLOv11 and introduces a high-low frequency sampling module and an efficient detection head structure, enabling the model to obtain relatively robust results on three different underwater datasets.

Firstly, regarding the improvement of the model, a high-frequency sampling module is adopted in the Backbone to replace the downsampling layers of two convolutional layers. On the one hand, it ensures the fusion performance of multi-scale features. On the other hand, it prevents a large amount of important texture information from being lost during the shallow layer feature transmission process. This information is very important in the complex underwater target detection task and more stable feature extraction and more effective information fusion are the advantages of the high-frequency sampling module. The Cut-Slice method ensures that the features can be downsampled without losing too much useful details, and the DwConv can well perform feature scaling, while the GConv can improve the computational efficiency when fusing different scale information.

For the deep convolutional layers in the backbone network, a low-frequency sampling module is used for replacement. This module, in addition to the Cut-Slice, GConv and DwConv in the high-frequency sampling module, also adds the Gelu function to prevent the gradient disappearance situation of deep features during the gradient backpropagation process. This is helpful for the stability and convergence of the model.

The improvement of the detection head structure brings an enhancement in the ablation experiment section. Due to its decoupled detection structure, compared with the traditional mixed detection head of other models, this separated detection structure can reduce the interference between detection box results and classification results. This characteristic makes this module show the most significant improvement on the URPC2020 dataset, which has a large number of objects to be detected and significant deformation of the image itself, as well as obvious differences in light and shade between images. Additionally, the biggest challenge for the stability of model training lies in the imbalance of categories in this dataset, which causes noise interference to the parameter updates of scarce categories for the ones with a larger number. This also demonstrates the importance of introducing EfficientHead to solve such problems.

Finally, the final model based on the improvements of these two modules outperforms the baseline model overall, and the compared models also show good generalization on the three datasets. It demonstrates the good adaptability of this model to the variable underwater datasets. The Luderick dataset is clearer in the environmental background compared to URPC2020 and Deepfish. To some extent, this also proves that this model does not only perform well in blurry or clear datasets. Similarly, this model can also be applied to detect environments with complex and variable environmental conditions, such as scenes with fog or dim lighting. For datasets with diverse data distributions and significant changes in the image domain, this method or further improvements can be attempted in future research.

References

[1] Wei Liu et al. “SSD: Single Shot MultiBox Detector”. In: Computer Vision ECCV 2016. Springer International Publishing, 2016, pp. 21–37. isbn: 9783319464480. doi: 10.1007/978-3-319-46448-0_2. url: http: //dx.doi.org/10.1007/978-3-319-46448-0_2.

[2] Alexey Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. 2021. arXiv: 2010.11929 [cs.CV]. url: https: //arxiv.org/abs/2010.11929.

[3] Ze Liu et al. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. 2021. arXiv: 2103.14030 [cs.CV]. url: https: //arxiv.org/abs/2103.14030.

DOI: 10.54254/2755-2721/2025.26117

[4] Xinlei Chen and Abhinav Gupta. An Implementation of Faster RCNN with Study for Region Sampling. 2017. arXiv: 1702.02138 [cs.CV]. url: https: //arxiv.org/abs/1702.02138.
[5] Kaiming He et al. Mask R-CNN. 2018. arXiv: 1703.06870 [cs.CV]. url: https: //arxiv.org/abs/1703.06870.
[6] Joseph Redmon and Ali Farhadi. YOLOv3: An Incremental Improvement. 2018. arXiv: 1804.02767 [cs.CV]. url: https: //arxiv.org/abs/1804.02767.
[7] Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. 2015. arXiv: 1502.03167 [cs.LG]. url: https: //arxiv.org/abs/1502.03167.
[8] Joseph Redmon et al. You Only Look Once: Unified, Real-Time Object Detection. 2016. arXiv: 1506.02640 [cs.CV]. url: https: //arxiv.org/abs/1506.02640.
[9] Joseph Redmon and Ali Farhadi. YOLO9000: Better, Faster, Stronger. 2016. arXiv: 1612.08242 [cs.CV]. url: https: //arxiv.org/abs/1612.08242.
[10] Tsung-Yi Lin et al. Feature Pyramid Networks for Object Detection. 2017. arXiv: 1612.03144 [cs.CV]. url: https: //arxiv.org/abs/1612.03144.
[11] Zheng Ge et al. YOLOX: Exceeding YOLO Series in 2021. 2021. arXiv: 2107.08430 [cs.CV]. url: https: //arxiv.org/abs/2107.08430.
[12] Ao Wang et al. YOLOE: Real-Time Seeing Anything. 2025. arXiv: 2503. 07465 [cs.CV]. url: https: //arxiv.org/abs/2503.07465.
[13] Sungbin Choi. “Fish Identification in Underwater Video with Deep Convolutional Neural Network: SNUMedinfo at LifeCLEF Fish task 2015”. In: Working Notes of CLEF 2015 Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015. Ed. by Linda Cappellato et al. Vol. 1391. CEUR Workshop Proceedings. CEUR-WS.org, 2015. url: https: //ceur-ws.org/Vol-1391/110-CR.pdf.
[14] David Zhang et al. “Unsupervised underwater fish detection fusing flow and objectiveness”. In: 2016 IEEE Winter Applications of Computer Vision Workshops (WACVW). Mar.2016, pp. 1–7. doi: 10.1109/WACVW.2016.7470121.
[15] Xiu Li, Youhua Tang, and Tingwei Gao. “Deep but lightweight neural networks for fish detection”. In: OCEANS 2017 - Aberdeen. 2017, pp. 1–5. doi: 10.1109/OCEANSE.2017.8084961.
[16] Abdullah Al Muksit et al. “YOLO-Fish: A robust fish detection model to detect fish in realistic underwater environment”. In: Ecological Informatics 72 (2022), p. 101847. issn: 1574-9541. doi: https: //doi.org/10.1016/j.ecoinf.2022.101847. url: https: //www.sciencedirect.com/science/article/pii/S1574954122002977.
[17] Wei Long et al. “Triple Attention Mechanism with YOLOv5s for Fish Detection”. English. In: Fishes 9.5 (2024), p. 151.
[18] Chaolong Xu and Zhibin Xie. “A lightweight underwater object detection with enhanced detail and edge-aware feature fusion”. English. In: Digital signal processing 167 (2025), p. 105456.
[19] Lijia Guo et al. “Underwater object detection algorithm integrating image enhancement and deformable convolution Underwater object detection algorithm integrating image enhancement and deformable convolution”. English. In: Ecological informatics 89 (2025), p. 103185.
[20] Dongcheng Liao et al. “Research on Underwater Target Detection Technology Based on SMV-YOLOv11n”. English. In: IEEE access 13 (2025), pp. 119820–119830.10
[21] Shengfu Luo et al. “YOLO-DAFS: A Composite-Enhanced Underwater Object Detection Algorithm”. In: Journal of Marine Science and Engineering 13.5 (2025). issn: 2077-1312. doi: 10.3390/jmse13050947. url: https: //www.mdpi.com/2077-1312/13/5/947.
[22] Rahima Khanam and Muhammad Hussain. “YOLOv11: An Overview of the Key Architectural Enhancements”. English. In: (2024).
[23] Wei Lu et al. “A Robust Feature Downsampling Module for Remote Sensing Visual Tasks”. English. In: IEEE transactions on geoscience and remote sensing 61 (2023), pp. 1–1.
[24] Guanglu Song, Yu Liu, and Xiaogang Wang. “Revisiting the Sibling Head in Object Detector”. English. In: IEEE, 2020, pp. 11560–11569.
[25] Chuyi Li et al. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. 2022. arXiv: 2209.02976 [cs.CV]. url: https: //arxiv.org/abs/2209.02976.
[26] Alzayat Saleh et al. “A realistic fish-habitat dataset to evaluate algorithms for underwater visual analysis”. English. In: Scientific reports 10.1 (2020), p. 14671.
[27] Ellen M. Ditria et al. “Annotated Video Footage for Automated Identification and Counting of Fish in Unconstrained Seagrass Habitats”. English. In: Frontiers in Marine Science8 (2021).
[28] Fenglei Han et al. “Marine Organism Detection and Classification from Underwater Vision Based on the Deep CNN Method”. In: Mathematical Problems in Engineering 2020.1 (2020), p. 3937580. doi: https:

DOI: 10.54254/2755-2721/2025.26117

//doi.org/10.1155/2020/3937580. eprint: https: //onlinelibrary.wiley.com/doi/pdf/10.1155/2020/3937580. url: https: //onlinelibrary.wiley.com/doi/abs/10.1155/2020/3937580.