International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 10 | Oct - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Enhanced RT - DETR Automatic Annotation for Underwater Images

Kavitha[1], Kalaivanan[2], Varshini[3], Sakthivel Murugan S[4]
[1,2,3] Dept. of Electronics and Communication Engineering,
Nellai Nadar College of Engineering and Technology, Chennai, India
[4]Dept. of Electronics and Communication Engineering,
National Institute of Technical Teachers Training and Research, Chennai, India

Abstract

Manual annotation of fish in underwater video is time-consuming and becomes difficult when frames contain small fish, low illumination, complex backgrounds, and coral structures. The difficulty is further increased by underwater-specific visual conditions such as color distortion, reduced visibility, background clutter, occlusion, and variations in the appearance of the same object across different underwater environments. while examining the limitations of detector generalization across different underwater domains. The study follows a progressive experimental methodology. Initially, conventional deep-learning object detectors were investigated to establish a baseline for underwater fish detection. YOLO11n, Faster R-CNN, RetinaNet, and SSD300 were evaluated under the experimental dataset conditions, providing a comparison of detection performance and computational characteristics. Based on the baseline investigation, the study was extended toward an end-to-end transformer-based detection framework using RT-DETR with a PResNet-18 backbone and a single fish class. The strongest examined RT-DETR checkpoint was obtained at epoch 50, achieving a COCO AP@[0.50:0.95] of 0.28847 and an AP50 of 0.60264. The trained detector was subsequently employed as an automatic annotation system for underwater video frames, where confidence-based filtering and duplicate prediction handling were applied to produce fish bounding boxes. An initial automatic annotation experiment involving 61 frames generated 466 predicted fish bounding boxes, demonstrating the practical feasibility of using the trained detector as an annotation-assistance tool. A longer real underwater video containing 2,668 frames was also processed, resulting in 2,247 frames with detected fish and 4,984 predicted fish bounding boxes. These detection counts demonstrate the operational use of the proposed annotation workflow, although they are not considered direct measures of detection accuracy because complete ground-truth annotations were not available for the full video sequences. To investigate real-world generalization, the trained detector was further evaluated on a separate underwater video with different visual characteristics. The experiment revealed significant domain-shift effects, including missed fish, failure to detect small fish, and false detections on coral and other background structures. These observations indicate that changing the confidence threshold alone is insufficient to address the problem because reducing the threshold can increase the number of background false positives. In response to this limitation, a target-domain refinement dataset was developed through manual annotation using Label Studio. Twenty-five target-domain frames were successfully annotated with 253 fish instances using a single fish class while explicitly excluding coral and background structures. The verified dataset was further divided into a 20-image training subset containing 91 fish annotations and a five-image unseen validation subset containing 162 fish annotations. The overall findings demonstrate that deep-learning-based automatic fish annotation is feasible for underwater video processing, but reliable cross-domain deployment remains strongly dependent on the similarity between training and target environments. The study therefore highlights domain shift as a major practical challenge in underwater object detection and establishes a target-domain annotation and refinement workflow for addressing this limitation. The present work focuses on the integration and evaluation of established detection methods rather than proposing a new detector architecture.

Keywords

Underwater object detection; automatic fish annotation; fish detection; YOLO11n; Faster R-CNN; SSD300; RT-DETR; UCOE-DETR; CGAA; WCAP; domain shift; target-domain refinement; deep learning.

I. INTRODUCTION

Underwater video analysis plays an important role in marine ecosystem monitoring, fish observation, biodiversity assessment, and automated aquatic surveillance. A major practical obstacle is the effort required to locate and annotate fish across large numbers of frames: even a short recording contains hundreds or thousands of frames, so frame-by-frame bounding-box annotation is laborious and difficult to scale. Automated detection can assist this process by proposing fish boxes that an annotator can review.

Underwater imagery is difficult for detectors. Illumination and colour change with depth and camera orientation, suspended particles reduce contrast, and seabed structures can hide fish or resemble them, and fish may be partly occluded [7]–[9]. Small fish are easily missed, the same kind of object can look different in different underwater environments, and coral can produce detections that look like the target object. A detector intended for annotation assistance must therefore be examined not only with standard detection metrics but also for its behaviour when the visual domain changes.

The present work follows a practical pipeline. Several established detectors were first evaluated as baselines. A PResNet-18-based RT-DETR detector was then trained and used to annotate underwater video automatically. A second underwater video was subsequently used to examine cross-domain generalization, and the failures observed there motivated the creation of a small manually annotated target-domain dataset. The project initially considered both fish and coral; the final scope is single-class fish detection, and coral is treated as background.

The objective is to investigate an automatic fish-annotation workflow under underwater domain variation. The work does not propose a new detection architecture. The main contributions are: (1) an automatic fish-annotation workflow for underwater video; (2) a comparative investigation of several object detectors; (3) implementation and evaluation of a PResNet-18-based RT-DETR detector; (4) analysis of cross-domain failures involving coral false positives and missed fish;

and (5) a manually annotated target-domain dataset with a train/validation split prepared for future refinement.

II. LITERATURE SURVEY

Deep-learning object detectors are commonly grouped into two-stage and one-stage families. Faster R-CNN [2] uses region proposals and offers strong localization, but its per-frame cost can be high for video processing. One-stage detectors such as SSD [3] and RetinaNet [4] trade some of that localization quality for speed, and YOLO-family detectors [1] are attractive for video because of their inference speed.

Transformer-based detection began with DETR [5], which formulates detection as direct set prediction without hand-designed components such as non-maximum suppression. RT-DETR [6] adapts this idea to real-time detection by combining a convolutional backbone with an efficient hybrid encoder and a transformer decoder. A single-class RT-DETR configuration is relevant to underwater fish annotation because it provides an alternative to conventional anchor-based or NMS-dependent detection pipelines.

UCOE-DETR [11] is an underwater-oriented extension of the RT-DETR framework that incorporates Cascaded Group Additive Attention (CGAA) and a Wavelet Composite Attention Pyramid (WCAP), aimed at improving robustness to underwater visual degradation and small-object detection. In this work RT-DETR serves as the implementation framework, and UCOE-DETR is treated as the research direction toward which the study extends.

Deep learning has also been applied to underwater fish analysis, including fish species classification in underwater video with data augmentation [7], fish species recognition [8], and fish detection and classification [9]. Related work on coral classification [10] illustrates that coral appearance is itself a complex visual category, which is relevant here because coral is the main background structure that can be confused with fish. These studies motivate evaluation beyond a single data source, since lighting, colour distribution, seabed appearance, camera geometry, and fish scale can all differ between videos.

Based on this background, the present work takes a practical adaptation-and-evaluation approach: established detectors are compared, an RT-DETR detector is used for automatic annotation, and cross-domain behaviour is examined explicitly. It is positioned as an application and analysis of existing detectors rather than as a new architecture.

III. PROPOSED SYSTEM

The proposed workflow consists of video frame extraction, annotation and dataset preparation, baseline detector evaluation, RT-DETR training, automatic annotation, cross-domain testing, and target-domain refinement. The overall progression is shown in Fig. 1.

Frames are first sampled from underwater videos and annotated. The early dataset contained fish and coral labels; the scope was later simplified to a single fish class so that coral and other structures are treated as background. This reflects the objective of reliable fish localization rather than coral recognition.

The trained detector then acts as an annotation assistant. For each frame, predictions are filtered by a confidence threshold, overlapping boxes are reduced, and annotation files and visualized images are saved. These boxes are model predictions and are not treated as ground truth.

To examine generalization, the detector is also applied to underwater videos from other visual domains. Failure cases are inspected, and a subset of target-domain frames is manually annotated without using incorrect model predictions as labels. These labels form the basis for future domain-specific refinement.

IV. METHODOLOGY

A. Video Sources and Datasets

Three underwater videos were used (Table I). The first video was the source of the initial manually annotated frames. A second video (911 frames, 1280×720) served as the target domain for generalization testing and manual refinement annotation. A third 4K video was used for a large-scale automatic annotation run. Frame extraction from the first video used a 0.5 s interval, giving 301 extracted frames, from which a smaller annotated subset was created. The initial labels were fish (class 0) and coral (class 1) before the task was reduced to fish only.



Fig. 1. Overall workflow for automatic fish annotation and target-domain refinement. The dashed box indicates a stage that has not yet been performed.

TABLE I. UNDERWATER VIDEOS USED

Video	Resolution	Frames	FPS	Duration (s)
Initial video	478×850	903	30.06	30.04
Second video	1280×720	911	30	30.37
Third video (4K)	3840×2160	2,668	29.97	89.02

The RT-DETR training data were a combined fish dataset of 105 images with 888 fish annotations (Table II). The annotations for this dataset were produced with a combination of manual and automatically generated boxes.

TABLE II. COMBINED RT-DETR FISH DATASET

Split	Images	Fish boxes	Class
Training	84	728	fish
Validation	10	81	fish
Test	11	79	fish
Total	105	888	fish

B. Baseline Object Detectors

YOLO11n (Ultralytics 8.4.152, PyTorch 2.14.0+cpu, Python 3.13.15) was used as the initial lightweight baseline. Faster R-CNN, RetinaNet, and SSD300 were then evaluated for a broader comparison. All experiments ran on CPU. The baselines were evaluated at different times and with different metric sets, so their results are reported with their own evaluation sources and are not ranked against one another.

C. PResNet-18-Based RT-DETR

The main detector is RT-DETR with a PResNet-18 backbone, a HybridEncoder, and the RT-DETR detection head, configured for one class (fish). It is built on the official RT-DETR PyTorch implementation (/RT-DETR, rtdetr_pytorch) and run in a Python 3.11.4 environment on CPU. The trained model is referred to as a PResNet-18-based RT-DETR detector. The study further investigates UCOE-DETR, an underwater-oriented extension of the RT-DETR framework [11], with particular interest in Cascaded Group Additive Attention (CGAA) and Wavelet Composite Attention Pyramid (WCAP) for improving robustness to underwater visual degradation and small-object detection. The CGAA and WCAP components are not part of the configuration trained and evaluated here, so all reported results refer to the PResNet-18-based RT-DETR and not to UCOE-DETR. The checkpoint examined most closely is checkpoint0050 (epoch 50, about 307 MB).

D. Automatic Annotation

An annotation script applies the trained checkpoint to each input frame, discards predictions below a confidence threshold of 0.30, removes duplicate overlapping boxes, and saves annotation files together with visualized images. In a single-image test on a 478×850 frame, the model produced 300 raw detections; 6 remained after confidence filtering and 3 after duplicate removal. Because complete ground truth is not available for the annotated videos, the number of boxes is reported only as an operational statistic and not as an accuracy measure.

E. Cross-Domain Testing

The epoch-50 detector was applied to the 911-frame second video using the same 0.30 threshold. Diagnostic frames (frames 0, 227, 455, 683, and 910) were extracted and inspected visually for missed fish, small-fish failures, and coral/background false positives. This video was not treated as a conventional test set with complete ground truth.

F. Target-Domain Refinement Dataset

Thirty evenly distributed frames were extracted from the second video and annotated manually in Label Studio with a single fish class. Annotators labelled every clearly visible fish, including small fish and clearly identifiable partial fish, and did not label coral or background. Incorrect RT-DETR predictions were not used as pre-annotations. The export contained annotations for 25 of the 30 frames (frames 0, 63, 502, 533, and 565 were absent from the export). Filenames were matched to the original frames, the data were converted to COCO format, and the dataset was validated for missing files and invalid boxes. The 25 images were then divided into a 20-image training split and a 5-image validation split consisting of the five latest frames (784, 816, 847, 879, and 910). Dataset and model configuration files for domain adaptation were prepared (PResNet-18, HybridEncoder, one fish class, no pretrained backbone), but fine-tuning has not been run.

V. RESULTS AND DISCUSSION

A. Baseline Comparison

Table III summarizes the recorded baseline results, with the evaluation source indicated for each row. YOLO11n has two recorded evaluations, labelled A and B; they are listed separately and are not combined. Faster R-CNN detected fish but was slow on CPU (about 4.09 s per image) and had limited recall. SSD300 produced very few useful detections on this dataset. The RetinaNet evaluation produced no usable predictions at the selected configuration, so the mAP calculation failed and no performance figure is reported for it.

TABLE III. BASELINE DETECTOR RESULTS

Model	mAP@50	mAP@50:95	Other result	Speed
YOLO11n (eval. A)	0.5331	0.2273	mAP@75 0.1542; fish AP 0.2343	~17.74 FPS
YOLO11n (eval. B)	0.582	0.248	P 0.650; R 0.518; fish P 0.706, R 0.599, mAP@50 0.664	~60.5 ms/img (CPU)
Faster R-CNN	—	—	TP 79; FP 31; FN 61; P 0.7182; R 0.5643; F1 0.6320	~0.24 FPS
RetinaNet	—	—	No usable predictions (empty score set)	—
SSD300	0.0110	0.0018	mAP@75 0.0000; fish AP 0.0030	~2.40 FPS
RT-DETR (epoch 50)†	0.60264	0.28847	Strongest examined checkpoint	CPU (not timed)

† RT-DETR values are COCO-style AP50 and AP@[0.50:0.95] recorded during training. They come from a different evaluation protocol than the other rows and should not be read as a direct ranking.

B. RT-DETR Checkpoint Results

Among the checkpoints examined, epoch 50 gave the highest AP and AP50 (Table IV); epochs 51 to 53 were lower. Epoch 50 is therefore described as the strongest examined checkpoint, not as the globally best checkpoint, since the complete training log was not reviewed.

TABLE IV. RT-DETR CHECKPOINT RESULTS

Epoch	AP@[0.50:0.95]	AP50
50	0.28847	0.60264
51	0.23410	0.50677
52	0.19984	0.50976
53	0.20314	0.49461

C. Automatic Annotation Results

Table V lists the automatic annotation runs. With the 0.30 threshold, 61 images were processed, 41 contained detected fish, and 466 boxes were produced. On the 89-second 4K video, 2,247 of 2,668 frames contained at least one detection,

(a)



with 4,984 boxes in total. These are detection counts. They are not accuracy, precision, or recall, because no complete ground truth exists for these videos, and a count of boxes does not indicate how many are correct.

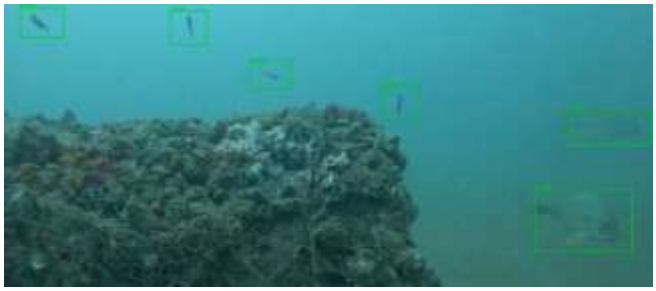
TABLE V. AUTOMATIC ANNOTATION EXPERIMENTS

Experiment	Frames	Frames with detections	Detected boxes
Image set (conf. 0.30)	61	41	466
89-s 4K video (conf. 0.30)	2,668	2,247	4,984
911-frame video	911	not verified	qualitative test

Fig. 2 shows representative frames from the annotated output. Visual inspection suggests that the detector localizes both small and larger fish against reef and open-water backgrounds, for example in panels (b). Panels (a) and (c) show the opposite behaviour: in (a), several boxes fall on rock or coral regions in a dark scene where no fish is clearly visible, and in (c) one box falls on a seaweed or debris cluster. These observations are qualitative, no per-frame ground truth was used, and the confidence values shown on the boxes are low in the difficult cases.

(b)





(c)

Fig. 2. Representative frames from the automatically annotated output video. (a), (c) Boxes placed on rock/coral or seaweed-like regions (apparent false detections). (b) Boxes on visible fish, including small fish. Judgements are by visual inspection only.

D. Domain-Shift Analysis

Applying the epoch-50 detector to the 911-frame, 1280×720 video exposed a clear generalization problem. Visible fish were missed, small fish were frequently not detected, and coral structures were sometimes detected as fish. In frame 0, which contains no fish, coral or background produced a false-positive detection. In frames 227 and 455, fish were visible but the detector mainly responded to coral or background structures instead.

These observations indicate that the problem is not simply an unsuitable confidence threshold. Lowering the threshold may recover weak detections, but it would be expected to increase background false positives as well. A more likely explanation is that the video differs from the training domain in resolution, lighting, background composition, fish appearance, scene density, and camera characteristics. A model that performs reasonably in its original domain does not necessarily generalize to another underwater domain.

E. Target-Domain Refinement Dataset

The manually annotated target-domain set contains 25 images and 253 fish boxes in one category. Validation found no missing files, no invalid boxes, and a uniform image size of 1280×720. Table VI gives the split. The validation split deliberately contains the later frames and many fish (162 boxes in five images) so that it can serve as an unseen target-domain evaluation set for future work.

TABLE VI. TARGET-DOMAIN REFINEMENT DATASET

Dataset	Images	Fish annotations
Complete annotated set	25	253
Training split	20	91
Unseen validation split	5	162

This dataset was created to address the domain shift observed when the original RT-DETR model was applied to a new video. At present it is a prepared resource, not evidence of improved performance.

F. Discussion and Limitations

The experiments indicate that a detector can serve as an annotation assistant for underwater fish, but also that its usefulness depends strongly on how closely the new video matches the training domain. Coral false positives and missed small fish directly reduce the quality of automatically generated labels, which is why the generated boxes are treated as predictions that require review.

Several limitations apply. The annotated datasets are small, and the validation split has only five images. The 89-second and 911-frame videos have no complete ground truth, so their counts cannot be converted into accuracy figures. The initial 105-image dataset was built from a mix of manual and automatically generated boxes, and the baseline results come from separate evaluations with different metrics. The domain-specific dataset has not yet been used for fine-tuning. All experiments ran on CPU. Future evaluation should use a larger verified target-domain test set and report precision, recall, F1-score, AP50, AP@ [0.50:0.95], missed-fish counts, and coral false-positive counts, with a controlled comparison before and after fine-tuning.

VI. CONCLUSION AND FUTURE WORK

This paper examined an automatic fish-annotation workflow for underwater video, progressing from conventional detector baselines to a PResNet-18-based RT-DETR detector. The strongest examined checkpoint, at epoch 50, recorded an AP@ [0.50:0.95] of 0.28847 and an AP50 of 0.60264. Automatic annotation produced 466 boxes on 61 images and 4,984 boxes across the 2,668 frames of a 4K video; these figures are operational counts and not accuracy measurements.

Testing on a different underwater video revealed missed fish, missed small fish, and coral/background false positives, which shows that domain shift is a major limitation for practical use. In response, 25 target-domain frames were manually annotated with 253 fish and split into 20 training and 5 validation images. Future work will use this dataset for controlled domain-specific fine-tuning with quantitative before-and-after evaluation, and may also investigate integrating CGAA and WCAP into the RT-DETR framework as in UCOE-DETR, reducing coral-related false positives, improving small-fish detection, exploiting temporal consistency across frames, and evaluating on a larger and more diverse collection of underwater videos.

ACKNOWLEDGMENT

This work was carried out as part of an internship at the Aazhi Research Laboratory, Department of Electronics and Communication Engineering, National Institute of Technical Teachers Training and Research (NITTTR), Chennai. The authors gratefully acknowledge the support of NITTTR Chennai.

REFERENCES

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in Proc. IEEE CVPR, 2016, pp. 779–788.

[2] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” IEEE TPAMI, vol. 39, no. 6, pp. 1137–1149, 2017.

[3] W. Liu et al., “SSD: Single Shot MultiBox Detector,” in Proc. ECCV, 2016, pp. 21–37.

[4] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal Loss for Dense Object Detection,” IEEE TPAMI, vol. 42, no. 2, pp. 318–327, 2020.

[5] N. Carion et al., “End-to-End Object Detection with Transformers,” in Proc. ECCV, 2020, pp. 213–229.

[6] Y. Lv et al., “DETRs Beat YOLOs on Real-time Object Detection,” arXiv:2304.08069, 2023.

[7] A. Salman et al., “Fish Species Classification in Underwater Videos Using Deep Convolutional Neural Network and Data Augmentation,” in Proc. IEEE ICIP, 2016, pp. 3483–3487.

[8] B. V. Deep and R. Dash, “Underwater Fish Species Recognition Using Deep Learning Techniques,” in Proc. 6th Int. Conf. Signal Processing and Integrated Networks (SPIN), 2019, pp. 655–660.

[9] V. Pagire and A. Phadke, “Underwater Fish Detection and Classification using Deep Learning,” in Proc. Int. Conf. Intelligent Controller and Computing for Smart Power (ICICCSP), 2022, pp. 1–4.

[10] A. Mahmood et al., “Deep Learning for Coral Classification,” in Handbook of Neural Computation, Academic Press, 2017, pp. 383–401.

[11] J. Li and L. Zheng, “UCOE-DETR: An accurate detection transformer with underwater characteristic-oriented encoder,” Digital Signal Processing, vol. 176, Art. no. 106033, 2026, doi: 10.1016/j.dsp.2026.106033.