2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

EnYOLO: A Real-Time Framework for Domain-Adaptive Underwater Object Detection with Image Enhancement

Junjie Wen[1,2], Jinqiang Cui[2†], Benyun Zhao[1], Bingxin Han[1], Xuchen Liu[1], Zhi Gao[3], Ben M. Chen[1]

***Abstract*— In recent years, significant progress has been made in the field of underwater image enhancement (UIE). However, its practical utility for high-level vision tasks, such as underwater object detection (UOD) in Autonomous Underwater Vehicles (AUVs), remains relatively unexplored. It may be attributed to several factors: (1) Existing methods typically employ UIE as a pre-processing step, which inevitably introduces considerable computational overhead and latency. (2) The process of enhancing images prior to training object detectors may not necessarily yield performance improvements. (3) The complex underwater environments can induce significant domain shifts across different scenarios, seriously deteriorating the UOD performance. To address these challenges, we introduce EnYOLO, an integrated real-time framework designed for simultaneous UIE and UOD with domain-adaptation capability. Specifically, both the UIE and UOD task heads share the same network backbone and utilize a lightweight design. Furthermore, to ensure balanced training for both tasks, we present a multi-stage training strategy aimed at consistently enhancing their performance. Additionally, we propose a novel domain-adaptation strategy to align feature embeddings originating from diverse underwater environments. Comprehensive experiments demonstrate that our framework not only achieves state-of-the-art (SOTA) performance in both UIE and UOD tasks, but also shows superior adaptability when applied to different underwater scenarios. Our efficiency analysis further highlights the substantial potential of our framework for onboard deployment.**

I. INTRODUCTION

The complex underwater environments pose serious challenges that notably degrade the quality of underwater images, limiting the capabilities of AUVs to execute high-level vision tasks like UOD [1][2]. Consequently, acquiring clear underwater images with UIE methods is commonly regarded as an essential prerequisite for vison-related underwater tasks. Nevertheless, despite the rapid advancements of UIE in recent years [3][4], its practical application in underwater vision tasks, like UOD for AUVs, remains relatively unexplored. This can be ascribed to the following factors.

Firstly, using UIE as a pre-processing step for UOD unavoidably introduces additional computational latency [5][6], since it requires the system to await the result of UIE before executing the UOD task. Thereby, such approach significantly diminishes its suitability for real-time application. Moreover, most deep learning-based UIE techniques demand considerable computational resource, limiting their feasibility for practical integration into real-world AUV deployments.

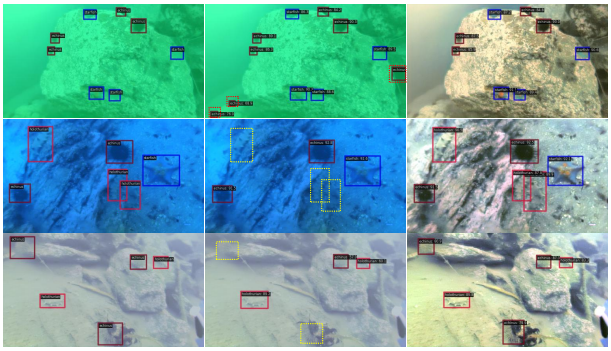


Fig. 1: Visualization of detection results in greenish, bluish, and turbid underwater environments. Our proposed EnYOLO conducts simultaneous UIE and UOD effectively. Yellow dotted rectangles indicate missed detections, while red dotted ones represent incorrect detections. Zoom in for a better view.

Secondly, directly feeding the enhanced images generated by UIE to an object detector may not lead to a performance improvement. This is because most SOTA UIE techniques are primarily designed to produce visually appealing images, which may not align with the requirements of the UOD task [7]. Additionally, the enhanced images may introduce artifacts that could confound the object detector [8].

Lastly, the complex underwater conditions frequently lead to various imaging effects (Fig. 1), causing significant domain shift across different scenarios. It suggests that a network trained in specific underwater conditions may encounter challenges when adapting to different scenarios. While previous researchers have explored domain adaptation for UIE [4][9], there has been limited investigation into addressing the domain adaptation challenge related to UOD.

To address the above issues, we present EnYOLO, an integrated real-time framework for simultaneously performing UIE and UOD with domain-adaptive capability. To be specific, both the UIE and UOD task heads leverage the same network backbone and employ lightweight architectures. Furthermore, we introduce a multi-stage training approach to maintain the balance in training both tasks, with the overarching goal of consistently improving their performance. Additionally, we propose a novel domain-adaptation method to mitigate the domain gaps across various underwater environments. Our main contributions are listed as follows:

- We propose a unified framework capable of real-time simultaneous execution of UIE and UOD tasks.

[1]Department of Mechanical and Automation Engineering, the Chinese University of Hong Kong, Shatin, N.T., Hong Kong.
[2]Department of Mathematics and Theories, Peng Cheng Laboratory, Shenzhen, China.
[3]School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, China.
[†]Corresponding author.

- We introduce a multi-stage training strategy aimed at consistently boosting both UIE and UOD tasks.
- We present a novel domain-adaptation technique to mitigate the domain shift problem for UOD.
- Extensive experiments demonstrate the effectiveness of our framework in achieving SOTA performance for both UIE and UOD while also displaying superior adaptability across diverse underwater scenarios.

II. Related Work

A. Underwater Image Enhancement

UIE techniques can be classified into traditional and learning-based approaches. While traditional methods can produce clear images by estimating backscattering and transmission under certain prior assumptions [10][11], their efficacy may decline in complex real-world scenarios.

In contrast, learning-based approaches directly acquire the mapping from degraded underwater images to their clear counterparts, exhibiting improved adaptability in complex situations. For instance, Wang *et al.* [12] introduced a Swin Transformer-based [13] UIE method that leverages local feature learning and long-range dependency modeling. Huang *et al.* [14] proposed a semi-supervised UIE technique incorporating contrastive regularization to enhance the visual quality of underwater images. However, these approaches involve high computational complexity, limiting their practical feasibility for integration into real-world deployments. Although Jamieson *et al.* [15] proposed a real-time algorithm that combines latest underwater image formation model with computational efficiency of deep learning frameworks, their primary focus is on visual quality enhancement, with unexplored applicability in high-level underwater vision tasks. In this study, we have designed a lightweight UIE architecture and explored its potential in the context of UOD task.

B. Underwater Object Detection

While generic object detection techniques have made remarkable advancements in diverse terrestrial applications [16][17], the intricate underwater environments present substantial challenges to UOD.

To enhance the performance of UOD, researchers usually leverage UIE techniques as a preliminary step to enhance image quality. For example, Jiang *et al.* [5] utilized WaterNet [18] to enhance underwater image quality, subsequently improving detection performance. Fan *et al.* [19] improved detection performance by enhancing the degraded underwater images at the feature level. Enhancing the image quality before detection has also been commonly adopted in object detection for challenging terrestrial weather conditions [8][20]. However, these approaches unavoidably introduce significant computational overhead and latency. Moreover, the presence of potential artifacts in the enhanced images can lead to a drop in detection performance in some environments [7]. Therefore, Cheng *et al.* [21] proposed a multi-task framework that jointly trains both UIE and UOD tasks in an end-to-end manner. Nevertheless, this approach relies on complex network architectures to balance the training process for both tasks, making it unfeasible for real-world deployment. In this study, we design a straightforward framework that unifies both UIE and UOD, incorporating a multi-stage training strategy aimed at consistently improving the performance of both tasks.

C. Underwater Domain Adaptation

Domain adaptation techniques have been studied in a wide variety of tasks by mitigating the feature distribution shifts between various domains [22][23]. In the context of underwater vision tasks, domain adaptation is mainly discussed in UIE. For instance, Uplavikar *et al.* [24] proposed a UIE network to handle the diversity of water types by adversarially learning the domain agnostic features. Chen *et al.* introduced a domain adaptation UIE framework by utilizing content and style separations in various domains. Wen *et al.* [4] developed a UIE network featuring both inter- and intra-domain adaptation strategies, enhancing its adaptability across a range of underwater scenarios. However, current techniques have primarily been confined to the realm of UIE, with limited exploration in the domain adaptation of UOD. Although Liu *et al.* [25] proposed a domain generalization approach using WCT2 style transfer [26] to enhance the capabilities of underwater object detector, their approach did not address the issue of domain distribution shift. In this study, we have introduced a simple but effective domain adaptation strategy for UOD, leveraging the enhanced feature embeddings derived from UIE.

III. Proposed Method

A. Problem Definition

Our proposed framework aims to effectively enable both UIE and UOD simultaneously. The datasets for training in this study are defined as follows. As depicted in Fig. 2(a), we use a paired synthetic underwater dataset $D_{ps} = \{(x_s, \hat{x}_s)_i, i \in [1, n_s]\}$ to facilitate the training of the UIE task; where n_s represents the synthetic dataset size, x_s denotes the degraded synthetic underwater image, and $\hat{x}_s$ is the corresponding clear image. For the training of UOD task, we utilize a labeled real-world underwater dataset $D_{lr} = \{(x_r, b_r, c_r)_i, i \in [1, n_r]\}$; where n_r refers to the real-world dataset size, x_r represents the real-world underwater image, b_r indicates the bounding box annotations, and c_r is the class labels. Furthermore, real-world underwater images from D_{lr} also constitute an unpaired real-world underwater dataset $D_{ur} = \{(x_r)_i, i \in [1, n_r]\}$ to enhance the performance of UIE module in real-world scenarios. Additionally, the enhanced result $\tilde{x}_r$ of UIE for each x_r, together with their corresponding b_r and c_r, formulate a labeled enhanced real-world dataset $D_{le} = \{(\tilde{x_r}, b_r, c_r)_i, i \in [1, n_r]\}$, which is also utilized for training the UOD task. During the inference, the network takes real-world underwater images x_r and subsequently predicts both enhanced images $\tilde{x}_r$ and detection results $(\tilde{b}_r, \tilde{c}_r)$, as shown in Fig. 2(b).

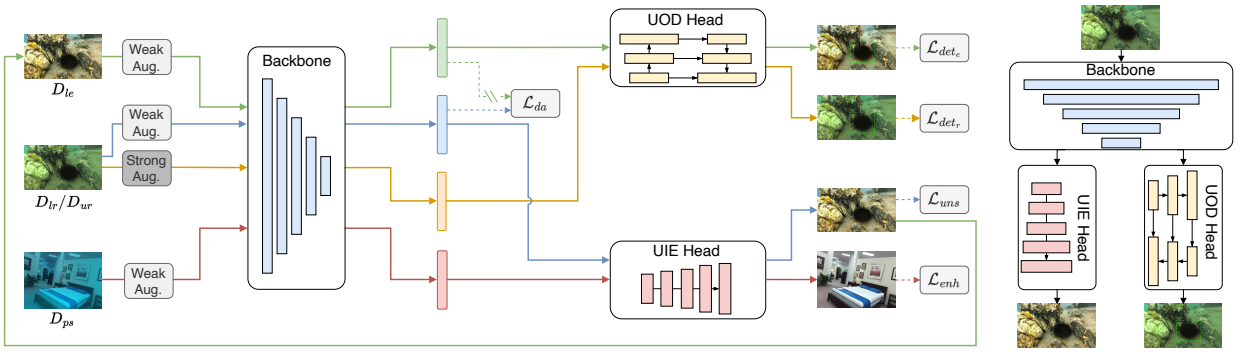


Fig. 2: Overview of our proposed EnYOLO framework. (a) The schematic illustration of training process. (b) The inference process.

B. Network Overview

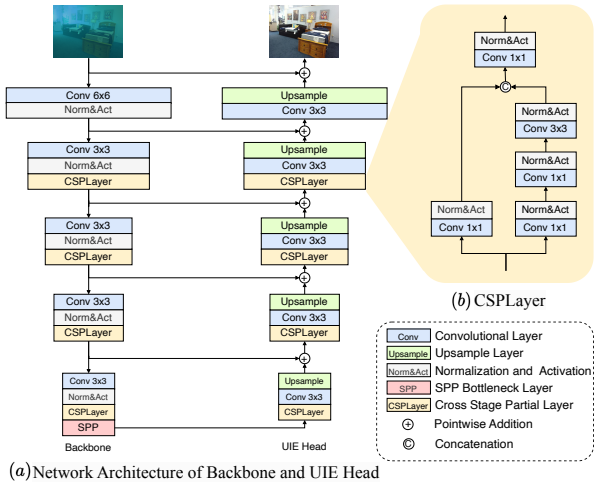
As illustrated in Fig. 2, our network architecture follows a straightforward multi-task structure, comprising three components: a network backbone, a UOD head, and a UIE head. The UIE and UOD heads share the same network backbone, with no feature exchange between them during inference. Although such design is common in various multi-task network paradigms [27][28], it holds particular significance for real-world underwater vision tasks. Firstly, during real-world testing, there is no dependency of the UOD head on the enhanced results from the UIE task, significantly reducing computational latency. Secondly, the decoupled nature of UOD and UIE heads introduces greater flexibility for real-world testing (Sec. IV-E).

It is clear that the network backbone and the UOD head constitute a general object detector. In order to enable real-time detection and maintain a lightweight architecture, we employ the classic one-stage object detector YOLOv5 [29], which incorporates CSPDarkNet53 [30] as the backbone. For the UIE head, we adopt the core concept of CSPLayer [30] to achieve a balance between network performance and computational efficiency. As illustrated in Fig. 3, the majority of convolutional filters within the CSPLayer employ a 1×1 kernel size, significantly reducing parameter count and computational demands. Moreover, the upsampling layer in our UIE head utilizes bilinear interpolation, which requires no additional network parameters and maintains computational efficiency.

C. Multi-Stage Training Strategy

The training process for both UOD and UIE is non-trivial since UIE primarily aims to enhance visual quality and fine details, while UOD mainly focuses on extracting relevant object features and their localizations. To address this issue, we introduce a multi-stage training strategy (Alg. 1) consisting of three training stages: the **Burn-In** stage, the **Mutual-Learning** stage, and the **Domain-Adaptation** stage.

(1) **Burn-In** Stage: In this stage, the entire network is trained to acquire fundamental capabilities in both UIE and UOD tasks. As shown in Fig. 2(a), the network is provided with paired synthetic underwater images D_{ps} with



(a) Network Architecture of Backbone and UIE Head

Fig. 3: Network Architecture for UIE task.

weak augmentations, including random horizontal flipping and cropping, for the UIE task. Similarly, the UOD task is optimized using the labeled real-world underwater dataset D_{lr} with strong augmentations, including mosaic patching, random color jittering, blurring, as well as random horizontal flipping and cropping. Since D_{ps} includes reference images, and D_{lr} is well-annotated, both tasks are trained in a supervised manner. During this stage, while the network acquires capability in enhancing synthetic underwater images, it may encounter difficulties when handling real-world underwater images. Similarly, although UOD is initialized with the capability to detect real-world underwater objects, it may face challenges in detecting objects across different underwater environments.

(2) **Mutual-Learning** Stage: During this stage, the performance of both UIE and UOD task is further improved through mutual knowledge learning. In the case of the UIE task, knowledge from real-world underwater images is leveraged by training with unpaired real-world underwater images D_{ur} to improve its capability in handling real-world scenarios. It is important to note that since D_{ur} lacks reference images, we apply an unsupervised loss $\mathcal{L}_{uns}$ (Sec. III-D) to guide the UIE optimization. Simultaneously, the enhanced

images $\tilde{x}_r$ generated by the UIE head are used to transfer knowledge to the UOD task, enhancing object detection in clearer underwater conditions. As illustrated in Fig. 2(a), the enhanced images $\tilde{x}_r$, along with the annotated bounding boxes b_r and class labels c_r, formulate a labeled enhanced underwater dataset D_{le} that is employed to improve the UOD performance in a supervised fashion.

(3) **Domain-Adaptation** Stage: In this stage, our goal is to mitigate domain discrepancies and enhance UOD performance in diverse underwater environments. It is assumed that after sufficient training iterations, the domain gap in the enhanced real-world underwater images generated by the UIE head has been effectively reduced [4]. Consequently, the feature embeddings of the enhanced images $E(\tilde{x}_r)$ can be considered domain invariant, while the embeddings of the original real-world underwater images $E(x_r)$ are adjusted to align with $E(\tilde{x}_r)$. Instead of employing adversarial learning for feature alignment [23], which requires additional network architectures and increases training complexity, we adopt a simpler approach by fixing the domain invariant embeddings $E(\tilde{x}_r)$ and reducing the domain shift of $E(x_r)$ with our proposed domain-adaptation loss $\mathcal{L}_{da}$ (Sec. III-D).

Algorithm 1 Multi-Stage Training Strategy

Input: Datasets D_{ps}, D_{lr}, D_{ur}; Burn-In step N_b, Mutual-Learning step N_m, Maximum step N, $N_b < N_m < N$.
Output: Final EnYOLO network Θ^N.

```
Initialize EnYOLO network. Θ^0.
for i ← 0 to N do
    Feed with D_ps and D_lr.
    Compute L_total ← L_enh + L_det_r.
    if N_b ≤ i then
        Feed with D_ur, and formulate D_le, then feed D_le.
        Compute L_total ← L_total + L_uns + L_det_e.
    end if
    if N_m ≤ i then
        Get E(x̃_r) and E(x_r).
        Compute L_total ← L_total + L_da.
    end if
    Optimize EnYOLO by minimizing L_total.
end for
```

D. Loss Design

As shown in Alg. 1, the total loss $\mathcal{L}_{total}$ is the accumulation of losses at different training stages, so we will introduce the loss functions at each training stage sequentially.

1) Burn-In Stage: In this stage, the UIE task is trained in a supervised manner with paired dataset $D_{ps} = \{(x_s, \hat{x}_s)\}$, then the enhancement loss $\mathcal{L}_{enh}$ is formulated as:

$$\mathcal{L}_{enh} = ||\tilde{x}_s - \hat{x}_s|| \tag{1}$$

where $\tilde{x}_s$ denotes the enhanced synthetic image generated by UIE head, and $||\cdot||$ represents the L1 norm.

The UOD task is optimized with labeled real-world dataset $D_{lr} = \{(x_r, b_r, c_r)\}$. Then, the detection loss $\mathcal{L}_{det_r}$ is:

$$\mathcal{L}_{det_r} = \mathcal{L}^c(x_r, b_r, c_r) + \mathcal{L}^b(x_r, b_r, c_r) + \mathcal{L}^o(x_r, b_r, c_r), \tag{2}$$

where $\mathcal{L}^c$ is the classification loss, $\mathcal{L}^b$ is the bounding box loss, and $\mathcal{L}^o$ is the objectness loss [29].

2) Mutual-Learning Stage: In this stage, besides D_{ps}, the UIE task is also optimized with an unpaired real-world dataset D_{ur}. It is assumed that the enhanced real-world images should comply with certain principles inherent to clear natural images. Inspired by [15], we propose an unsupervised loss $\mathcal{L}_{uns}$ based on "gray-world" assumption, expressed as:

$$\mathcal{L}_{uns} = \frac{1}{3}\sum_c \left(\left(\frac{1}{N} \sum_{i,j} \tilde{x}_r(i,j) \right) - 0.5 \right)^2, \tag{3}$$

where $\tilde{x}_r$ represents the enhanced real-world underwater images, c is the color channel, N is the number of pixels, and (i, j) denotes the pixel position.

For the UOD task, we replace x_r with $\tilde{x}_r$, then the detection loss $\mathcal{L}_{det_e}$ is expressed as:

$$\mathcal{L}_{det_e} = \mathcal{L}^c(\tilde{x}_r, b_r, c_r) + \mathcal{L}^b(\tilde{x}_r, b_r, c_r) + \mathcal{L}^o(\tilde{x}_r, b_r, c_r), \tag{4}$$

3) Domain-Adaptation Stage: In this stage, we align $E(x_r)$ with $E(\tilde{x}_r)$ by our proposed domain-adaptation loss $\mathcal{L}_{da}$. Inspired by [31], $\mathcal{L}_{da}$ reduces the domain discrepancy between $E(\tilde{x}_r)$ and $E(x_r)$ by minimizing their mean-squared error and covariance distances, which is:

$$\mathcal{L}_{da} = \text{MSE}(E(x_r), E(\tilde{x}_r)) + ||C(E(x_r)) - C(E(\tilde{x}_r))||_F^2, \tag{5}$$

where $C(\cdot)$ denotes the feature covariance matrice, $\text{MSE}(\cdot)$ represents the mean-squared error, and $||\cdot||_F^2$ is the squared matrix Frobenius norm.

IV. EXPERIMENTS

A. Implementation Details

Datasets: The paired synthetic underwater dataset D_{ps} employed for UIE are extracted from the Syrea dataset [4]. This dataset comprises a total of 20,688 training pairs. The labeled dataset used for UOD is derived from DUO dataset [32], with 6,671 training images and 1,111 testing images. The DUO dataset covers four distinct labeled categories: holothurian, echinus, scallop, and starfish.

Training Setups: The YOLOv5 detector [29] serves as our baseline network, and CSPDarkNet53 [33] is employed as the backbone for our proposed EnYOLO. The network is trained using the SGD optimizer for a total of $N = 150k$ steps with the Burn-In step set to $N_b = 80k$ and the Mutual-Learning step set to $N_m = 120k$. The base learning rate is set as $l_r = 0.02$, decaying by a factor of 0.1 at the epoch N_b and N_m, respectively. For the UOD task, a batch-size of 16 is employed, while a batch-size of 8 is utilized for the UIE task. Our framework is trained using PyTorch on two NVIDIA 3090 GPUs.

B. Experiment on Underwater Image Enhancement

We evaluate the UIE performance of our proposed EnYOLO through a comprehensive comparison with other SOTA UIE techniques using real-world underwater images, which are selected from the UIEB dataset [34] and the DUO Test set [32]. The compared methods include the traditional approach UDCP [10] and learning-based methods such as UNet [35], GLNet [36], and SyreaNet [4].

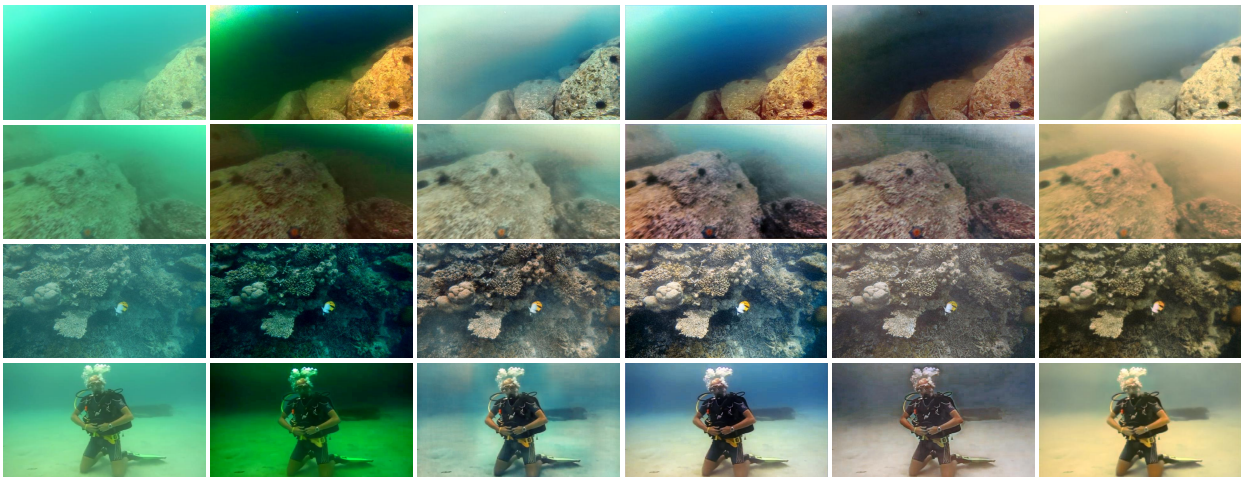


Fig. 4: Visual comparison between various UIE methods. The first two rows contain sample images from the DUO test set [32], and the last two rows contain sample images from the UIEB dataset [34]. Zoom in for a better view.

Fig. 4 illustrates the visual comparisons. As shown, UDCP fails to effectively address the greenish/bluish color cast, and SyreaNet introduces noticeable artifacts. While UNet and GLNet can produce visually appealing results, they introduce obvious artifacts (Row 4 for UNet) and shift the original greenish color cast to a bluish hue (Row 1 and 4 for GLNet), which could be detrimental for UOD (Sec. IV-C). In contrast, our proposed EnYOLO not only effectively mitigates the greenish/bluish effect but also ensures no artifacts that could potentially confound UOD task are introduced. It's worth noting that while EnYOLO demonstrates impressive performance in handling real-world underwater images across various environments, it may not excel as much in restoring fine details. This trade-off is necessary to strike a balance between the objectives of both UIE and UOD tasks. We also conduct a quantitative evaluation using Image Quality Assessment (IQA) metrics specifically designed for underwater images, including UIQM [37], UCIQE [38], and the learning-based URanker [38]. The results are presented in Tab. I, demonstrating comparable performance by our method EnYOLO against other SOTA methods.

TABLE I: Quantitative comparison of UIE performance for EnYOLO and other SOTA methods.

Methods	UIQM↑	UCIQE↑	URanker↑
UDCP	1.290	0.556	1.411
UNet	1.371	0.553	1.528
GLNet	<u>1.541</u>	**0.619**	<u>1.812</u>
SyreaNet	**1.656**	<u>0.582</u>	1.781
EnYOLO	1.523	0.571	**1.830**

C. Experiment on Underwater Object Detection

We also conduct a comprehensive UOD evaluation of our proposed EnYOLO against various SOTA approaches, including the baseline YOLOv5 [29], FasterRCNN [16], CSAM [5], JADSNet [21], IA-YOLO [8], DENet [39], and *Enhance*+YOLOv5. CSAM and JADSNet are two detection methods specifically designed for underwater scenarios. IA-YOLO and DENet were originally developed to address adverse challenging weather conditions in terrestrial environments. To assess their adaptability to underwater environments, we utilized their official source code and conducted retraining using the same training dataset. For *Enhance*+YOLOv5, we employed SOTA UIE methods GLNet [36] and SyreaNet [4] as the pre-processing step, and the resulting enhanced images are subsequently employed to train YOLOv5 for UOD.

Tab. II presents a thorough comparison of mAP between EnYOLO and other SOTA detection techniques. Our EnYOLO achieves a significant improvement of $+2.63\%$ in mAP over the baseline network, outperforming all other SOTA detection methods. It's noteworthy that utilizing UIE methods such as GLNet or SyreaNet as a pre-processing step yields a decreased mAP score. This indicates that while the enhancement methods aim to improve visual quality, they can have an adverse impact on the detection performance.

To further evaluate the domain adaptability of these methods, we synthesize underwater images of different environments using the DUO Test set, following the technique adopted in [4]. Visual comparisons are presented in Fig. 1. The last three columns of Tab. II show that all methods experienced a performance drop when operating in new environments, and the most substantial drop occurs in the bluish environment (-40.04% for YOLOv5), mainly due to the domain discrepancies from the training set. Our proposed EnYOLO outperforms all other methods significantly across various environments, achieving an impressive $+21.19\%$ increase in the challenging bluish environment compared to the baseline, demonstrating the remarkable adaptability of our method to diverse underwater environments.

D. Ablation Study

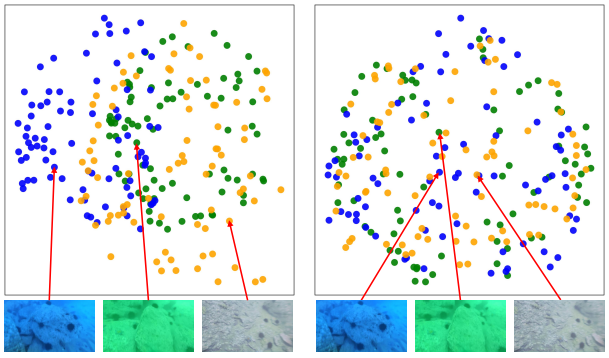
1) Effectiveness of UIE head: The effectiveness of the UIE task for UOD can be tested by removing the UIE head, which is the same with baseline model (Ab_1 in Tab. III).

TABLE II: Quantitative comparison of UOD performances for various methods on DUO Test set. The subscripts *ho*, *ec*, *sc*, and *st* respectively represent distinct categories: *holothurian*, *echinus*, *scallop*, and *starfish*. The subscripts *green*, *blue*, and *turbid* denote greenish, bluish and turbid underwater environments synthesized from DUO Test set.

Method	Backbone	mAP	mAP$_{ho}$	mAP$_{ec}$	mAP$_{sc}$	mAP$_{st}$	mAP$_{green}$	mAP$_{blue}$	mAP$_{turbid}$
YOLOv5(baseline)	CSPDarkNet53	58.08	58.18	64.88	43.45	66.81	39.26	12.04	32.48
FasterRCNN	ResNet50	57.41	59.56	62.12	41.69	65.38	25.54	9.67	26.94
CSAM	CSPDarkNet53	57.81	60.76	63.61	42.41	65.20	38.31	12.23	32.34
JADSNet	ResNet50	58.53	60.51	62.62	45.70	65.51	39.41	13.42	31.85
IA-YOLO	CSPDarkNet53	58.28	60.52	62.44	44.27	65.90	40.18	14.77	33.91
DENet	CSPDarkNet53	58.44	**61.17**	63.90	45.55	65.91	39.67	15.02	34.07
GLNet+YOLOv5	CSPDarkNet53	57.21	58.02	63.77	42.81	66.11	42.31	16.68	34.63
SyreaNet+YOLOv5	CSPDarkNet53	57.18	57.40	63.14	42.78	65.24	40.33	17.81	34.23
EnYOLO(Ours)	CSPDarkNet53	**60.71**	60.29	**65.57**	**48.45**	**68.54**	**45.73**	**33.23**	**44.31**

TABLE III: Ablation studies on various modules.

Ablations	Ab$_1$	Ab$_2$	Ab$_3$	Ab$_4$	Ab$_5$	EnYOLO
UIE head		✓	✓	✓	✓	✓
Burn-In			✓	✓	✓	✓
Mutual-Learn.		✓		✓		✓
Domain-Adapt.		✓			✓	✓
mAP	58.08	48.21	57.52	59.82	59.73	**60.71**
mAP$_{green}$	39.26	25.31	40.13	42.33	43.25	**45.71**
mAP$_{blue}$	12.04	10.29	25.45	28.97	29.88	**33.23**
mAP$_{turbid}$	32.48	23.86	39.47	41.25	40.32	**44.31**



(a) Before Domain Adaptation (b) After Domain Adaptation

Fig. 5: Visualization of tSNE feature embeddings from various underwater images. Green, blue, and orange dots denote features of images from greenish, bluish, and turbid underwater environments, respectively.

2) Effectiveness of each training stage: The effectiveness of the Burn-In stage can be seen in Ab$_2$ of Tab. III, which shows a significant decrease in mAP of -9.87% to Ab$_1$, indicating the requirement for the network to acquire basic abilities before further refinement. In contrast, Ab$_3$ exhibits only a marginal decrease in mAP compared to Ab$_1$. The effectiveness of Mutual-Learning and Domain-Adaptation stages can be verified in the results of Ab$_4$ and Ab$_5$ in Tab. III, respectively. We further visualize the influence of domain-adaptation by visualizing the backbone feature embeddings of different underwater images. As shown in Fig. 5, the feature embeddings from different underwater environments show a mixed pattern after domain-adaptation.

E. Framework Flexibility and Efficiency Analysis

We introduce three modes for real-world AUV testing: Detection, Enhance, and Dual Modes. These options allow for more flexible operation. In Detection/Enhance Mode, the AUV can only employ a single task based on specific requirements, significantly reducing computational cost. In Dual Mode, both UIE and UOD tasks run simultaneously. This enables researchers to visualize detections alongside enhanced underwater images, thus facilitating a more comprehensive understanding of the detections.

We then conduct an efficiency analysis. As illustrated in Tab. IV, in the Dual Mode, the inference latency is only slightly increased by 3.17ms compared to the Detection Mode, achieving an impressive frame rate of 74.29 fps. It significantly outperforms all other methods for simultaneous enhancement and detection (Row 3 to 5), highlighting our proposed method's superior suitability for deployment on AUV's onboard systems.

TABLE IV: Efficiency Analysis of our framework and other methods (input size 640×640, on a single NVIDIA RTX-3090 GPU)

Methods	Latency ↓	FPS ↑	GFLOPs ↓	Params ↓
YOLOv5	**10.29** ms	**97.15**	**20.04**	**20.88** M
FasterRCNN	20.44 ms	48.92	91.30	41.75 M
IA-YOLO	64.41 ms	15.53	192.81	61.70 M
GLNet+YOLOv5	93.09 ms	10.74	75.39	66.18 M
SyreaNet+YOLOv5	74.87 ms	13.36	112.53	49.88 M
EnYOLO (Det. Mod.)	**10.29** ms	**97.15**	**20.04**	**20.88** M
EnYOLO (Enh. Mod.)	11.71 ms	85.40	21.85	22.85 M
EnYOLO (Dual Mod.)	13.46 ms	74.29	24.04	31.57 M

V. CONCLUSIONS

In this study, we propose EnYOLO, a unified real-time framework designed for simultaneous UIE and UOD with domain-adaptive capability. The UIE and UOD heads share the same backbone and employ lightweight designs. To balance dual training, we introduce a multi-stage training strategy aimed at consistently improving the performance of both tasks. A novel domain-adaptation approach to mitigate domain shifts for UOD is also proposed. Extensive experimentation attests that EnYOLO achieves SOTA performance in both UIE and UOD tasks, while demonstrating superior UOD adaptability across varying underwater environments. The efficiency analysis highlights EnYOLO's impressive real-time performance, demonstrating its substantial potential for onboard deployment.

REFERENCES

[1] A. Jesus, C. Zito, C. Tortorici, E. Roura, and G. De Masi, "Underwater object classification and detection: first results and open challenges," *OCEANS 2022-Chennai*, pp. 1–6, 2022.

[2] K. Katija, "Autonomous agents for observing marine life," *Science Robotics*, vol. 8, no. 80, p. eadi6428, 2023.

[3] D. Akkaynak and T. Treibitz, "Sea-thru: A method for removing water from underwater images," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1682–1691, 2019.

[4] J. Wen, J. Cui, Z. Zhao, R. Yan, Z. Gao, L. Dou, and B. M. Chen, "Syreanet: A physically guided underwater image enhancement framework integrating synthetic and real images," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5177–5183, 2023.

[5] L. Jiang, Y. Wang, Q. Jia, S. Xu, Y. Liu, X. Fan, H. Li, R. Liu, X. Xue, and R. Wang, "Underwater species detection using channel sharpening attention," in *Proceedings of the 29th ACM International Conference on Multimedia*, pp. 4259–4267, 2021.

[6] F. Zocco, C.-I. Huang, H.-C. Wang, M. O. Khyam, and M. Van, "Towards more efficient efficientdets and low-light real-time marine debris detection," *ArXiv*, vol. abs/2203.07155, 2022.

[7] S. Sun, W. Ren, T. Wang, and X. Cao, "Rethinking image restoration for object detection," *Advances in Neural Information Processing Systems*, vol. 35, pp. 4461–4474, 2022.

[8] W. Liu, G. Ren, R. Yu, S. Guo, J. Zhu, and L. Zhang, "Image-adaptive yolo for object detection in adverse weather conditions," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, pp. 1792–1800, 2022.

[9] Y.-W. Chen and S.-C. Pei, "Domain adaptation for underwater image enhancement via content and style separation," *arXiv preprint arXiv:2202.08537*, 2022.

[10] P. Drews, E. Nascimento, F. Moraes, S. Botelho, and M. Campos, "Transmission estimation in underwater single images," in *Proceedings of the IEEE international conference on computer vision workshops*, pp. 825–830, 2013.

[11] D. Berman, T. Treibitz, and S. Avidan, "Diving into haze-lines: Color restoration of underwater images," in *Proc. British Machine Vision Conference (BMVC)*, vol. 1, 2017.

[12] R. Wang, Y. Zhang, and J. Zhang, "An efficient swin transformer-based method for underwater image enhancement," *Multimedia Tools and Applications*, vol. 82, no. 12, pp. 18691–18708, 2023.

[13] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

[14] S. Huang, K. Wang, H. Liu, J. Chen, and Y. Li, "Contrastive semi-supervised learning for underwater image restoration via reliable bank," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18145–18155, 2023.

[15] S. Jamieson, J. P. How, and Y. Girdhar, "Deepseecolor: Realtime adaptive color correction for autonomous underwater vehicles via deep learning methods," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3095–3101, 2023.

[16] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.

[17] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[18] M. A. Syariz, C.-H. Lin, M. V. Nguyen, L. M. Jaelani, and A. C. Blanco, "Waternet: A convolutional neural network for chlorophyll-a concentration retrieval," *Remote Sensing*, vol. 12, no. 12, p. 1966, 2020.

[19] B. Fan, W. Chen, Y. Cong, and J. Tian, "Dual refinement underwater object detection network," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, pp. 275–291, Springer, 2020.

[20] C. Li, H. Zhou, Y. Liu, C. Yang, Y. Xie, Z. Li, and L. Zhu, "Detection-friendly dehazing: Object detection in real-world hazy scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[21] N. Cheng, H. Xie, X. Zhu, and H. Wang, "Joint image enhancement learning for marine object detection in natural scene," *Engineering Applications of Artificial Intelligence*, vol. 120, p. 105905, 2023.

[22] X. Liu, Z. Gao, and B. M. Chen, "Ipmgan: Integrating physical model and generative adversarial network for underwater image enhancement," *Neurocomputing*, vol. 453, pp. 538–551, 2021.

[23] Y.-J. Li, X. Dai, C.-Y. Ma, Y.-C. Liu, K. Chen, B. Wu, Z. He, K. Kitani, and P. Vajda, "Cross-domain adaptive teacher for object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7581–7590, 2022.

[24] P. M. Uplavikar, Z. Wu, and Z. Wang, "All-in-one underwater image enhancement using domain-adversarial learning.," in *CVPR workshops*, pp. 1–8, 2019.

[25] H. Liu, P. Song, and R. Ding, "Towards domain generalization in underwater object detection," in *2020 IEEE INTERNATIONAL CONFERENCE ON IMAGE PROCESSING (ICIP)*, pp. 1971–1975, IEEE, 2020.

[26] J. Yoo, Y. Uh, S. Chun, B. Kang, and J.-W. Ha, "Photorealistic style transfer via wavelet transforms," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9036–9045, 2019.

[27] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[28] D. Wang, J. Wen, Y. Wang, X. Huang, and F. Pei, "End-to-end self-driving using deep neural networks with multi-auxiliary tasks," *Automotive Innovation*, vol. 2, pp. 127–136, 2019.

[29] G. Jocher, "YOLOv5 by Ultralytics," May 2020.

[30] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "Cspnet: A new backbone that can enhance learning capability of cnn," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 390–391, 2020.

[31] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14*, pp. 443–450, Springer, 2016.

[32] C. Liu, H. Li, S. Wang, M. Zhu, D. Wang, X. Fan, and Z. Wang, "A dataset and benchmark of underwater object detection for robot picking," in *2021 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pp. 1–6, IEEE, 2021.

[33] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[34] C. Li, C. Guo, W. Ren, R. Cong, J. Hou, S. Kwong, and D. Tao, "An underwater image enhancement benchmark dataset and beyond," *IEEE Transactions on Image Processing*, vol. 29, pp. 4376–4389, 2019.

[35] T. Falk, D. Mai, R. Bensch, Ö. Çiçek, A. Abdulkadir, Y. Marrakchi, A. Böhm, J. Deubner, Z. Jäckel, K. Seiwald, *et al.*, "U-net: deep learning for cell counting, detection, and morphometry," *Nature methods*, vol. 16, no. 1, pp. 67–70, 2019.

[36] X. Fu and X. Cao, "Underwater image enhancement with global–local networks and compressed-histogram equalization," *Signal Processing: Image Communication*, vol. 86, p. 115892, 2020.

[37] K. Panetta, C. Gao, and S. Agaian, "Human-visual-system-inspired underwater image quality measures," *IEEE Journal of Oceanic Engineering*, vol. 41, no. 3, pp. 541–551, 2015.

[38] M. Yang and A. Sowmya, "An underwater color image quality evaluation metric," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 6062–6071, 2015.

[39] Q. Qin, K. Chang, M. Huang, and G. Li, "Denet: Detection-driven enhancement network for object detection under adverse weather conditions," in *Proceedings of the Asian Conference on Computer Vision*, pp. 2813–2829, 2022.