Underwater Image Enhancement Based On Knowledge Distillation

Lijun Zhang[1] and Xiucheng Wu[2]

[1] School of Automation, China University of Geosciences, Wuhan, No. 388, Lumo Road,
Hongshan District, Wuhan 430074, China
lijunzh@cug.edu.cn

[2] School of Automation, China University of Geosciences, Wuhan, No. 388, Lumo Road,
Hongshan District, Wuhan 430074, China
xiuchengwu@cug.edu.cn

Abstract. In underwater environments where color distortion and significant attenuation of visible light spectra occur, image enhancement techniques are widely applied to improve the performance of object detection and recognition. However, traditional image enhancement algorithms have certain limitations in terms of processing speed and flexibility, while currently popular deep learning-based image enhancement methods demand increasingly large computational resources, making them challenging to apply in engineering practice. To address this issue, this paper proposes the FUNIE-CLAHE architecture, designed to enhance the performance of the YOLOv5s object detection model while ensuring processing speed. The FUNIE-CLAHE architecture utilizes the output images from the FUNIE-GAN network generator to fit the output of the traditional image enhancement algorithm CLAHE, thereby significantly improving image quality. The enhanced images are used as input for the YOLOv5s model. Experimental results on the TrashICRA19 dataset show that this method increases mAP by 8.6% compared to unenhanced input, 14.3% compared to input enhanced by the FUNIE-GAN network, and 6% compared to input enhanced by the traditional CLAHE algorithm. The proposed FUNIE-CLAHE architecture not only has advantages in computational efficiency but also effectively improves the detection performance of YOLOv5s while maintaining processing speed (FPS). This study demonstrates the potential of combining deep learning and traditional algorithms for computer vision tasks.

Keywords: Underwater Image Enhancement, Knowledge Distillation, Object Detection

1 Introduction

In recent years, underwater object detection has garnered extensive attention and application in fields such as marine environmental protection, underwater resource exploration, marine biology research, and underwater archaeology. However, the complex lighting conditions, suspended particles, and water scattering effects in underwater environments significantly degrade image quality. Underwater images often exhibit color distortion, such as green and blue hues, due to the varying attenuation rates of red, green, and blue light[1]. Additionally, particles suspended in water absorb a significant portion of the light before it reaches the camera, altering the light's direction and resulting in low contrast, blurriness, and haziness. These degraded underwater images increase the difficulty of detection tasks.

Recently, many deep learning-based methods have been proposed to address the problem of image restoration. Extensive research has been conducted in the specific field of underwater image restoration, demonstrating that deep learning methods perform better in complex environments compared to traditional image processing techniques. Traditional methods typically rely on prior knowledge and manually designed features, which often show limitations when faced with the diverse and dynamic underwater environment. In contrast, deep learning-based methods can automatically learn complex features and structures of images, achieving better restoration results under a broader range of conditions.

Despite achieving good results on some datasets, most deep learning-based methods are trained in a supervised manner, relying on paired datasets of low-quality and high-quality images. This supervised learning approach depends heavily on large annotated datasets, which are expensive and time-consuming to obtain in practical applications, especially in the field of underwater image enhancement. Currently, many underwater paired datasets consist of images synthesized using physical models or Generative Adversarial Networks (GAN)[2]. Moreover, the variability in underwater lighting and particle interference makes it difficult for these paired datasets to comprehensively cover all complex situations, limiting the model's generalization ability.

On the other hand, traditional underwater image enhancement methods, while advantageous in computational complexity and real-time performance compared to deep learning methods, often yield limited enhancement effects. Traditional methods rely on predefined physical models and algorithmic rules, usually optimized for specific scenarios, lacking adaptability to diverse and complex environments. These methods may fail to achieve consistent image quality improvement under different underwater conditions, and they often miss subtle image features. For example, varying depths and lighting conditions underwater result in different degrees of scattering, absorption, and reflection, leading to reduced contrast, color shifts, and blurred details. Although traditional methods can be optimized for specific situations, they often fall short of achieving ideal enhancement in diverse underwater environments. In contrast, deep learning methods, trained on large-scale underwater image datasets, can learn more complex and abstract feature representations, adapting better to diverse and complex underwater environments. Deep learning models can automatically learn patterns of optical characteristics changes, better preserving detail information and color accuracy during image enhancement.

Additionally, traditional metrics such as PSNR and SSIM, while reflecting reconstruction quality to some extent, often do not fully align with human subjective perception in underwater environments. Underwater images are affected by light attenuation, color shifts, scattering, and noise, complicating image quality assessment. Recently proposed metrics such as UIQI[3], UCIQE[4] , and UIQM[5] consider structural and color information of images, improving evaluation accuracy to some extent. However, Wang et al[6]. have demonstrated that image evaluation metrics are not always positively correlated with object detection accuracy.

To address these issues, this paper proposes the FUNIE-CLAHE architecture, which generates high-quality enhanced images by learning the output characteristics of traditional algorithms, without relying on large annotated datasets. Our FUNIE-CLAHE architecture not only has significant advantages in computational efficiency but also maintains stable enhancement effects in complex underwater environments. We use the mAP (mean Average Precision) metric from the YOLO object detection network to evaluate the enhanced images.

Experimental results show that using the enhanced images generated by the FUNIE-CLAHE architecture as input for the YOLOv5s model significantly improves the mAP metric on the TrashICRA19[18] dataset . Compared to traditional methods, our method enhances image quality and detection performance while significantly reducing computational costs and dependence on annotated data.

In summary, our main contributions are as follows: (1) This study presents an innovative image enhancement method that combines the strengths of traditional algorithms and deep learning, achieving efficient underwater image enhancement and significant improvement in object detection performance. (2) We use the mAP metric to evaluate image quality in practical application scenarios, considering both image quality improvement and performance enhancement in object detection tasks, ensuring practical significance and operability of our evaluation. (3) Extensive experiments demonstrate the effectiveness of our proposed method.

2 Related Work

2.1 Underwater Image Enhancement

Traditional underwater image enhancement methods can be mainly divided into two categories: those based on physical models and those not based on physical models[7] . Methods based on physical models[8] enhance underwater images by establishing optical models that simulate the physical processes of light propagation, scattering, and absorption during underwater imaging. These methods typically rely on modeling the physical characteristics of the underwater environment, such as the absorption coefficient, scattering coefficient, and reflectivity of seawater. The images are then corrected and enhanced based on the established physical model. Although these methods have a certain scientific basis in theory, their practical application is often limited by the complexity of the underwater environment and the uncertainty of parameters, making accurate image enhancement challenging.

Non-physical model-based methods primarily rely on image processing techniques and mathematical methods such as filtering[9], histogram equalization[10], and wavelet transform[11]. These methods do not require modeling the physical properties of the underwater environment but directly process image pixels to achieve enhancement. While these methods can achieve good results in certain situations, their enhancement effects are often not ideal due to the lack of modeling of underwater optical processes. They struggle to adapt to different underwater environments and lighting conditions.

In recent years, deep learning-based methods for underwater image enhancement have emerged. These methods enhance underwater images efficiently by learning complex features and patterns through deep neural network models. Compared to traditional methods based on physical models or image processing techniques, deep learning methods can better capture high-level features and semantic information in images, resulting in higher-quality image enhancement.

One common deep learning-based method for underwater image enhancement is using Convolutional Neural Networks for end-to-end learning and optimization[12]. These methods train deep CNN models to learn intrinsic features and structural information from a large number of underwater image datasets, and then use the trained models to enhance underwater images. By stacking multiple convolutional layers and residual connections, these methods achieve efficient enhancement of underwater images.

In addition to CNN-based methods, there are also underwater image enhancement methods based on Generative Adversarial Networks[13]. These methods train generator and discriminator networks to work together to generate realistic underwater images, thereby enhancing them. Through adversarial training, GANs learn the real distribution of underwater images, producing more realistic and clear images.

Moreover, some underwater image enhancement methods incorporate attention mechanisms[2]. By introducing attention mechanisms into deep learning models, these methods better capture and utilize important feature information in images. Attention mechanisms dynamically focus on different regions of the image, improving the model's performance in dealing with complex and variable underwater environments.

However, most existing models cannot guarantee robustness and real-time performance across different datasets. The difficulty in obtaining and annotating high-quality underwater image data further limits their generalization in practical engineering applications. Figure 1 visualizes the output images of eight underwater image enhancement algorithms compared to the output of the FUNIE-CLAHE architecture. It is evident that the images generated by the FUNIE-CLAHE architecture are visually superior to those produced by other enhancement algorithms. The enhanced images not only perform better in terms of contrast and color restoration but also retain more detail, making them clearer and more natural. Quantitative analysis in Figure 3 also demonstrates that the FUNIE-CLAHE architecture has a significant advantage over other underwater image algorithms.

2.2 Knowledge Distillation

Knowledge Distillation is a machine learning technique aimed at transferring knowledge from a large, complex model (Teacher Model) to a smaller and more efficient model (Student Model), thereby enhancing the performance of the latter. This method was first introduced by Hinton et al [14]. in 2015 and has since been widely applied in model compression, transfer learning, and multi-task learning.

The core idea of Knowledge Distillation is to use the soft labels generated by the teacher model to train the student model. Soft labels refer to the probability distribution at the output layer of the teacher model, rather than the hard labels commonly used in traditional training, which are the one-hot vectors of the categories. Soft labels contain rich information about the similarities between data categories, helping the student model to learn and generalize better.

When training the student model, a combination of the traditional cross-entropy loss function (based on hard labels) and the distillation loss function (based on soft labels) is used. The distillation loss function is typically the Kullback-Leibler divergence, which measures the difference between the probability distribution output by the student model and the soft labels from the teacher model. In this way, the student model learns not only the features of the data but also the teacher model's understanding of the relationships between data categories. By transferring knowledge from the large model to the smaller model, the parameter count and computational complexity are significantly reduced, making the model more suitable for resource-constrained environments such as mobile devices and embedded systems.

In recent years, knowledge distillation technology has achieved significant developments and progress in both academic research and practical applications. For example, multi-teacher models and multi-task distillation[15], self-distillation[16], and hierarchical distillation[17] have been explored. Knowledge distillation technology has found extensive applications not only in image classification but also in natural language processing, speech recognition, recommendation systems, and other fields. With the continuous advancement of deep learning technology and the increasing availability of computational resources, knowledge distillation technology is expected to play a crucial role in a broader range of areas, driving the development and application of AI.

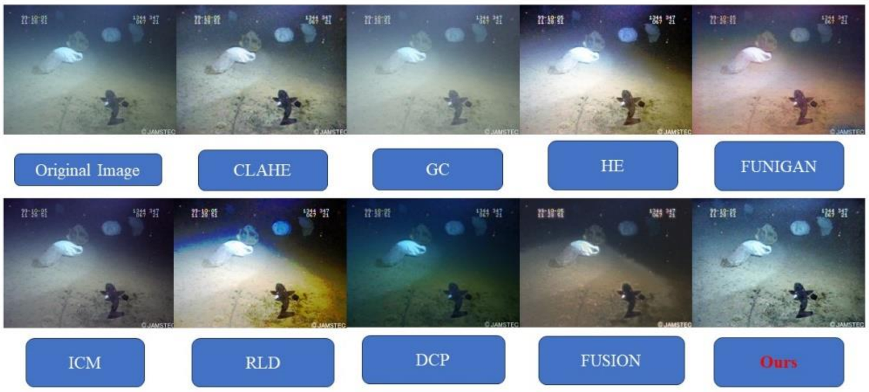

Fig. 1. Visualization of outputs from different underwater image enhancement algorithms. The Original Image is a randomly selected frame from the TrashICRA19[18] test video stream. Ours refers to the enhanced image using the FUNIE-CLAHE architecture. CLAHE[10], GC[19], HE[20], FUNIGAN[13], ICM[21], RLD[22], DCP[23], FUSION[24].

3 Method

3.1 FUNIE-CLAHE Architecture

Underwater environments are complex and highly variable, with different depths, lighting conditions, and levels of turbidity affecting image quality. Existing image enhancement algorithms often rely on paired datasets or specific underwater scenarios, limiting their generalization ability in practical applications. To address this issue, we propose the FUNIE-CLAHE architecture, as shown in Figure 2.

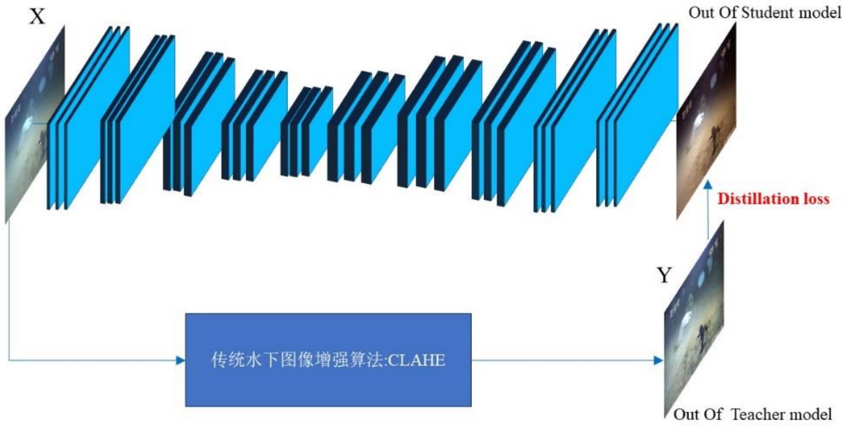

Fig. 2. FUNIE-CLAHE Network Architecture

For an unpaired distorted image X, we use the traditional underwater image enhancement algorithm CLAHE to compute the enhanced image Y. Our goal is to learn a mapping F:X→Y. We adopt a knowledge distillation-based network architecture, with the traditional CLAHE algorithm serving as the teacher model and the generator of FUNIE-GAN as the student model. In this knowledge distillation framework, the teacher model CLAHE provides high-quality enhanced images Y as the learning target for the student model FUNIE. The student model FUNIE is an encoder-decoder network designed for efficient underwater image enhancement. The encoder part is responsible for extracting features from the input image, while the decoder part utilizes these features to generate the enhanced image. The encoder comprises multiple convolutional layers, each including convolution operations, batch normalization, and activation functions. Through layer-by-layer convolution, the encoder gradually extracts both low-level and high-level features, transforming the input image into a compact feature representation. The channel dimensions of these features increase from the initial 3 channels to {32, 128, 256, 512}. The decoder is symmetrical to the encoder and contains multiple deconvolution (or upsampling) layers, each also including batch normalization and activation functions. The decoder uses the features extracted by the encoder to progressively restore the spatial resolution of the image, generating the enhanced output.

3.2 Knowledge Distillation Loss

Distillation loss directly measures the difference between the images generated by the student model and the output images of the teacher model, guiding the student model to better learn the augmentation strategies of the teacher model. In the FUNIE-CLAHE architecture, the teacher model CLAHE provides a soft label T(X) to guide the student model FUNIE predicted output G(X) through the distillation loss function. This method not only allows the student model to better mimic the behavior of the teacher model but also leverages the knowledge of the teacher model to enhance the performance of the student model.

Existing methods have shown that introducing an L1 loss term in the objective function can enable the student model G(X) to sample better in the global space. Therefore, we add the following loss term in the distillation loss function:

$$LOSS_1(G,T) = L1\big(G(X),T(X)\big) \tag{1}$$

Since the outputs of the student model G(X) and the teacher model T(X) are RGB channel color images, we incorporate a content loss term in the distillation loss. Inspired by the feature loss mentioned in Yang[25], we take the output of the teacher model T(X) as the input to VGG-19 to extract high-dimensional features and compute the loss between the high-dimensional features of the student model G(X) and those of the teacher model T(X), where MSE denotes the mean squared loss:

$$LOSS_2(G,T) = MSE\left(\varphi\big(\mathrm{G(X)}\big),\varphi\big(\mathrm{T(X)}\big)\right) \tag{2}$$

Finally, the distillation loss expression is as follows:

$$LOSS(G,T) = L1\big(\mathrm{G(X)},\mathrm{T(X)}\big) + MSE\left(\varphi\big(\mathrm{G(X)}\big),\varphi\big(\mathrm{T(X)}\big)\right) \tag{3}$$

4 Experiment

We implemented the FUNIE-CLAHE architecture using the PyTorch library. It was trained and underwent distillation learning on the TrashICRA19 dataset, which consists of 5720 images, each with a resolution of 480 x 320. The enhanced images produced by FUNIE-CLAHE were then input into an object detection network. The quality of image enhancement was evaluated using the mean Average Precision (mAP) metric of the detection network. Training was conducted on an NVIDIA GeForce GTX 3080, with models undergoing 100 iterations and a batch size of 8. We compared and evaluated our proposed architecture against different enhancement algorithms and various object detection networks.

4.1 Object Detection Model Selection

Deep learning-based object detection methods can be broadly classified into two categories: two-stage detectors and one-stage detectors. Two-stage detectors perform the detection task in two stages: the first stage generates candidate regions, and the second stage classifies and refines these regions. Although this approach usually achieves high detection accuracy, it requires two processing steps, leading to high computational overhead and inference time, making it unsuitable for real-time applications. Therefore, we used the YOLO series of neural networks, a representative of one-stage detectors, to evaluate the original unenhanced TrashICRA19 dataset.

YOLO (You Only Look Once) series networks represent a significant breakthrough in object detection. Their main feature is performing object detection end-to-end through a single neural network, achieving extremely high detection speed and good accuracy. YOLOv1 initially proposed transforming the object detection problem into a regression problem, directly predicting object bounding boxes and class probabilities through a single neural network. By inputting the entire image into the network, it eliminates the sliding window and candidate region generation steps, greatly improving detection speed. It uses a convolutional neural network (CNN) to extract features, and the final convolutional layer outputs a tensor of size S×S×(B×5+C), where S is the grid size, B is the number of bounding boxes predicted per grid, and C is the number of classes. However, YOLOv1[26] performs poorly on small object detection because each grid can only predict a fixed number of bounding boxes, making it difficult to detect multiple objects within the same grid due to spatial constraints.

YOLOv2[27] introduced the anchor box mechanism, predicting bounding boxes through a predefined set of anchor boxes, enhancing the network's adaptability to objects of different scales. By dynamically changing the input image resolution during training, it improved the model's ability to detect objects of varying scales, addressing some of YOLOv1's shortcomings. YOLOv3[28] used Darknet-53 as the backbone network, with more convolutional layers and residual blocks to improve feature extraction capabilities. It adopted a feature pyramid network (FPN) to perform detection at three different scales, significantly enhancing the detection capability for small objects.

YOLOv4[29] employed CSPDarknet53 as the backbone network, introducing the cross-stage partial (CSP) structure to reduce computation and improve accuracy. It used the PANet (Path Aggregation Network) for feature fusion, further enhancing the detection capability for multi-scale objects. New techniques such as Mosaic data augmentation and Self-Adversarial Training (SAT) were introduced in the data augmentation part to improve the model's generalization ability. YOLOv5 integrated some new ideas from other detection algorithms, including the Focus structure and CSP structure, adding the FPN+PAN structure in the NECK part. With numerous improvements in architecture and optimization, it has become widely popular in the industry for its simplicity and usability.

The recently proposed YOLOv8 architecture by the authors of YOLOv5 also incorporates the CSP module idea in the Backbone, replacing the C3 module from YOLOv5 with the C2f module for further lightweight design. The Head module uses the current mainstream decoupled head structure, separating the classification and detection heads, and switches from Anchor-Based to Anchor-Free. The performance of various detection models on the TrashICRA19 dataset is shown in Table 1.

Table 1. Performance of YOLO Series Networks on the TrashICRA19 Dataset.

Model	SIZE(MB)	mAP(%)	FPS(GPU)	FPS(CPU)
Origin(YOLOv2)	200	47.9	50	3
YOLOV3	240	**50.65**	73	3
YOLOV4	250	46.30	61	2
YOLOV5S	28	48.40	**118**	5
YOLOV8S	**23**	48.10	102	5

From Table 1, we can observe the following key points regarding the performance of YOLO series networks on the TrashICRA19 dataset: Origin (YOLOv2) represents the detection results on the validation set from the original TrashICRA19 paper. YOLOv3 achieves the best performance in terms of mean Average Precision (mAP) on the TrashICRA19 dataset; YOLOv5s has the fastest running speed (frames per second, FPS) on an NVIDIA GeForce GTX 3080 GPU; YOLOv8s has the smallest model size but demonstrates poorer generalization on this dataset compared to YOLOv5s; Balancing mAP accuracy and FPS running speed, we have chosen YOLOv5s as the baseline model for subsequent evaluations of underwater image enhancement algorithms.

4.2 Quantitative Evaluation

Unlike traditional underwater image quality evaluation metrics such as PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural Similarity Index Measure), which provide objective evaluations based on the signal-to-noise ratio, contrast, and color differences of the image itself, we are driven by specific application scenarios. Therefore, we use mAP (mean Average Precision) to evaluate the performance of enhanced underwater images in object detection tasks, as illustrated in Figure 3.

In Figure 3, the two axes represent processing speed (FPS) and the detection metric (mAP). The horizontal axis shows traditional underwater image enhancement algorithms versus deep learning-based methods. The FunieGAN network, for instance, significantly outperforms traditional algorithms in terms of processing speed due to the advantage of parallel computation on hardware. However, the enhanced images from FunieGAN show a substantial drop in mAP when evaluated with YOLOv5s.

Despite being a state-of-the-art (SOTA) model in the underwater image domain as of 2023, U-shape also performs poorly in terms of mAP. On the other hand, the traditional enhancement algorithm CLAHE demonstrates stronger generalization compared to FunieGAN and U-shape. Our FUNIE-CLAHE architecture combines the strengths of both approaches, significantly improving YOLOv5s detection performance while maintaining high processing speed (FPS). Figure 4 shows the visualization results on YOLOv5s after a single image is enhanced by the enhancement algorithm.

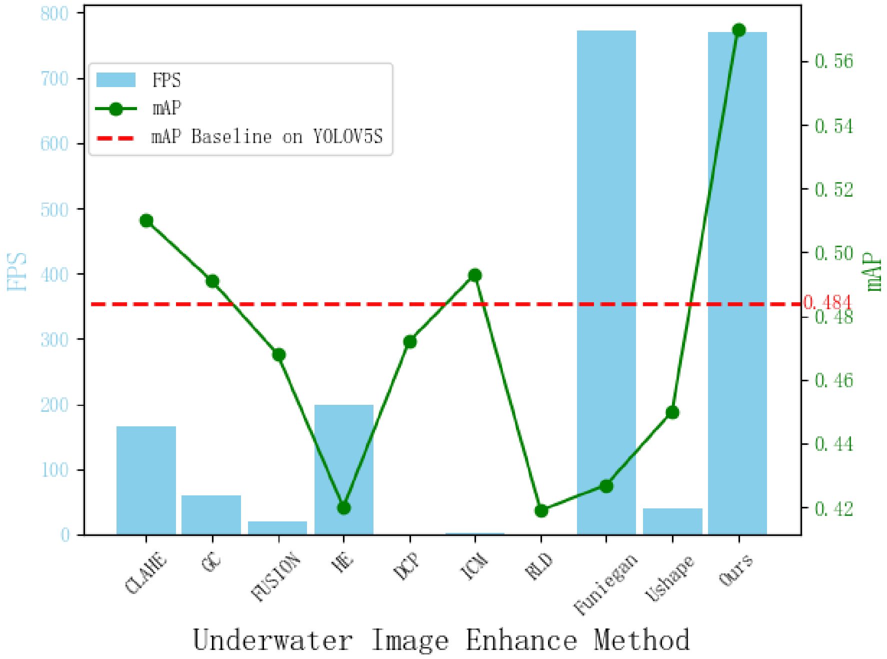

Fig. 3. Quantitative evaluation of underwater image enhancement methods. CLAHE[10], GC[19] HE[20], FUNIGAN[13], ICM[21], RLD[22], DCP[23], FUSION [24], Ushape [2].

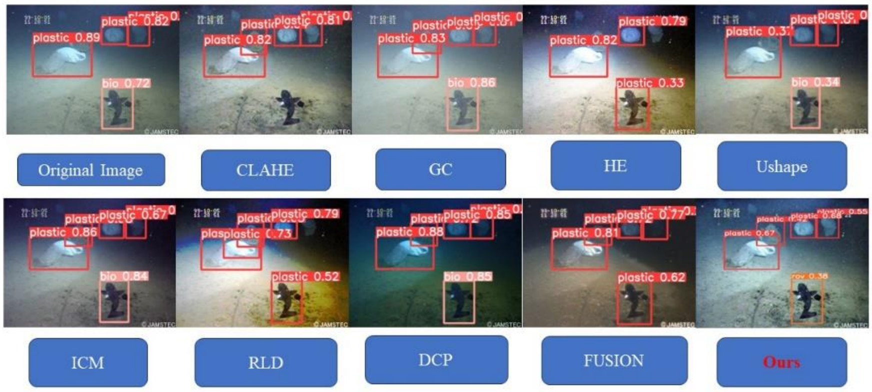

Fig. 4. Effect of Enhancement Algorithm on YOLOv5s Visualization. CLAHE[10], GC[19] HE[20], FUNIGAN[13], ICM[21], RLD[22], DCP[23], FUSION [24], Ushape [2].

5 Conclusion

The proposed FUNIE-CLAHE architecture in this paper effectively enhances the performance of the YOLOv5s model in underwater object detection tasks by learning the mapping from distorted images to enhanced images. We utilize the traditional CLAHE algorithm as the teacher model and the generator of FUNIE-GAN as the student model. By jointly optimizing feature loss, content loss, and knowledge distillation loss, the student model is able to generate high-quality enhanced images. Experimental results demonstrate that using the enhanced images generated by our proposed method as inputs to the YOLOv5s model significantly improves the mean Average Precision (mAP) metric on the TrashICRA19 dataset, validating the effectiveness and practicality of our approach. In the future, we plan to investigate its feasibility in other underwater human-machine collaboration applications, marine life recognition, and other related areas.

References

1. Zhou J, Yang T, Zhang W J A I. Underwater vision enhancement technologies: A comprehensive review, challenges, and recent trends [J]. 2023, 53(3): 3594-621.
2. Peng L, Zhu C, Bian L J I T o I P. U-shape transformer for underwater image enhancement [J]. 2023.
3. Liu Y, Gu K, Cao J, et al. UIQI: a comprehensive quality evaluation index for underwater images [J]. 2023.
4. Yang M, Sowmya A J I T o I P. An underwater color image quality evaluation metric [J]. 2015, 24(12): 6062-71.
5. Panetta K, Gao C, Agaian S J I J o O E. Human-visual-system-inspired underwater image quality measures [J]. 2015, 41(3): 541-51.
6. Liu H, Jin F, Zeng H, et al. Image enhancement guided object detection in visually degraded scenes [J]. 2023.
7. Wang Y, Song W, Fortino G, et al. An experimental-based review of image enhancement and image restoration methods for underwater imaging [J]. 2019, 7: 140233-51.
8. Peng Y-T, Chen Y-R, Chen Z, et al. Underwater image enhancement based on histogram-equalization approximation using physics-based dichromatic modeling [J]. 2022, 22(6): 2168.
9. Yu H, Li X, Lou Q, et al. Underwater image enhancement based on color-line model and homomorphic filtering [J]. 2022, 16(1): 83-91.
10. Reza A M. Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement [J]. Journal of VLSI signal processing systems for signal, image and video technology, 2004, 38: 35-44.
11. Zhang W, Zhou L, Zhuang P, et al. Underwater image enhancement via weighted wavelet visual perception fusion [J]. 2023.
12. Wang Y, Guo J, Gao H, et al. UIEC^ 2-Net: CNN-based underwater image enhancement using two color space [J]. 2021, 96: 116250.
13. Islam M J, Xia Y, Sattar J J I R, et al. Fast underwater image enhancement for improved visual perception [J]. 2020, 5(2): 3227-34.
14. Hinton G, Vinyals O, Dean J J a p a. Distilling the knowledge in a neural network [J]. 2015.

15. Zhang H, Chen D, Wang C. Confidence-aware multi-teacher knowledge distillation; proceedings of the ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), F, 2022 [C]. IEEE.
16. Xiao Z, Xing H, Zhao B, et al. Deep contrastive representation learning with self-distillation [J]. 2023.
17. Liang C, Zuo S, Zhang Q, et al. Less is more: Task-aware layer-wise distillation for language model compression; proceedings of the International Conference on Machine Learning, F, 2023 [C]. PMLR.
18. Fulton M, Hong J, Islam M J, et al. Robotic detection of marine litter using deep visual detection models; proceedings of the 2019 international conference on robotics and automation (ICRA), F, 2019 [C]. IEEE.
19. Farid H J I t o i p. Blind inverse gamma correction [J]. 2001, 10(10): 1428-33.
20. Hummel R. Image enhancement by histogram transformation [J]. 1975.
21. Iqbal K, Salam R A, Osman A, et al. Underwater Image Enhancement Using an Integrated Colour Model [J]. IAENG International Journal of computer science, 2007, 34(2).
22. Ghani A S A, Isa N A M. Enhancement of low quality underwater image through integrated global and local contrast correction [J]. Applied Soft Computing, 2015, 37: 332-44.
23. He K, Sun J, Tang X J I t o p a, et al. Single image haze removal using dark channel prior [J]. 2010, 33(12): 2341-53.
24. Ancuti C, Ancuti C O, Haber T, et al. Enhancing underwater images and videos by fusion; proceedings of the 2012 IEEE conference on computer vision and pattern recognition, F, 2012 [C]. IEEE.
25. Yang Z, Li Z, Jiang X, et al. Focal and global knowledge distillation for detectors; proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, F, 2022 [C].
26. Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection; proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, F, 2016 [C].
27. Redmon J, Farhadi A. YOLO9000: better, faster, stronger; proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, F, 2017 [C].
28. Redmon J, Farhadi A. Yolov3: An incremental improvement [J]. arXiv preprint arXiv:180402767, 2018.
29. Bochkovskiy A, Wang C-Y, Liao H-Y M. Yolov4: Optimal speed and accuracy of object detection [J]. arXiv preprint arXiv:200410934, 2020.