Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

ULGF: A Diffusion Model-Based Image Generation Framework for Underwater Object Detection

Yaoming Zhuang

zhuangyaoming524@163.com

Northeastern University https://orcid.org/0000-0001-8815-0801

Longyu Ma

Northeastern University

Jiaming Liu

Northeastern University

Yonghao Xian

Northeastern University

Baoquan Chen

Northeastern University

Li Li

Northeastern University

Chengdong Wu

Northeastern University

Wei Cui

Agency for Science, Technology and Research (A*STAR)

Zhanlin Liu

ProRata.ai

Physical Sciences - Article

Keywords:

Posted Date: October 30th, 2025

DOI: https://doi.org/10.21203/rs.3.rs-6203542/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: There is **NO** Competing Interest.

Version of Record: A version of this preprint was published at Communications Engineering on December 29th, 2025. See the published version at https://doi.org/10.1038/s44172-025-00579-z.

ULGF: A Diffusion Model-Based Image Generation Framework for Underwater Object Detection

Yaoming Zhuang[a,*], Longyu Ma[a,b], Jiaming Liu[a], Yonghao Xian[a],Baoquan Chen[a],Li Li[c], Chengdong Wu[a], Wei Cui[d], Zhanlin Liu[e]

Underwater object detection plays a crucial role in applications such as marine ecological monitoring and underwater rescue operations. However, challenges such as limited underwater data availability and low scene diversity hinder detection accuracy. In this paper, we propose the Underwater Layout-Guided Diffusion Framework (ULGF), a novel diffusion model-based framework designed to augment underwater detection datasets. Unlike conventional methods that generate underwater images by integrating in-air information, ULGF operates exclusively on a small set of underwater images and their corresponding labels, requiring no external data. Our approach enables the generation of high-fidelity, diverse, and theoretically infinite underwater images, significantly enhancing object detection performance in real-world underwater scenarios. Furthermore, we evaluate the quality of the generated underwater images, demonstrating that ULGF produces images with a smaller domain gap. We have publicly released the ULGF source code and the generated dataset for further research.

Underwater object detection is a fundamental technology in underwater robotics, with applications spanning underwater resource collection, marine scientific research, underwater archaeology, and underwater search and rescue operations[1-4]. It involves the application of object detection techniques in underwater environments, with the primary objective of accurately identifying and localizing target objects in challenging aquatic conditions.

Currently, most object detection algorithms perform poorly in underwater environments. This is primarily due to the challenges in image acquisition and the complex, dynamic conditions of the underwater world[5-8]. During dataset collection, it is difficult to flexibly control the relationships between various targets, which limits the controllability and diversity of the dataset. As a result, the lack of rich and diverse training data prevents models from effectively learning relevant underwater features, leading to poor generalization and low detection accuracy in real-world underwater object detection tasks.

Current mainstream underwater image generation methods face significant challenges in practical applications. Physics-based approaches rely on mathematical derivations but struggle to model complex underwater scenes, limiting their applicability. These methods fail to accurately capture environmental variability and fine details. On the other hand, data-driven methods, which use deep learning techniques have shown promise but require large, high-quality underwater datasets. Additionally, these methods often rely on establishing correspondences between underwater and in-air images, which is difficult. Most existing approaches generate underwater images by converting in-air ones, yet this fails to accurately replicate the unique optical properties and complexities of underwater environments, resulting in low-authenticity synthetic images.

In this study, we propose a Layout-Guided Diffusion Framework (ULGF) for underwater image generation, aiming to achieve theoretically infinite and flexible image generation for specific underwater scenes, thereby enhancing the performance of underwater object detection models (Fig.1E). The key innovation of ULGF lies in its ability to produce high-fidelity underwater images without requiring external datasets. Experimental evaluations on multiple mainstream object detection frameworks demonstrate that ULGF significantly improves detection accuracy. Additionally, we showcase how ULGF can generate high-detail underwater images from limited datasets without requiring manual annotations.

A comparative analysis with mainstream underwater image generation methods demonstrates the superior image quality and diversity of our approach. To facilitate further research in this field, we have open-sourced the code of the underwater framework and the generated underwater image dataset, allowing researchers to seamlessly apply it to underwater object detection tasks and further advance the development of underwater image generation and detection technologies.

Results

ULGF Provides a More Reliable Dataset for Underwater Object Detection

[a]Faculty of Robot Science and Engineering, Northeastern University, Shenyang 110819, China; [b]College of Information Science and Engineering, Northeastern University, Shenyang 110819, China; [c]JangHo School of Architecture, Northeastern University, Shenyang 110819, China; [d]Institute for Infocomm Research, Astar 138632, Singapore; [e]ProRata.ai, Bellevue, Washington, 98004, USA; e-mail: zhuangyaoming@mail.neu.edu.cn

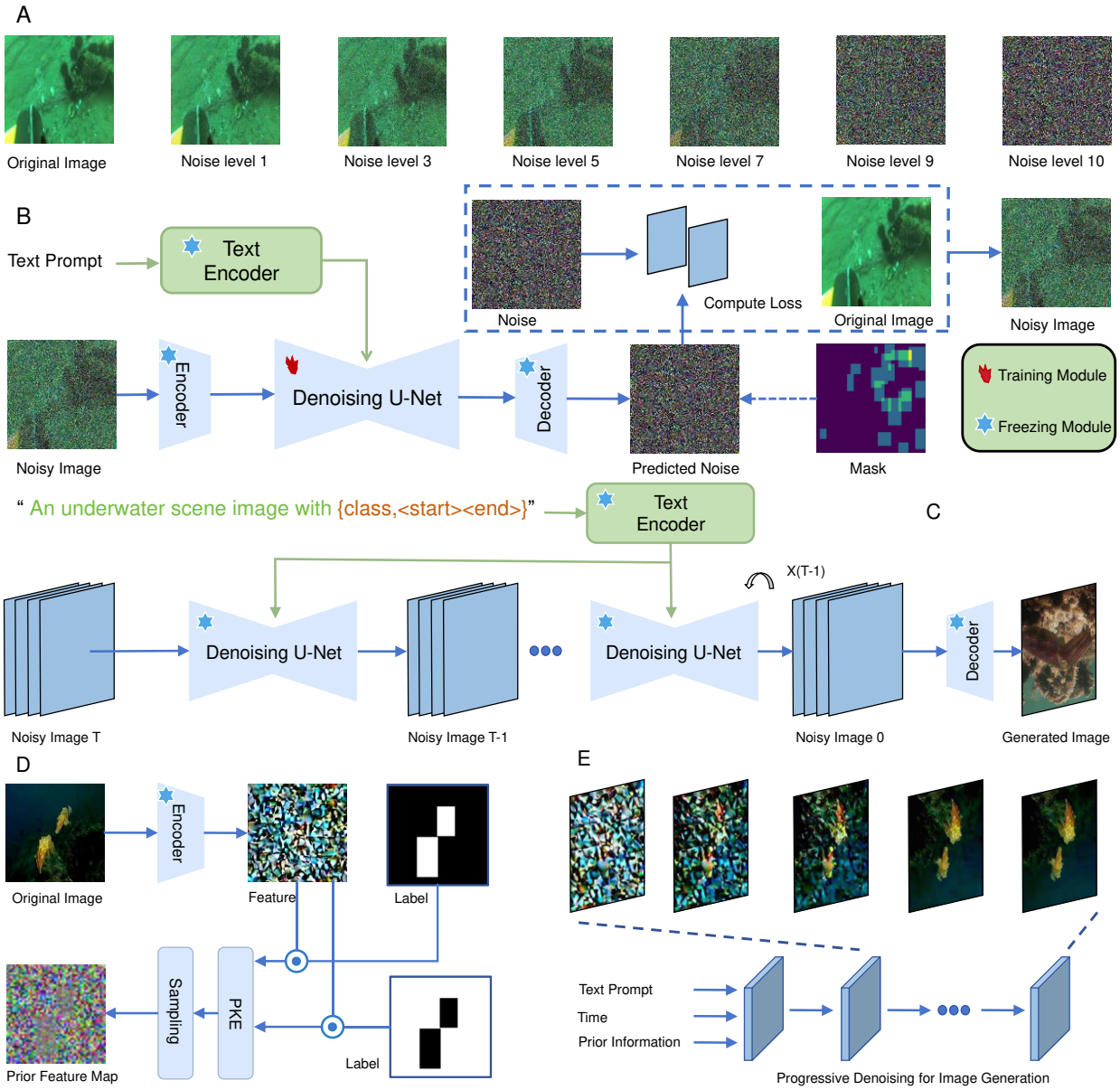


Fig.1 | Details of ULGF implementation. A Illustrates the process of adding noise to underwater images. Every image input to ULGF undergoes the addition of random horizontal noise. Assuming the original image has a noise level of 0 and the random noise level is 10, we present several typical noisy images. **B** Shows the training process of ULGF. The denoising U-Net[9] model is trained by measuring the difference between the added noise and the predicted noise. It is important to note that the text and image encoders remain frozen during this process. **C** Describes the inference process of ULGF. Through T iterations of denoising the original noise, an image with the specified layout is generated. **D** Illustrates the feature extraction process for different targets. The underwater image is encoded to extract feature information, with the label corresponding to the target's position annotation in the image, and PKE records the feature distribution of each target and environment. By sampling the characteristic distributions recorded in PKE, a prior feature distribution map with a specified layout is obtained. **E** Visualizes the process of generating denoised underwater images.

We collected a total of 10 types of images from the RUOD[10] underwater dataset (Fig.2A). The original dataset uses the YOLO label format, which includes the target class and the coordinates of the bounding box for each image. The YOLO data labels consist of the class number and the normalized bounding box coordinates for each target. To meet our task requirements, we organized and converted the RUOD dataset and transformed it into the format of {image, description, mask} for more efficient training and inference (Fig.1C).

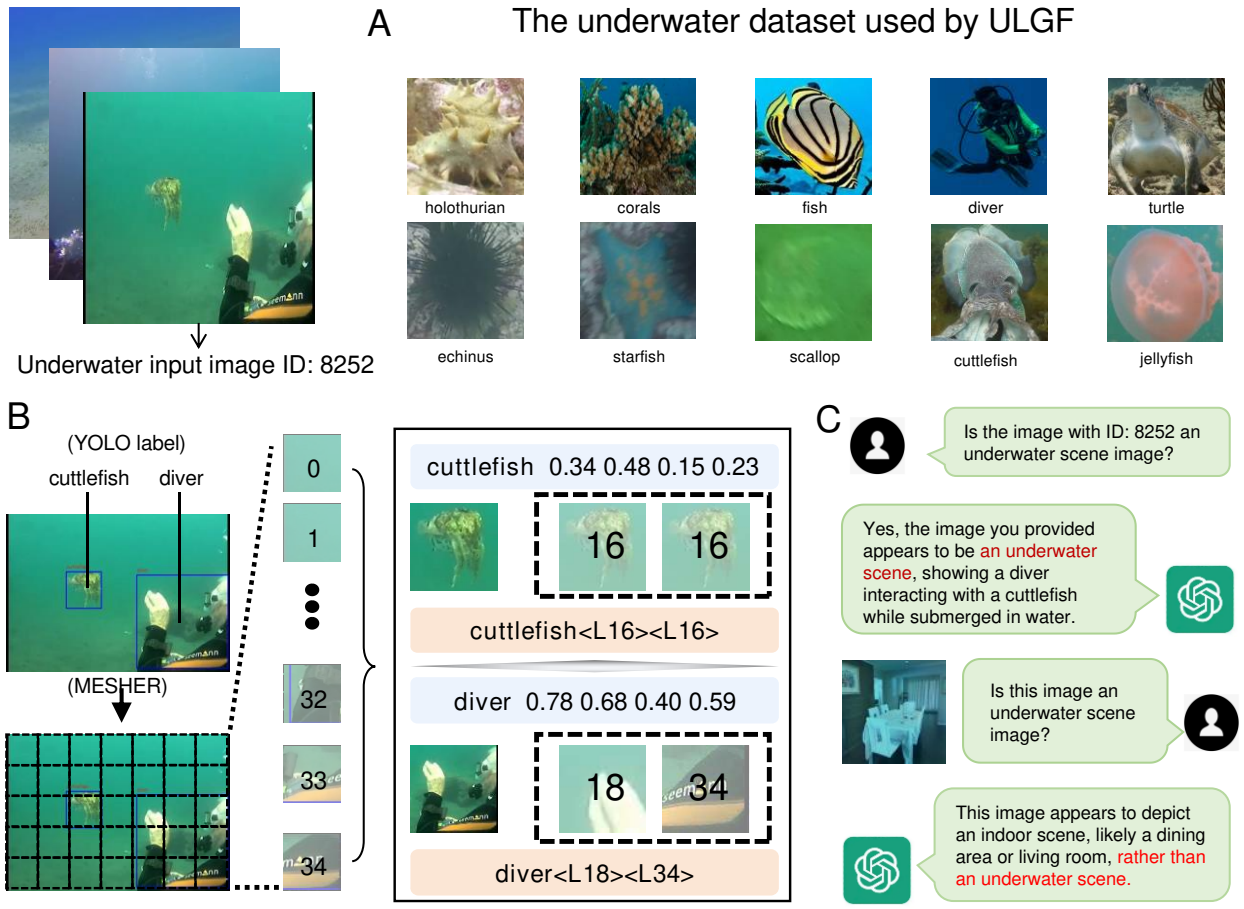


Fig.2 | Dataset preprocessing and label creation. A illustrates the dataset used for training the ULGF model. This dataset consists of ten distinct types of underwater targets. **B** presents the labeling process for underwater images. The YOLO labels from the original dataset were re-annotated using Geo, and combined with scene descriptions from C, textual prompts corresponding to underwater images were generated. **C** demonstrates the process of scene description for a given image using ChatGPT. ChatGPT can analyze the scene depicted in an input image. Additionally, C indicates that ChatGPT effectively filters out underwater images generated from in-air images, providing evidence that the vision-language model employed by ChatGPT can be utilized to assess the quality of generated images.

In the data preprocessing process, we first resize the images to ensure consistent input dimensions. Then, we randomly flip the input images with a probability of 0.5 to enhance the model's robustness. During training, we randomly remove the corresponding prompt words with a probability of 0.1. These operations help improve the model's generalization ability during training and prevent overfitting.

The input image descriptions are divided into two parts: an overall description and a target layout description. The overall description is generated with the assistance of ChatGPT to provide a concise summary of each image. This is particularly important because most existing methods generate underwater images based on land images, which ChatGPT may classify as non-typical underwater images and exclude[11]. By leveraging ChatGPT's generation capabilities, we ensure that the descriptions align with underwater environments, enhancing the dataset's semantic quality and improving training performance (Fig.2C).

The layout description of the objects records the position of each target using Geo encoding[12], specifically including the pixel coordinates of the top-left and bottom-right corners of the target. This precise spatial mapping facilitates mask generation in subsequent steps. The generated masks, derived from the layout description, play a crucial role in training by guiding loss function calculations, ensuring that the model prioritizes actual target regions. By integrating images, descriptions, and masks, the training targets can more accurately focus on the target areas, improving detection and recognition accuracy.

The trained ULGF model performs inference using only the image's overall description and the layout description of objects. This simplified input reduces computational load while maximizing the semantic information learned during training. The overall description provides a high-level semantic understanding, while the object layout description specifies precise object locations, enabling efficient and accurate inference.

To ensure the validity and diversity of the generated data for object detection, we rigorously filtered the validation set images to ensure that they do not overlap with those in the training set. Specifically, no validation image was reused from training, preventing data leakage and ensuring accurate and fair model evaluation.

To maintain label consistency and accuracy, we exclusively used underwater image labels from the training set. To enhance data richness and diversity, we limited the number of objects per image and applied data augmentation techniques, such as random flipping of bounding boxes. These strategies increase dataset variability and mitigate overfitting to specific images or scenes. The detailed methodology is provided in the Methods section.

In this study, we evaluated several mainstream object detection models to access their performance in underwater image object detection tasks. The YOLO[13] series is among the most widely used object detection models, known for its excellent detection accuracy and real-time performance. The design concept of YOLO is to transform the object detection task into a regression problem, allowing object localization and classification in a single forward pass. The YOLO series models have been continuously optimized with each version update, improving both detection accuracy and speed. We evaluated different versions of the YOLO model (YOLOv5, YOLOv6[14], YOLOv8, YOLOv11) to examine whether the generated underwater images consistently improve detection tasks as the detection model evolves.

RTDETR[15] is a recently proposed object detection model based on the Transformer architecture. Unlike traditional Convolutional Neural Network (CNN) models, the Transformer has the ability to model global information when processing image features, enabling it to capture long-range dependencies between objects in the image. RTDETR places particular emphasis on real-time performance, achieving efficient inference while maintaining accuracy, making it an important model for our evaluation.

ULGF significantly enhances target detection performance through dataset expansion (Fig.3D). We define mean Average Precision (mAP) as the average precision over IoU thresholds from 0.5 to 0.95, and mAP50 as the average precision at IoU = 0.5. As shown in Figure 3D, ULGF improves detection accuracy across all tested models. Compared to YOLOv5, ULGF achieves a 1.2% increase in mAP50 and a 0.9% increase in mAP. With YOLOv6, it improves mAP50 by 1.4% and mAP by 0.7%, while in YOLOv8, it yields a 1.1% increase in both mAP50 and mAP. For YOLOv11, ULGF achieves 1.5% higher mAP50 and 1.4% higher mAP, and in REDETR, it leads to a 1.4% improvement in mAP50 and a 0.9% improvement in mAP. These results confirm that ULGF consistently enhances underwater object detection performance. Additionally, ULGF is compatible with other detection enhancement strategies, making it a versatile augmentation approach.

We conducted a detailed analysis of category-wise accuracy improvements in object detection using ULGF. Using YOLOv8 as an example, we trained the detection model with the ULGF-augmented dataset and observed accuracy improvements across all categories. Among them, divers showed the greatest enhancement, with a 3.0% increase in AP50 and a 3.6% increase in AP (Fig. 5D). Additionally, we visualized the training process and found that after 150 epochs, the model trained with ULGF consistently outperformed the baseline model. This improvement remained stable throughout training, demonstrating the robust effectiveness of ULGF as it is shown in.

ULGF Is an Underwater Image Generation Method That Does Not Depend on In-air Images

Our framework verifies the quality and authenticity of generated underwater images by comparing them with the underwater images used during training. Specifically, we compared the underwater images generated by ULGF with the real underwater images used in training. This comparison helps evaluate whether the generated images successfully simulate the visual features of underwater environments.

To demonstrate the superiority of ULGF, we designed two sets of comparison experiments. The first set compares the underwater images generated by other methods with their corresponding land images. The goal is to test whether these methods can accurately preserve the spatial structure of the aerial images while effectively incorporating the unique visual changes of the underwater environment during generation.

The second set compares the generated underwater images with the real underwater images used in training. This experiment aims to evaluate whether these methods can maintain consistency with the training data when generating underwater images, thus verifying the stability and reliability of their generation capabilities. Since ULGF does not rely on land images, we only used real underwater images for the metric calculations in our experiments. Additionally, for methods that do not use underwater images for training, we performed metric calculations with the UIEB dataset[16].

In addition to the above metrics, we further employ the large language model ChatGPT to describe the image scenes[11], in order to validate the realism and superiority of our generated underwater images. To eliminate any human interference or bias, no guiding prompts are provided in the experiment. ChatGPT is asked to directly describe our generated underwater images as well as the underwater images generated by other methods (such as aerial image generation methods), allowing for an objective evaluation of the differences in scene realism and underwater characteristics across different generation methods (Fig.2C).

Experimental analysis shows that ULGF outperforms mainstream underwater image generation methods in Fréchet Inception Distance[17] (FID), a metric measures the similarity between generated images and real images by comparing their distribution differences in feature space. ULGF achieves an FID of 23, while UWCNN[18], UWGAN[19], UWNR[20], and UWCNN* have FID values of 218.09, 168.79, 229.863, and 225, respectively. Traditional underwater image generation methods learn the style of underwater scenes and generate underwater images based on aerial images (Fig. 5B). This approach results in generated images carrying many features of the aerial images, leading to poor FID scores. In contrast, ULGF generates images solely based on real underwater images, giving it an advantage in terms of image authenticity. Although ULGF performs slightly worse than UWNR in terms of Inception Score[21] (IS), it outperforms UWCNN, UWNR, and UWCNN*. This is because the IS metric measures the

diversity of generated images through classification[22]. Since ULGF generates images of underwater scenes and focuses on generating targets within individual images, its performance on the IS metric is relatively modest.

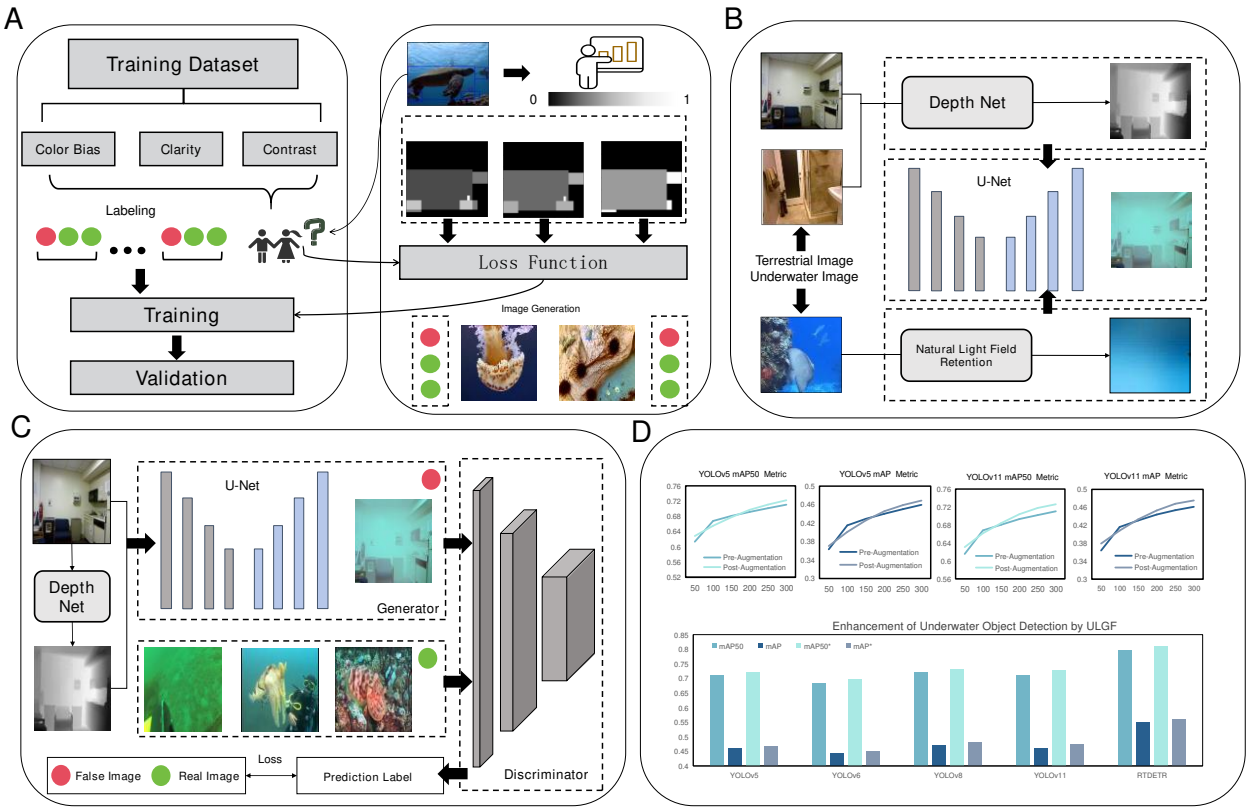


Fig.3 | Overview of object detection results and underwater image generation methods. A illustrates the process of generating images in various styles using ULGF and evaluates them based on three metrics: color bias, clarity, and contrast. During the training process, we investigate the impact of the weights assigned to occluded regions on the quality of the generated images. **B** presents a method for generating underwater images using both underwater and land images. By obtaining depth information from land images and light field information from underwater images, a U-Net network is used to render images with an underwater style. **C** follows a similar approach to generating underwater images from land images, but it guides the generator by determining whether the generated underwater images resemble real underwater images. **D** provides a detailed demonstration of the effectiveness of the dataset augmented by ULGF during training. We selected the classic YOLOv5 and YOLOv11 as representatives to observe specific training details. When the training batch exceeds 150, ULGF provides more accurate results for underwater object detection. ULGF improves the accuracy of all tested detectors. Metrics marked with * indicate results trained on the augmented dataset.

In other common evaluation metrics, such as PSNR, MSE, and SSIM[23], our method does not show a significant advantage over mainstream underwater image generation algorithms. This is because other underwater generation methods often use techniques similar to direct style transfer, which preserve the layout and structure of the original image. As a result, these methods typically achieve better results in PSNR, MSE, and similar metrics. In contrast, our generation method significantly alters the image layout to create new underwater scenes, which inevitably affects the structural similarity of the images.

We also discussed the impact of introducing prior information on image quality. Experiments show that adding a certain amount of noise at the early stage of denoising can generate more realistic underwater images (Fig. 5F). Although ULGF is designed for underwater object detection, researchers who require higher image realism can consider introducing prior information into the image denoising process (Fig. 4B).

ULGF Can Generate Underwater Images with Specific Styles and Layouts

We conducted a detailed evaluation and analysis of the style of images inputted into the ULGF model. First, we assessed the images in the existing dataset, focusing on key attributes such as color bias[24], blur, and contrast (Fig.3A). Through these evaluations, we were able to accurately identify the style characteristics of each image, thereby providing labels for the ULGF. Building on this, we successfully generated underwater images with diverse stylistic features during the image generation process, achieving varied visual effects.

During the model generation process, we specified arbitrary sizes and types of targets as conditions and, in combination with the image style information, successfully

generated realistic underwater images (Fig. 4A). These images closely resemble real-world underwater environments in terms of visual effect, demonstrating the powerful capabilities of the ULGF model in underwater image generation tasks.

Discussion

Underwater object detection aims to identify and locate target objects, such as aquatic products, personnel, or other targets by analyzing image features. With the development of underwater detection technology, underwater object detection has broad application prospects in fields such as marine research, environmental protection, and aquaculture[25]. However, dataset scarcity remains a major bottleneck in underwater object detection. Due to the unique challenges of underwater environments, acquiring sufficient labeled datasets is both costly and difficult, limiting the diversity of training data and hindering the generalization and accuracy of detection models. Additionally, underwater images are affected by factors such as water quality, lighting, and biological movement, often resulting in significant variations in image quality. In practical data collection, it is difficult to control the impact of these factors on underwater images, further exacerbating the limitations of insufficient datasets on model performance, leading to poor results in real-world detection.

In this study, we propose a framework called ULGF for underwater image generation, aiming to improve underwater object detection performance. The framework has been validated on mainstream detectors and datasets. To the best of our knowledge, this is the first work to build an underwater object detection training dataset using a layout-guided diffusion model[26]. We have open-sourced the framework and provided a generated underwater dataset with accurate annotations for use by other researchers. This framework supports custom image generation, providing a powerful tool for further development in underwater object detection.

We evaluate the proposed underwater image generation framework from three aspects: underwater image generation quality, layout accuracy, and detection accuracy. To minimize the impact of parameter settings on the results in detection tasks, we only modify the input underwater dataset during the experiments without changing other parameter configurations. To consider the specific impact of generated data on detection tasks, we maintain consistent detector parameter configurations across different tests to ensure fairness in evaluation. In the layout accuracy assessment, we use the YOLOv8 detector for testing all methods to eliminate any bias from detector selection. For the evaluation of underwater image generation quality, we compare the images generated by the proposed framework with those generated by other mainstream open-source underwater image generation methods, thereby comprehensively assessing the performance of the framework.

Currently, non-deep learning-based underwater image generation methods do not rely on large amounts of labeled data for training. Therefore, they can effectively generate underwater images in situations where data is scarce or labeled data is difficult to obtain. Furthermore, because these methods are typically based on physical models or rule-based systems, they have low computational overhead and fast processing speed, making them suitable for real-time or resource-constrained applications. However, these methods also have certain limitations, mainly due to their high dependence on condition design. Non-deep learning methods often struggle to adapt effectively to changes in different underwater domains, lacking sufficient adaptability. As a result, the generated underwater images are relatively uniform, lacking diversity and flexibility, and are difficult to handle complex and dynamic underwater environments.

Deep learning-based underwater image generation methods can be divided into supervised learning and unsupervised learning approaches. Supervised learning methods rely on the alignment between underwater and aerial images to generate and transform images. However, in practical underwater object detection scenarios, obtaining corresponding aerial images for underwater images is often not feasible. In contrast, unsupervised learning methods, while not requiring corresponding aerial image data, still require a large amount of underwater data to learn the features of underwater scenes (Fig.3B and Fig.3C). Although unsupervised methods partially overcome the data alignment issue, they still face the problem of data scarcity due to their dependence on large amounts of underwater data for learning.

From a technical implementation perspective, mainstream underwater image generation methods convert in-air images into underwater images to expand the dataset. However, these methods suffer from domain shift during the conversion process, resulting in poor image quality. There are significant visual differences between in-air and underwater images, making it difficult for the converted underwater images to accurately reflect the characteristics of the underwater environment, thereby affecting the generation quality. In underwater object detection tasks, the content of in-air images cannot effectively guide underwater object detection. in-air images fail to capture the relationship between underwater objects and their environment, with their layout and background being obviously inconsistent with actual underwater scenes. Additionally, generating underwater images based on in-air images requires collecting a large number of real images, which increases the cost of image collection and annotation.

ULGF only requires input text and layout descriptions during inference, avoiding the reliance on in-air images and their corresponding datasets. This approach not only reduces the complexity of data collection but also allows for flexible generation of underwater images that meet specific needs. It effectively mitigates the negative impact of domain shift, improving the quality and applicability of underwater image generation.

We also investigated the generation of images with different styles. To this end, we evaluated attributes such as color bias, blurriness, and contrast in images from the existing dataset and annotated each image with relevant labels. To avoid subjective factors influencing the evaluation results, we invited 8 volunteers to participate in the image assessment and discarded any images with controversial evaluations. We generated images in different styles, demonstrating the framework's adaptability to various scenes.

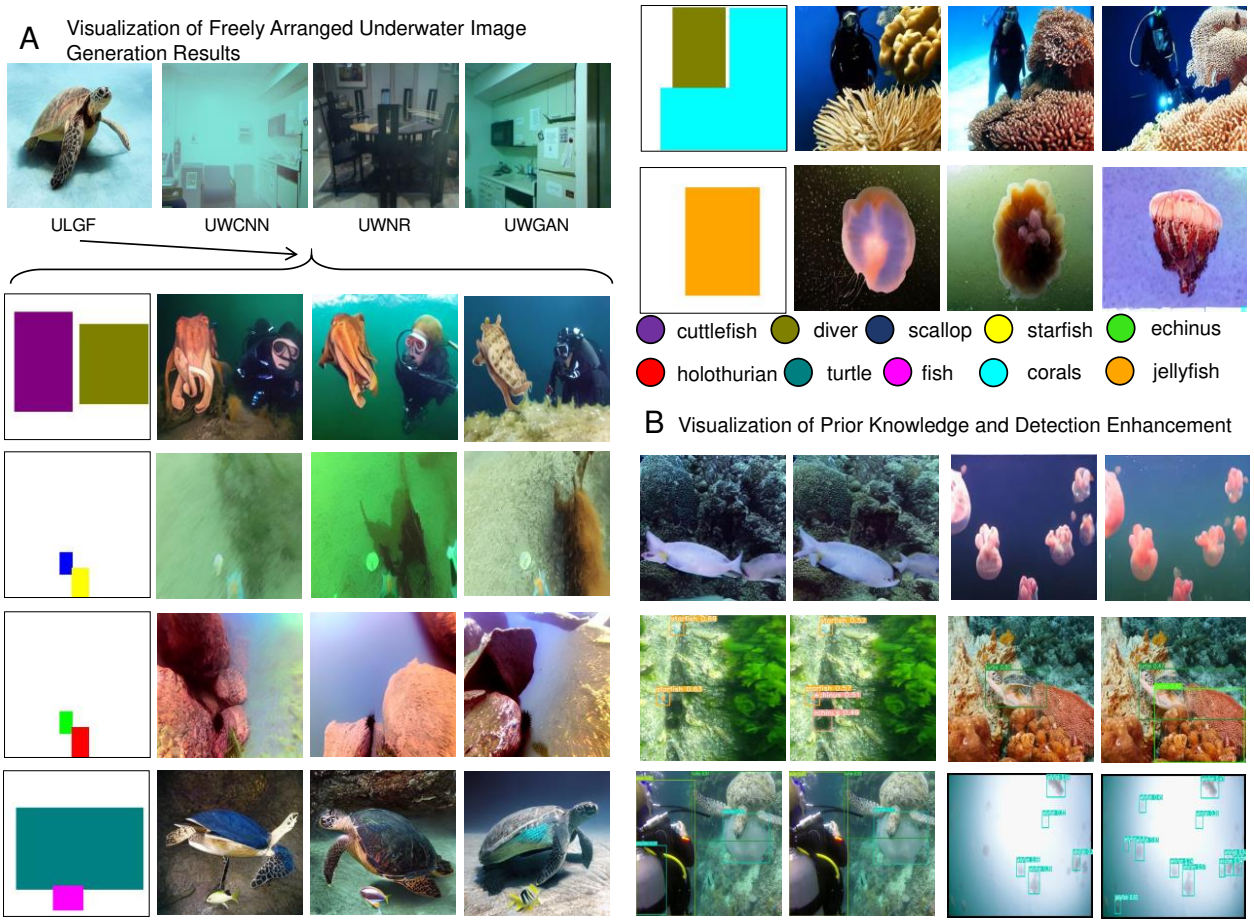


Fig. 4 | Visualization of underwater image generation and object detection results. A shows the powerful generation capabilities of ULGF. Unlike mainstream underwater image generation methods, ULGF does not require in-air images as input. ULGF is capable of generating underwater targets with specified layouts and types. Additionally, by providing different random seeds to ULGF, diverse images can be generated. In the demonstration, we randomly generate three sets of images for each layout label. **B** illustrates the effect of fusing prior knowledge with noise for image generation, as well as the improvement in detection performance on the test dataset after augmenting the dataset with ULGF. After incorporating prior knowledge, the textures of the targets appear more natural. With the augmented dataset from ULGF, the detector shows a significant reduction in both false negatives and false positives.

Although our model shows good performance in underwater image generation tasks, there is still room for improvement. It must be acknowledged that, despite the theoretically unlimited number of generated images, the data-driven training approach is heavily reliant on the dataset. In particular, when the underwater dataset is small in scale, the model's performance is affected, leading to a lack of diversity in the generated image styles. Furthermore, the image generation speed slows down as the image size increases. While this has a minimal impact on real-time underwater object detection tasks, it remains a concern. Another area for improvement is that the current model still struggles to generate realistic texture details for complex objects. Specifically, when handling fine details of underwater object surfaces or textures in complex environments, the generated results do not fully replicate the actual scene.

In addition, the training set used in our evaluation contains a relatively small amount of available underwater data, which somewhat limits the performance of the detection model. Our framework has a high requirement for label accuracy in the underwater dataset because when handling unannotated underwater objects, the generated image treats them as background and generates them randomly. Such generated data may interfere with the model's effective extraction of object features during underwater object detection training, potentially leading to missed detections in real-world scenarios. Therefore, we recommend that when selecting an underwater detection dataset, it is essential to ensure that the dataset has sufficient scene diversity and accurate labels to improve the model's training effectiveness and detection accuracy.

Future research can focus on customizing diverse underwater generation environments by incorporating factors such as lighting conditions, suspended particle distribution, and other environmental variables, thereby reducing reliance on existing datasets. Currently, research on integrating image

generation with real underwater scenes is still in its early stages, offering vast exploration opportunities. One possible direction is to further develop a multi-modal training framework that integrates factors such as underwater lighting, visibility, and light scattering, and designs underwater images in different styles. This approach could reduce dependence on specific datasets and enhance the detection model's adaptability and detection performance in unknown underwater scenarios. For example, combining physical lighting models with diffusion generation techniques (IClight[27]) could generate more realistic underwater physical images, improving the authenticity and reliability of the data. Additionally, combining targets not included in the dataset with underwater scenes could effectively enhance the robustness and generalization ability of the detection system when facing unknown objects, providing stronger support for the practical application of underwater object detection.

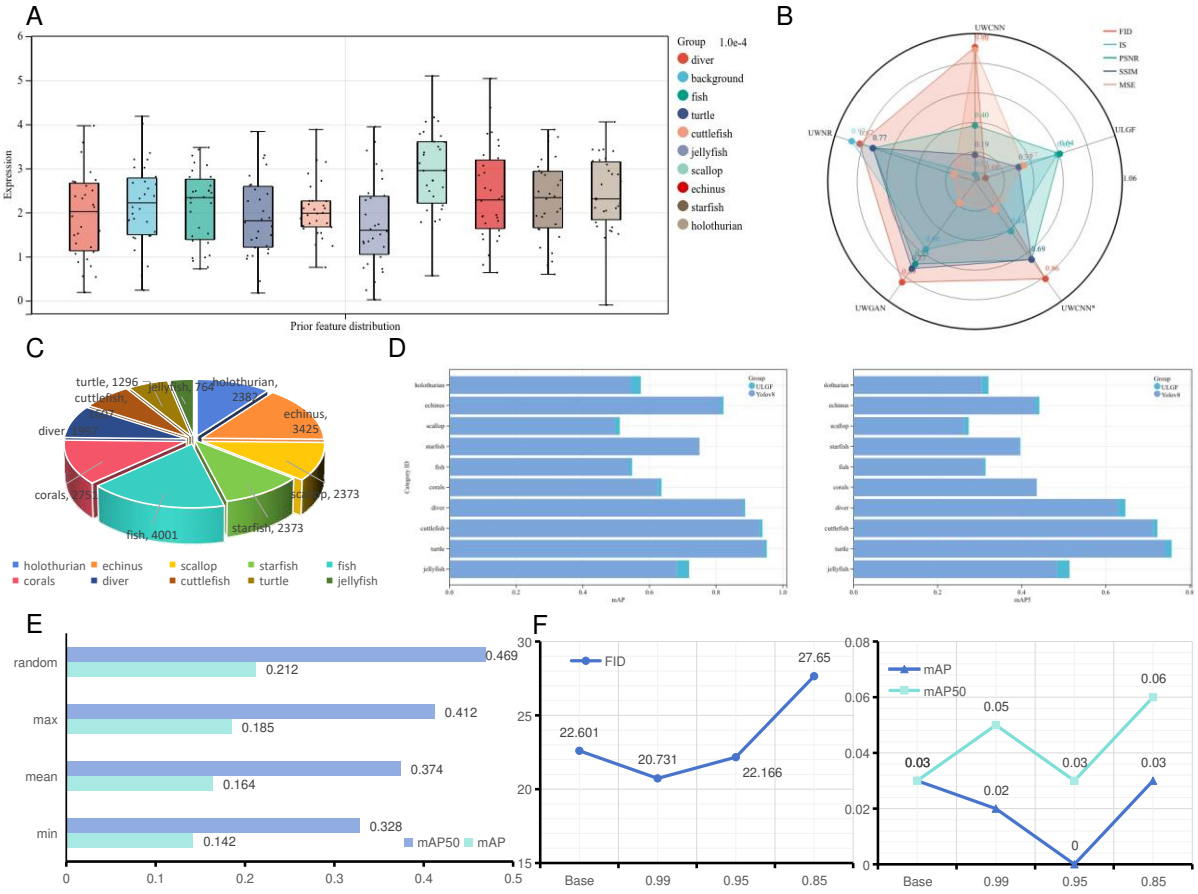


Fig. 5 | Experimental details. A shows the prior distributions of different target types. We randomly selected 30 samples for each target type for visualization, roughly simulating the prior features of each target. **B** compares the evaluation metrics of ULGF with other mainstream underwater image generation methods. We choose the UWCNNtype1 method as UWCNN and UWCNNtype9 as UWCNN*.**C** presents the count and percentage of each target category in the test dataset. **D** visualizes the accuracy improvements of ULGF for various target categories. **E** shows the variation in layout accuracy of the generated images under different occlusion strategies. **F** illustrates the impact of the level of prior knowledge fusion on the generated image's FID score and detection task accuracy.

In summary, by leveraging the advantages of layout-guided diffusion in image generation, our method exclusively relies on underwater images to produce high-quality synthetic data, eliminating the limitations of traditional aerial-to-underwater conversion methods. Compared to using aerial images, the ULGF-generated underwater images are more realistic, flexible, and controllable, allowing scene and object layouts to be adjusted as needed. This approach removes the need for manual aerial image collection and annotation, providing a novel solution to address dataset scarcity in underwater object detection.

Methods

Selection and Preprocessing of Underwater Datasets

When selecting an underwater dataset, it is essential to consider the specific needs of the detection task, ensuring that the dataset includes annotations for diverse object categories. Some underwater datasets focus on image enhancement, aiming to improve image quality through denoising and contrast adjustment. However, these datasets contain fewer samples and are not directly applicable to object detection tasks, making them unsuitable for our needs. Our framework is designed for object recognition and localization within an image, differing from certain underwater datasets that rely on paired reference images for processing. Consequently, datasets specifically designed for underwater object detection are better suited to our task requirements.

In this study, we trained our model using the object detection dataset RUOD (Fig. 5C), which contains 9,800 training images and 4,200 test images, with a total of 74,903 detection targets across 10 underwater species. In the first step of this study, we constructed an underwater image generation dataset based on the object detection dataset. This dataset consists of triplets in the format {image, description, mask}. The image component consists of underwater training images featuring diverse underwater scenes and target objects. The description is a text-based narrative of the scene and object positions, generated with the assistance of ChatGPT to provide natural language descriptions of targets and their spatial locations. The target positions are encoded using Geo's bounding box method, enabling text-driven image generation with precise target placement control (Fig.2B). The mask is used to train the framework loss, and in the latent space, the target box position is adjusted by rounding to ensure accurate alignment between the generated image and the target box.

We designed a standard data processing pipeline for training. Input images are horizontally flipped with a probability of 0.5 to enhance model robustness. The pixel values of the images are then normalized, and images with incorrect aspect ratios are padded to the required size. For each image, a maximum of 22 boxes are annotated, with any excess boxes discarded. During training, some unlabeled images are used to assist in training ULGF.

Image Generation Strategy

The generation strategy is designed based on the actual scene. In this framework, we reference the label distribution of the training set to design the types and layout of the generated images. However, this does not mean the generation scheme is fixed. In practice, our layout is flexible. If small or dense target detection is required, our framework can generate a large number of relevant and realistic images. The image generation scheme is as follows: targets with a box area smaller than 0.0001 are discarded, as such small areas cannot effectively represent the target's features. Additionally, a horizontal flip with a probability of 0.5 is applied to create detection box labels, ensuring image diversity. The number of generated images in detection tasks matches that of the training set. To ensure reproducibility, a fixed random seed is set for each image.

Network Architecture and Training

Our layout-guided diffusion-based underwater image generation framework consists of three parts: underwater description encoding, image information compression, and image denoising generation. This framework is based on the Latent Diffusion Model (LDM), with the underwater text description module using CLIP[28]. CLIP has gained multimodal learning ability through training on large-scale image-text pairs, enabling it to understand complex natural language and efficiently match it with image content. Image information compression addresses the issue of slow training and generation speed for large images. We use a Variational Autoencoder[29] (VAE) to compress the original images into latent space, reducing the image dimensions while preserving key features. We use the encoded underwater description as a generation condition, with the compressed image in latent space as input, and train the image generation model using a U-Net network (Fig.1B).

We trained for 60 epochs. During training, we froze the CLIP and VAE components and used pre-trained weights to focus on fine-tuning other parts of the model, reducing unnecessary computation. With the VAE fixed, we fine-tuned all parameters of the U-Net using the AdamW optimizer and a cosine learning rate schedule. The learning rate for U-Net was set to 1.5e−4, with a linear warm-up of 3000 steps to gradually increase the learning rate. This approach helped stabilize the training process and improve convergence. The batch size was set to 16, which was the optimal value that could be handled by our NVIDIA RTX 3090. During training, there was a 10% chance of replacing the text prompt with an empty string. This encouraged the model to generate unconditional images, reducing over-reliance on text prompts and enhancing the diversity and flexibility of the generated images.

Loss Function Design

The diffusion model learns the data distribution of real images by progressively denoising random noise sampled from a normal distribution[30]. To learn how to gradually recover images from noise, we add noise at different time steps to the real images (Fig.1A). Let x_t represent the noisy image, x_0 be the real image, ϵ_t be the noise, and α_t be the noise coefficient. The noisy image can be represented as:

$$x_t = \sqrt{\alpha_t} x_0 + \sqrt{(1-\alpha_t)}\epsilon_t \quad (1)$$

The training process of the diffusion model is the process of predicting and separating the noise in the noisy images. The loss function is designed to minimize the difference between the predicted noise and the actual added noise. In this study, we use ε to represent the noise distribution added to the image, ϵ_θ be the predicted noise distribution, z_t be the noise input, τ_θ be the conditional encoder, y be the introduced condition, ε be the variational autoencoder's processing function, x be the input image, t be the time step, $\odot$ be the pixel-wise multiplication operator, and M be the corresponding pixel value. The loss function can be expressed as:

$$Loss = E_{\varepsilon(x),\varepsilon,t}\left\|\epsilon - \epsilon_\theta\left(z_t, t, \tau_\theta(y)\right)\right\|^2 \odot M \quad (2)$$

However, in underwater object detection tasks, directly making indiscriminate predictions on random noise is clearly unreasonable. The reason is that we prefer the model to learn the specific features of different types of targets rather than the environment. In detection tasks, we are detecting targets, so it is necessary to assign higher weight to the targets and reduce the focus on background information. We refer to the

foreground and background weighting method in Geo, where smaller targets are assigned higher weights.

There are many overlapping and occluded objects in underwater scenes. Assigning different weights to foreground objects makes it difficult to balance the weight distribution in occluded areas. To help the model handle occlusion more effectively, we discuss this issue. For pixels belonging to three or fewer objects, we randomly assign weights based on the number of object categories while considering the physical occlusion relationships. If a pixel belongs to too many objects, we randomly assign one object's weight to that pixel. This strategy achieves the highest accuracy in image layout (Fig. 5E).

Underwater Image Inference

The inference process involves progressively denoising a normal distribution noise to generate a real underwater image (Fig.1C). It has been observed that the difference between the noisy images during training and the noise in inference is significant, resulting in poor realism of the generated objects and the emergence of strange structures. We drew inspiration from the prior noise construction in object segmentation tasks[31] and designed a prior image inference method specifically for underwater object detection.

We perform feature extraction on the images encoded by the VAE, with the specific details being that we classify the bounding boxes according to the number of target classes in the labels[31] (Fig. 5A). We extract the feature information of each target from the encoded latent space (Fig.1D). By selecting a high-dimensional space for information extraction, we can obtain more refined target features. We average the features of each target to reduce the influence of other types of targets and the environment that may exist in the occluded areas. Let the class be denoted as cls, P be the pixel points belonging to cls, and num be the total number of pixel points in this class. The feature distribution of num (denoted as N) can be represented as:

$$N_{cls} = N(\sum_{i=1}^{num} \frac{P_i}{num}, \sum_{i=1}^{num} \frac{1}{num}(P_i - \bar{P})^2) \quad (3)$$

In the inference process, we extract the specific underwater scene and target prior features, and then fuse them with the underwater layout prompts and random normal distribution noise. The fused image, awaiting denoising, already contains the geometric information guidance. When combined with the text prompt guidance, it generates a realistic image.

Computational Requirements

Our framework requires at least 22GB of GPU memory for training, while only 5GB of GPU memory is needed for inference. This ensures that most personal computers or laptops with dedicated GPUs can handle the computational requirements of our framework. Additionally, during underwater object detection training, with the batch size and image size remaining unchanged, the additional burden imposed by our model is minimal. We have conducted a series of comparisons with current mainstream object detection frameworks.

We evaluate our framework's performance in generating underwater object detection datasets from three key aspects. The first aspect focuses on improvements in underwater object detection performance. To ensure fair evaluation, we train the underwater image generation model using only the training set of the object detection dataset, preventing it from learning information from the validation set scenes. The mean Average Precision (mAP) serves as the evaluation metric for this aspect. Additionally, we evaluate the quality of the generated underwater images. For physical model-based and non-paired underwater image generation methods, we assess quality by comparing the generated dataset with our underwater dataset using specific metrics. For methods based on underwater-aerial image pairs, we evaluate quality by comparing the generated underwater images with their corresponding training images. The Fréchet Inception Distance (FID) serves as the evaluation metric. Next, we validate the quality of the generated underwater images. For underwater image generation methods based on physical models and non-paired underwater image generation methods, we perform an evaluation by comparing the generated underwater dataset with our underwater dataset using specific metrics. For underwater image generation methods based on underwater-aerial image pairs, we evaluate the generated underwater images by comparing them with their training images. The evaluation metric we use is FID.

Moreover, we also use the mAP metric to evaluate the layout accuracy of the generated images. At the same time, in this design, a higher mAP score indicates better authenticity of the generated images, higher accuracy, and greater alignment with the real dataset.

Data availability

In terms of data availability, the datasets and models generated or used in this study have been made available in the following ways: The RUOD dataset can be accessed at https://github.com/dlut-dimt/RUOD. Both the underwater image generation dataset and the pre-trained detection model have been open-sourced and stored on Google Drive for further research and reproduction. The weights related to ULGF are also available on Google Drive for download and evaluation.

Code availability

The code for ULGF can be found at the following URL: DOI is https://github.com/maxiaoha666/ULGF. The code is intended for academic research and non-commercial use only.

References

1. Wu, Z., Chen, X. et al. Self-Supervised Underwater Image Generation for Underwater Domain Pre-Training. IEEE Trans. Instrum. Meas. 1–14 (2024).
2. Zhou, J., He, Z., Lam, K. M. et al. AMSP-UOD: When vortex convolution and stochastic perturbation meet underwater object detection. Proc. AAAI Conf. Artif. Intell. 38, 7659–7667 (2024).
3. Zeng, L., Sun, B. & Zhu, D. Underwater target detection based on Faster R-CNN and adversarial occlusion network. Eng. Appl. Artif. Intell. 100, 0952-1976 (2021).
4. Sun, B., Zhang, W., Xing, C. & Li, Y. Underwater moving target detection and tracking based on enhanced

You Only Look Once and deep simple online and real-time tracking strategy. Eng. Appl. Artif. Intell.143, 0952-1976 (2025).
5. Cao, J., Zeng, Z., Zhang, X., Zhang, H., Fan, C., Jiang, G. & Lin, W. Unveiling the underwater world: CLIP perception model-guided underwater image enhancement. Pattern Recognit. 162, 111395 (2025).
6. Fu, Z., Wang, W., Huang, Y., Ding, X. & Ma, K. Uncertainty Inspired Underwater Image Enhancement. Proc. Eur. Conf. Comput. Vis. 465-482 (2022).
7. Zhang, W., Zhuang, P., Sun, H.-H., Li, G., Kwong, S. & Li, C. Underwater Image Enhancement via Minimal Color Loss and Locally Adaptive Contrast Enhancement. IEEE Trans. Image Process. 31, 3997–4010 (2022).
8. Peng, L., Zhu, C. & Bian, L. U-Shape Transformer for Underwater Image Enhancement. IEEE Trans. Image Process. 32, 3066–3079 (2021).
9. Ronneberger, O., Fischer, P. & Brox, T. U-net: Convolutional networks for biomedical image segmentation. Med. Image Comput. Comput. Assist. Interv. MICCAI 2015, 234–241 (Springer Int. Publ., 2015).
10. Fu, C., Liu, R., Fan, X. et al. Rethinking general underwater object detection: Datasets, challenges, and solutions. Neurocomputing 517, 243–256 (2023).
11. Zhang, F., You, S., Li, Y. et al. Atlantis: Enabling underwater depth estimation with stable diffusion. Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. 11852–11861 (2024).
12. Chen, K., Xie, E., Chen, Z. et al. GeoDiffusion: Text-prompted geometric control for object detection data generation. Proc. Int. Conf. Learn. Represent. 1–12 (2024).
13. Redmon, J., Divvala, S., Girshick, R. et al. You only look once: Unified, real-time object detection. Proc. IEEE Conf. Comput. Vis. Pattern Recognit. 779–788 (2016).
14. Li, C., Li, L., Jiang, H. et al. YOLOv6: A single-stage object detection framework for industrial applications. arXiv preprint arXiv:2209.02976 (2022).
15. Zhao, Y., Lv, W., Xu, S. et al. DETRs beat YOLOs on real-time object detection. Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. 16965–16974 (2024).
16. Li, C., Guo, C., Ren, W. et al. An underwater image enhancement benchmark dataset and beyond. IEEE Trans. Image Process. 29, 4376–4389 (2019).
17. Bynagari, N.B. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. Asian J. Appl. Sci. Eng. (2019).
18. Li, C., Anwar, S. & Porikli, F. Underwater scene prior inspired deep underwater image and video enhancement. Pattern Recognit. 98, 0031-3203 (2020).
19. Wang, N., Zhou, Y., Han, F. et al. UWGAN: Underwater GAN for real-world underwater color restoration and dehazing. arXiv preprint arXiv:1912.10269 (2019).
20. Ye, T., Chen, S., Liu, Y. et al. Underwater light field retention: Neural rendering for underwater imaging. Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. 488–497 (2022).
21. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A. & Chen, X. Improved techniques for training GANs. Proc. 30th Int. Conf. Neural Inf. Process. Syst. 2234–2242 (2016).
22. Szegedy, C., Liu, W., Jia, Y. et al. Going deeper with convolutions. Proc. IEEE Conf. Comput. Vis. Pattern Recognit. 1–9 (2015).
23. Wang, Z., Bovik, A.C., Sheikh, H.R. et al. Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. 13, 600–612 (2004).
24. Yan, J., Hu, H., Wang, Y. et al. Underwater image enhancement via multiscale disentanglement strategy. Sci Rep 15, 6076 (2025).
25. Liu, K., Ding, M., Pan, B. et al. A maneuverable underwater vehicle for near-seabed observation. Nat Commun 15, 10284 (2024).
26. Rombach, R., Blattmann, A., Lorenz, D. et al. High-resolution image synthesis with latent diffusion models. Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. 10684–10695 (2022).
27. Zhang, L., Rao, A. & Agrawala, M. Scaling In-the-Wild Training for Diffusion-based Illumination Harmonization and Editing by Imposing Consistent Light Transport. Proc. 13th Int. Conf. Learn. Represent (2024).
28. Radford, A., Kim, J.W., Hallacy, C. et al. Learning transferable visual models from natural language supervision. Proc. Int. Conf. Mach. Learn. 8748–8763 (PMLR, 2021).
29. Kingma, D.P. & Welling, M. Auto-Encoding Variational Bayes. arXiv preprint arXiv:1312.6114 (2013).
30. Ho, J., Jain, A. & Abbeel, P. Denoising diffusion probabilistic models. Adv. Neural Inf. Process. Syst. 33, 6840–6851 (2020).
31. Gao, H., Gao, M., Li, J. et al. SCP-Diff: Spatial-Categorical Joint Prior for Diffusion Based Semantic Image Synthesis. Proc. Eur. Conf. Comput. Vis. 37–54 (Springer Nature Switzerland, 2024).

Acknowledgements

This research was supported in part by the National Natural Science Foundation of China (62403108, 42301256), the Liaoning Provincial Natural Science Foundation Joint Fund (2023-MSBA-075), the Ministry of Industry and Information Technology Project (TC220H05X-04), the Scientific Research Foundation of Liaoning Provincial Education Department (LJKQR20222509), the Fundamental Research Funds for the Central Universities (N2426005).

Author contributions

Y.Z.: Data management, annotations of training dataset, data analysis and interpretation. L.M.: Data management and preparation, development and validation of algorithms, comparative experiments, data analysis and interpretation, manuscript drafting. J.L.: Comparative experiments, manuscript drafting. All authors: Manuscript review and editing, critical revision for important intellectual content.

Competing interests

The authors declare no relevant competing interests.

1