UNIOCEAN: GENERATIVE GAUSSIAN SPLATTING FOR UNIFIED UNDERWATER 3D RECONSTRUCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Underwater 3D reconstruction with Gaussian splatting has recently achieved remarkable progress. However, existing methods suffer from two critical limitations: **(1)** These methods typically rely on particle-specific modeling paradigms, limiting their adaptability across diverse underwater scenes with static, dynamic, and uncertain scattering conditions. **(2)** Underwater attenuation further exacerbates the inherent low-frequency bias of 3D Gaussians induced by their kernel distribution, leading to inferior representation of local high-frequency textures. To address these limitations, we propose **UniOcean**, a novel generative Gaussian splatting framework for unified and high-fidelity underwater 3D reconstruction. Specifically, we propose an Adaptive Particle Modeling strategy that exploits a learnable neural attenuation function to adaptively estimate depth-dependent absorption and scattering terms, thereby uniformly modeling diverse underwater particle effects. Furthermore, we design a Progressive Gaussian Generation module that leverages optimized coarse Gaussians to generate fine-grained texture Gaussians guided by the generative prior, thus yielding refined Gaussians featuring both global appearance and local textures. Benefiting from the above collaborative strategies, our method achieves high-quality representation while exhibiting excellent adaptability. Extensive experiments demonstrate that our UniOcean significantly outperforms existing methods across diverse underwater 3D scenes with static, dynamic, and varying synthetic scattering conditions.

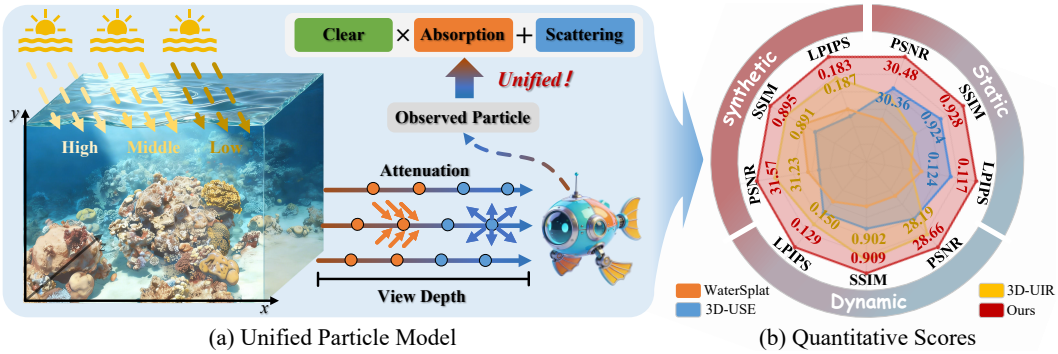
1 INTRODUCTION

Novel View Synthesis (NVS) plays a critical role in exploring complex underwater environments and holds great potential for a wide range of applications, including subsea robotic navigation and marine science Jiang et al. (2022); Cong et al. (2024). In recent years, Neural Radiance Fields (NeRF) Mildenhall et al. (2021) and 3D Gaussian Splatting (3DGS) Kerbl et al. (2023) have demonstrated remarkable success in novel view synthesis by effectively modeling geometric structure and scene appearance Chen et al. (2024); Liu et al. (2024b); Xie et al. (2025); Wu et al. (2025b); Wang et al. (2025). However, real-world underwater imaging frequently suffers from severe visual degradation, restricting their reliability in practical underwater applications.

To mitigate this dilemma, various studies have been developed to model volumetric medium effects in 3D space. Early methods typically adopt NeRF-based pipelines to implicitly encode degradation and restore the appearanc Levy et al. (2023); Tang et al. (2024). Despite the promising results, NeRF inherently suffers from high computational overhead and slow rendering speed due to dense volumetric sampling. Fortunately, the emerging 3DGS exhibits compelling advantages in both efficiency and rendering fidelity. By representing scenes with a collection of Gaussian ellipsoids, 3DGS enables real-time novel view synthesis while maintaining visually plausible appearance Kweon & Park (2026); Li et al. (2024); Liu et al. (2024a); Yuan et al. (2026a); Yang & Leonard (2024). Nevertheless, these methods still face two critical limitations: **(1)** They typically model the underwater participating medium using tailored physical models, limiting their adaptability across diverse conditions. **(2)** Underwater attenuation further compounds the inherent low-frequency bias of 3D Gaussians induced by their kernel properties, hindering the recovery of high-frequency textures.

To address the above challenges, we investigate the intrinsic optical properties of underwater particles from both spatial and frequency perspectives. On the one hand, light propagating through un-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Unified Particle Model (b) Quantitative Scores

Figure 1: We present UniOcean, a novel generative Gaussian splatting framework for unified underwater 3D reconstruction. (a) Unlike existing methods that rely on particle-specific formulations, we develop a unified particle modeling paradigm that adaptively decomposes complex optical attenuation into absorption and scattering parts. (b) Our UniOcean significantly outperforms existing methods across static, dynamic, and synthetic scattering underwater scenes.

derwater medium undergoes global attenuation governed by depth-dependent absorption and scattering effects following the Beer-Lambert law Gordon (1989), resulting in scene-independent particle distributions, as shown in Fig. 1(a). On the other hand, underwater particles inevitably suppress high-frequency textures due to complex light attenuation. Notably, unlike other signals that exhibit abrupt temporal fluctuations and broader frequency coverage to learn high-frequency information, Gaussian signal presents a relatively smooth time-domain profile and a narrowband concentration in the low-frequency region, as shown in Fig. 2(a). This suggests that 3D Gaussians inherently suffer from a low-pass limitation, preventing the initialized Gaussians from being sufficiently densified, as shown in Fig. 2(b). These observations reveal a shared attenuation pattern across diverse underwater scenes: *underwater particles usually induce similar absorption and scattering patterns that attenuate high-frequency textures.* Inspired by this, we aim to uniformly model the particles within a generative paradigm, thereby effectively handling diverse underwater conditions.

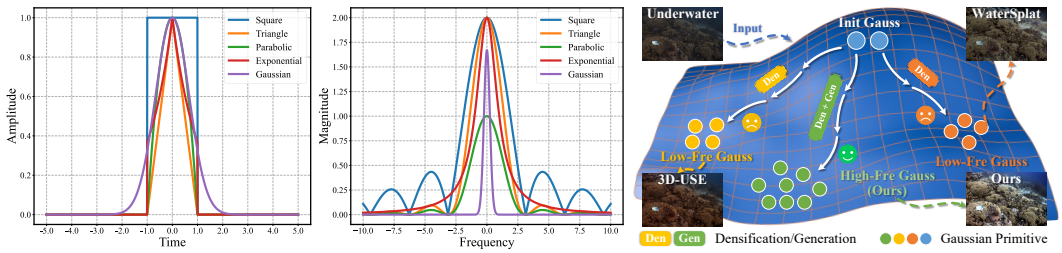
Building upon the above observations, we propose a novel generative Gaussian splatting framework, termed **UniOcean**, for unified and high-fidelity underwater 3D reconstruction. Specifically, we propose a Progressive Gaussian Generation (PGG) strategy that first optimizes coarse Gaussians with density-aware primitive masking and then generates texture Gaussians using a customized feature generator guided by generative priors. Such a paradigm is capable of jointly learning global scene appearance and local high-frequency textures, thereby delivering refined Gaussian representations. Subsequently, to uniformly model global light attenuation, we devise a Adaptive Particle Modeling (APM) module that estimates depth-dependent absorption and scattering maps to reconstruct degraded views. Such a unified formulation adaptively represents diverse particle-induced effects, providing physically informed guidance for Gaussian optimization and yielding a clear scene appearance. Extensive experiments show that our UniOcean significantly outperforms existing methods across static, dynamic, and synthetic scattering underwater scenes, as shown in Fig. 1(b).

The main contributions of this work are as follows:

- We propose UniOcean, a novel generative Gaussian splatting framework for unified underwater 3D reconstruction. To the best of our knowledge, this is the first work to address diverse underwater conditions through a unified generative paradigm.
- We propose the APM strategy that uniformly models the underwater particles using depth-dependent absorption and scattering components to adaptively handle diverse conditions.
- We devise the PGG module that employs coarse Gaussian masking and texture Gaussian generation to jointly model global appearance and local texture details.

2 Related Work

Novel View Synthesis. NeRF Mildenhall et al. (2021) has attracted widespread attention in novel view synthesis Müller et al. (2022); Chen et al. (2023); Reiser et al. (2021); Kurz et al. (2022); Liu et al. (2020); Chen et al. (2022). Despite these advances, NeRF-based methods remain computationally expensive due to the inefficient volume rendering, limiting their practical applicability. With the



(a) Time-Frequency Curve (b) Optimization Trajectory

Figure 2: (a) Time-frequency visualization and analysis. Gaussian curve exhibits a smoother temporal profile and a narrowband concentration in the low-frequency region, revealing its inherent low-frequency learning bias. (b) Previous densification often converge to suboptimal representations, whereas our method generates high-frequency texture Gaussians to achieve high-fidelity rendering.

advent of 3DGS Kerbl et al. (2023), novel view synthesis has achieved significant improvements in both rendering quality and efficiency. For example, Mip-Splatting Yu et al. (2024) and MS3DGS Yan et al. (2024) employ multi-resolution Gaussian representations to alleviate aliasing artifacts. To improve reconstruction quality under sparse-view settings, FSGS Zhu et al. (2024), DropGaussian Park et al. (2025a), and NexusGS Zheng et al. (2025) introduce dedicated regularization strategies to enhance geometric consistency. Moreover, several methods extend 3DGS with temporal modeling to support dynamic reconstruction Yang et al. (2024b); Wu et al. (2024); Park et al. (2025b); Kwak et al. (2025). Nevertheless, these 3DGS-based methods generally assume high-quality observations for stable optimization and therefore struggle under complex degraded conditions.

Novel View Synthesis in Underwater Conditions. In addition to early implicit neural fields Levy et al. (2023); Tang et al. (2024); Zhou et al. (2025); Liu et al. (2025), various 3DGS-based methods have recently been developed to explicitly represent the underwater medium. For instance, Water-Splatting Li et al. (2024) employs independent neural volumetric fields to capture scattering media. Along this line, Aquatic-GS Liu et al. (2024a) introduces a hybrid 3D representation for underwater scenes. Similarly, SeaSplat Yang & Leonard (2024) leverages an underwater image formation model to jointly learn the medium parameters and the underlying 3D representation. To improve rendering quality, RestorGS Qiao et al. (2025), PlenodiumWU et al. (2026), and 3D-UIR Yuan et al. (2026a) incorporates depth information to guide the medium modeling and jointly optimizes it with Gaussian attributes. To further improve rendering quality, 3D-USE Yuan et al. (2026b) introduces an underwater scene-level enhancement module to refine the 3D scene representation. However, the above methods struggle to uniformly model underwater particles while rendering fine-grained textures, resulting in suboptimal results. In contrast, we propose a generative Gaussian Splatting framework for unified and high-quality underwater 3D reconstruction.

3 Methodology

3.1 Problem Formulation

Neural Attenuation Function. Underwater imagery generally suffers from severe visual degradation caused by complex attenuation effects. Traditional formulations decompose the image into two primary components: direct scene radiance and backscatter light Akkaynak & Treibitz (2018):

$$I(x) = J(x) \cdot e^{-\beta^D \cdot z(x)} + B^{\infty} \cdot \left(1 - e^{-\beta^B \cdot z(x)}\right), \tag{1}$$

where $J(x)$ denotes the scene radiance, B^{∞} denotes the backscatter light at infinite distance, and β^D and β^B denote the attenuation coefficients for the direct and backscatter parts that interact with the viewing-ray distance $z(x)$, respectively. However, these variables require either empirical setting or tailored designs under different conditions, limiting their generalization. In contrast, we define a unified formulation composed of absorption $A(x)$ and scattering $S(x)$ components:

$$\underbrace{I(x)}_{\text{Degraded}} = \underbrace{J(x)}_{\text{Clear}} \odot \underbrace{A(x)}_{\text{Absorption}} + \underbrace{S(x)}_{\text{Scattering}}. \tag{2}$$

Our goal is no longer to estimate each individual physical variable, but to optimize a neural attenuation function F_θ that uniformly models the absorption and scattering effects during light propagation, inspired by the Beer-Lambert law Gordon (1989). The optimization objective can be formulated as:

$$\theta^* = \arg\min_{\theta} \mathbb{E}_{v_i \sim \mathcal{D}} \left[\mathcal{L}\left(\mathcal{I}(F_\theta, v_i),\, I_{\text{gt}}^{v_i} \right) \right], \tag{3}$$

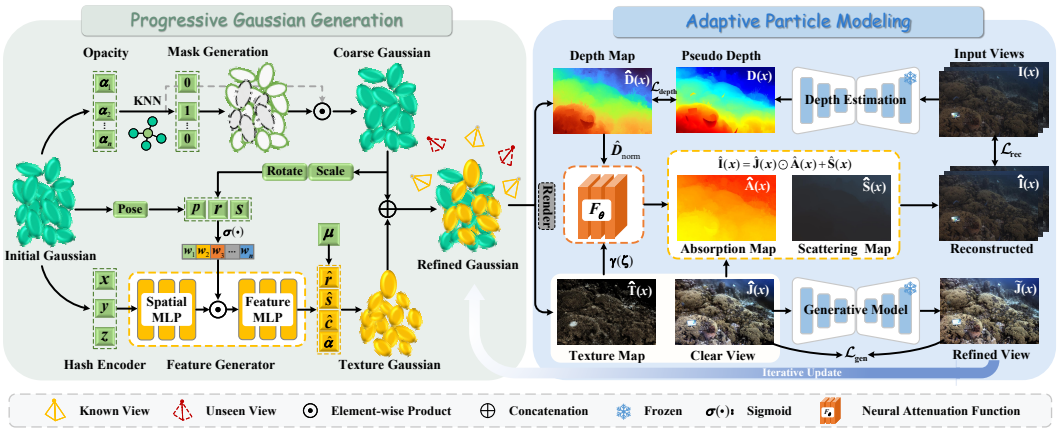


Figure 3: Overview of our UniOcean. **Left**: Progressive Gaussian Generation. The optimized coarse Gaussians are leveraged to generate fine-grained texture Gaussians guided by generative models, thereby producing refined Gaussians capturing both global appearance and local textures. **Right**: Adaptive Particle Modeling. The absorption and scattering maps are adaptively estimated via a learnable neural attenuation function, thus uniformly modeling diverse underwater particle effects. The two modules are jointly optimized for high-quality underwater 3D reconstruction.

where $\mathcal{L}$ denotes the composite loss function and θ represents the optimizable parameters. $\mathcal{D}$ denotes the set of training views, $\mathcal{I}$ denotes the reconstruction procedure for the input view defined in Eq. (2), and I_{gt}^{i} denotes the ground-truth image corresponding to view v_i.

Generative Gaussian Splatting. 3DGS Kerbl et al. (2023) constructs scene representations with a set of 3D Gaussians, each of which is parameterized by several key attributes: center position $\mu_i \in \mathbb{R}^3$, scale vector $s_i \in \mathbb{R}^3$, rotation quaternion $q_i \in \mathbb{R}^4$, spherical harmonic coefficients c_i, and opacity term o_i. Formally, the basis function of the i-th Gaussian can be expressed as:

$$G_i(x) = e^{-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)}, \tag{4}$$

where the covariance matrix Σ_i is defined by the scaling factor s_i together with the rotation quaternion q_i. The spherical harmonics of each Gaussian are decoded to produce its color c_i, whereas the α_i is computed by modulating the opacity o_i with its projected contribution on the image plane G_i'. Therefore, the final color of pixel p is obtained by blending the N Gaussians:

$$C(p) = \sum\nolimits_{i \in N} c_i \alpha_i \prod\nolimits_{j=1}^{i-1}(1-\alpha_j), \quad \alpha_i = o_i G_i'(p). \tag{5}$$

Our goal is not to merely optimize the original Gaussian primitives $\mathcal{G}_c$, but to additionally generate fine-grained texture Gaussians $\mathcal{G}_\xi$ to enrich scene details. Both the original and texture Gaussians are jointly optimized and fused for final refinement. The objective is formulated as:

$$\mathcal{G}_c^*, \xi^* = \arg\min_{\mathcal{G}_c, \xi} \mathbb{E}_{v_i \sim \mathcal{D}} \left[\mathcal{L}\left(\mathcal{I}(F_\theta, \mathcal{R}(\mathcal{G}_c \cup \mathcal{G}_\xi, v_i),\ I_{\text{gt}}^{v_i}) \right) \right] \tag{6}$$

where $\mathcal{R}$ denotes the rendering operation, and ξ represents the optimizable parameters used for Gaussian generation. Building upon the above analysis, we propose UniOcean, a generative Gaussian splatting framework for unified 3D reconstruction across diverse underwater conditions, as illustrated in Fig. 3. UniOcean consists of two key designs: a progressive Gaussian generation module and a unified particle modeling strategy. These components work collaboratively to optimize 3D Gaussians, enabling unified and high-quality underwater 3D reconstruction.

3.2 Progressive Gaussian Generation

To ensure effective Gaussian representation, we propose a Progressive Gaussian Generation (PGG) strategy that first optimizes coarse Gaussians to capture global appearance and then generates fine-grained texture Gaussians based on the optimized coarse parameters to learn local high-frequency textures. These two groups of Gaussians are finally fused to yield a refined representation. It consists of two core steps: coarse Gaussian masking and texture Gaussian generation.

Coarse Gaussian Masking. Underwater objects are inevitably occluded by participating medium, hindering the densification of Gaussians. To address this issue, we design a Gaussian masking strat-

egy that progressively selects subsets of occluding Gaussians for removal during training, allowing the remaining Gaussians to receive greater visibility for effective optimization. Specifically, Gaussians in dense regions have shorter distances to their neighbors, while those in sparse regions have larger ones. Thus, we compute a density score using K-Nearest Neighbors (KNN) in the 3D space:

$$s_\rho = -\frac{1}{K}\sum\nolimits_{k\in\mathcal{N}_K(i)} \|g_i - g_k\|_2, \tag{7}$$

where $g_i \in \mathbb{R}^3$ denotes the center of the i-th Gaussian, $\mathcal{N}_K(i)$ denotes the index set of K nearest neighbors, and $||\cdot||_2$ denotes the Euclidean distance. Then, a threshold τ is applied to the density score s_ρ to identify high-density regions, thereby forming a uncertainty mask:

$$\mathcal{M} = \mathbb{I}\big(\text{Norm}(s_\rho) > \tau\big), \quad \text{s.t.} \quad \tau \sim \mathcal{U}(0,1), \tag{8}$$

where $\mathbb{I}(\cdot)$ denotes the indicator function and $Norm(\cdot)$ denotes the normalization function. After applying this mask to the opacity values, redundant occluding Gaussians are suppressed. As the number of Gaussian primitives grows during training, we gradually increase the masking ratio over the iterations by further randomly selecting a subset of elements from $\mathcal{M}$ with ratio r for removal:

$$\tilde{\alpha_i} = \mathcal{M}_i(r) \odot \alpha_i, \quad r = r_0 \cdot (I_{cur}/I_{total}), \tag{9}$$

where $\odot$ denotes element-wise multiplication. r_0 denotes the initial masking ratio. I_{cur} and I_{total} represent the current and total numbers of training iterations, respectively.

Texture Gaussian Generation. Given optimized coarse Gaussians, we embed the camera pose and Gaussian features into a tailored feature generator to produce fine-grained texture Gaussians. Specifically, we first apply a multi-resolution hash encoding Müller et al. (2022) to the Gaussian center position μ_i to capture multi-level spatial details. The encoded spatial features is derived by:

$$F_s = f_{\xi 1}(\phi_{\text{hash}}(x_i, y_i, z_i)), \tag{10}$$

where $f_{\xi 1}$ denotes the compact spatial MLP, and (x_i, y_i, z_i) denotes the center position of the Gaussian in 3D space. Subsequently, the camera pose p_i, along with the rotation r_i and scale s_i of the coarse Gaussian, are fed into a feature MLP ($f_{\xi 2}$) to predict feature-related attention weights and adaptively modulate the spatial features. A multi-head MLP ($f_{\xi 3}$) is then employed to decode the enhanced features into new texture Gaussian attributes, formally expressed as:

$$F_h = \sigma(f_{\xi 2}([p_i, r_i, s_i])) \odot F_s, \quad \hat{r}_i, \hat{s}_i, \hat{c}_i, \hat{\alpha}_i = f_{\xi 3}(F_h), \tag{11}$$

where σ denotes the sigmoid function. Notably, rather than learning new position attributes, we directly copy the positions of coarse Gaussian as the center position of texture Gaussian, i.e., $\hat{\mu}_i = \mu_i$. This is because the coarse Gaussians already provide accurate spatial locations, enabling fine-scale detail modeling and faster convergence. Finally, the coarse Gaussians $\mathcal{G}_c^i = \{\mu_i, r_i, s_i, c_i, \tilde{\alpha}_i\}$ and texture Gaussians $\mathcal{G}_\xi^i = \{\hat{\mu}_i, \hat{r}_i, \hat{s}_i, \hat{c}_i, \alpha_i\}$ are combined to form the refined Gaussians:

$$\mathcal{G}_{\text{refined}} = \left\{\mathcal{G}_c^i \mid i \in [1, N_c]\right\} \cup \left\{\mathcal{G}_\xi^i \mid i \in [1, N_\xi]\right\}, \tag{12}$$

where the symbol $\cup$ denotes the concatenation of Gaussians, and N_c and N_ξ denote the numbers of coarse and texture Gaussians, respectively. During training, the two sets of Gaussians are jointly optimized to facilitate high-quality 3D reconstruction of clear underwater scenes.

3.3 Adaptive Particle Modeling

To uniformly handle diverse underwater conditions, we propose an Adaptive Particle Modeling (APM) strategy that adaptively models global light attenuation by estimating depth-dependent absorption and scattering components using a neural attenuation function, thereby facilitating physically-aware Gaussian representations and high-quality appearance recovery. The strategy primarily consists of two key components: global particle modeling and local texture generation.

Global Particle Modeling. Based on the refined Gaussians, we first rasterize the 3D representation into 2D images for particle modeling. Specifically, the clear view is rendered from the refined Gaussians according to Eq. (5), while the corresponding depth map is obtained through an additional rasterization branch. Different from color rendering that accumulates RGB values along camera rays, this auxiliary depth branch performs depth-aware alpha compositing over sorted Gaussians:

$$\hat{D} = \sum\nolimits_{i\in N} d_i \alpha_i \prod\nolimits_{j=1}^{i-1}(1-\alpha_j). \tag{13}$$

We then employ the pseudo-depth D from monocular depth estimation model Yang et al. (2024a) to regularize the rendered depth. Since the depths have different absolute scales across images, we normalize the depths to mitigate global scale discrepancies and stabilize training gradients:

$$\hat{D}_{\text{norm}} = 1 - \frac{\hat{D} - \hat{D}_{\text{min}}}{\hat{D}_{\text{max}} - \hat{D}_{\text{min}} + \epsilon}, \tag{14}$$

where $\hat{D}_{\text{max}}$ and $\hat{D}_{\text{min}}$ denote the maximum and minimum values of the depth map, respectively. ϵ denotes a small constant (e.g., 10^{-6}). To accurately model underwater particles, we leverage the rendered depth map $\hat{D}_{\text{norm}}(x)$ together with camera extrinsic parameters ζ encoding cross-view geometric correspondences to estimate the absorption $\hat{A}(x)$ and scattering map $\hat{S}(x)$:

$$\hat{A}(x), \hat{S}(x) = F_\theta(\hat{D}_{\text{norm}}(x), \gamma(\zeta)), \tag{15}$$

where F_θ denotes a compact convolutional neural network, and γ represents the positional encoding. With the estimated absorption and scattering components, the degraded view can be effectively reconstructed via the unified formulation in Eq. (2). Accordingly, the refined Gaussians are further optimized by minimizing the discrepancy between the reconstructed and observed degraded images.

Local Texture Generation. To address occlusions caused by underwater particles and capture fine-grained textures, we devise a local detail enhancement strategy that improves scene granularity by synthesizing unseen views and refining them using a generative model. Specifically, we first compute the average camera position t of two adjacent observed views and then apply random perturbations Δt to interpolate an unseen view. This process is formulated as:

$$\hat{J}_{\text{unseen}} = \mathcal{R}(\mathcal{G}_{\text{refined}}, P'), \quad P' = (t + \Delta t, q), \tag{16}$$

where Δt is sampled from a Gaussian distribution $\mathcal{N}(0, \delta)$. P' and q denote the updated camera position and mean rotation, respectively. A single-step generative model $\mathcal{DM}$ Wu et al. (2025a) conditioned on text prompt y_{text} is applied to restore artifacts and synthesize fine details:

$$\tilde{J}_{\text{refine}}(x) = \mathcal{DM}(\hat{J}_{\text{unseen}}(x), y_{\text{text}}). \tag{17}$$

Ultimately, the enhanced views serve as auxiliary supervision to optimize the rendered outputs, thereby feeding back to facilitate the densification of the refined Gaussian primitives.

3.4 Optimization

Reconstruction Loss. Following 3DGS Kerbl et al. (2023), we employ a photometric loss to align the reconstructed image with the degraded input. To further account for the spatially varying effects induced by underwater particles, we introduce a weighting matrix $W = 1/(\text{sg}(\hat{I}_{rec}) + \epsilon)$ to assign greater weights to regions suffering from severe optical attenuation:

$$\mathcal{L}_{\text{rec}} = (1 - \lambda)\mathcal{L}_{\text{reg-l1}}(W \odot \hat{I}_{\text{rec}}, W \odot I_{\text{gt}}) + \lambda\mathcal{L}_{\text{reg-ssim}}(W \odot \hat{I}_{\text{rec}}, W \odot I_{\text{gt}}), \tag{18}$$

where $\mathcal{L}_{\text{reg-l1}}$ and $\mathcal{L}_{\text{reg-ssim}}$ denote the regularized loss terms, λ is a weight coefficient set to 0.2, $\odot$ denotes element-wise multiplication, and $\text{sg}(\cdot)$ represents the stop-gradient operator.

Depth Regularization Loss. To faithfully model geometric structures, we leverage pseudo-depth outputs from a pre-trained monocular depth estimator Yang et al. (2024a) to regularize the synthesized pseudo-views at both global and local levels:

$$\mathcal{L}_{\text{depth}}^{\text{global}} = 1 - \text{PCC}\left(\hat{D}, D_{\text{norm}}\right), \quad \mathcal{L}_{\text{depth}}^{\text{local}} = \frac{1}{K}\sum\nolimits_{k=0}^{K-1}\left(1 - \text{PCC}\left(\hat{D}^k, D_{\text{norm}}^k\right)\right), \tag{19}$$

where PCC denotes the Pearson correlation, $\hat{D}$ and D_{norm} denote the rendered depth and pseudo-depth, respectively, and K denotes the number of local patches. Thus, $\mathcal{L}_{\text{depth}} = \mathcal{L}_{\text{depth}}^{\text{global}} + L_{\text{depth}}^{\text{local}}$.

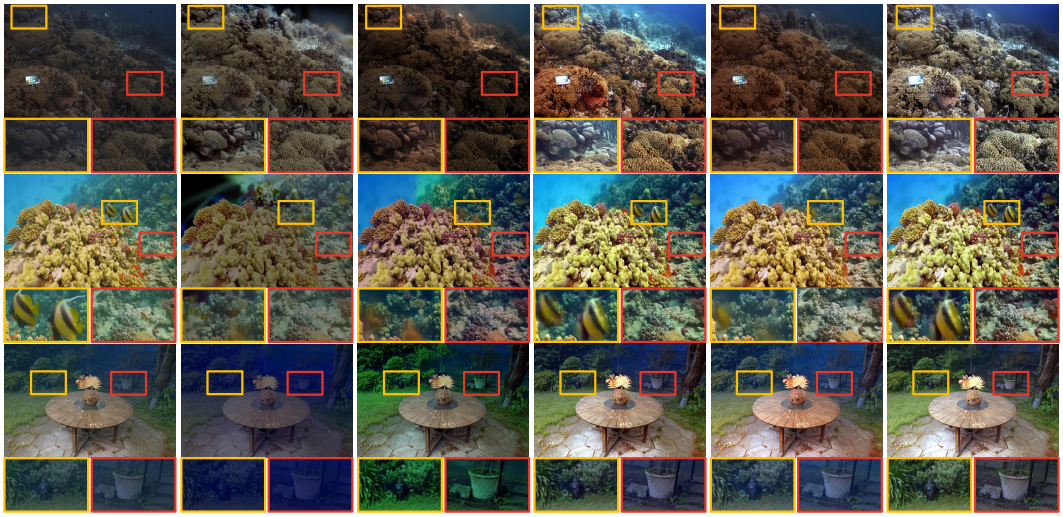
Brightness Balance Loss. To enforce consistent and physically plausible brightness under varying underwater conditions, we introduce a brightness balance loss that encourages the brightness of the rendered clear view $L_{\hat{J}}$ to approach a target level L_{target} across Φ non-overlapping local regions:

$$\mathcal{L}_{\text{bri}} = \frac{1}{\Phi}\sum\nolimits_{\phi=1}^{\Phi}|L_\phi - L_{\text{target}}|, \tag{20}$$

Generative Guidance Loss. To suppress floating artifacts and generate fine-grained textures, we incorporate a generative loss to iteratively refine the synthesized water-free views. This loss exploits

Table 1: Quantitative comparison of different methods on the SeaThru-NeRF dataset Levy et al. (2023). The best and second-best scores are marked as red and orange, respectively.

Method	Curacao			IUI3 Red Sea			Panama			J.G. Red Sea		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
ZipNeRF Barron et al. (2023)	19.95	0.442	0.421	16.93	0.474	0.412	19.01	0.349	0.482	19.02	0.349	0.483
SeaThru-NeRF Levy et al. (2023)	30.19	0.873	0.210	25.90	0.785	0.304	27.84	0.834	0.224	21.84	0.767	0.249
3DGS Kerbl et al. (2023)	28.31	0.873	0.221	22.98	0.843	0.245	29.20	0.893	0.152	21.49	0.854	0.216
SeaSplat Yang & Leonard (2024)	30.76	0.898	0.186	27.49	0.832	0.213	28.58	0.887	0.180	22.69	0.845	0.196
RecGS Zhang et al. (2024)	29.31	0.893	0.211	24.36	0.823	0.289	27.82	0.879	0.208	21.81	0.834	0.224
WaterSplatting Li et al. (2024)	32.20	0.948	0.116	29.84	0.889	0.203	31.61	0.942	0.080	24.74	0.892	0.116
Plenodium WU et al. (2026)	33.88	0.954	0.108	30.20	0.893	0.205	32.41	0.945	0.072	24.88	0.895	0.120
3D-UIR Yuan et al. (2026a)	30.97	0.907	0.187	28.30	0.841	0.252	29.81	0.900	0.181	23.37	0.857	0.187
3D-USE Yuan et al. (2026b)	33.96	0.957	0.104	30.21	0.896	0.206	32.33	0.947	0.071	24.95	0.896	0.116
UniOcean (Ours)	34.10	0.958	0.093	30.36	0.907	0.204	32.37	0.951	0.068	25.08	0.895	0.104



(a) Input (b) WaterSplatting (c) SeaSplat (d) 3D-UIR (e) 3D-USE **(f) UniOcean (Ours)**

Figure 4: Visual comparison of different methods on the SeaThru-NeRF, U-IW, and S-U datasets. Our method achieves superior visual quality on static, dynamic, and synthetic scattering scenes.

the strong image prior of a pretrained generative model to provide additional guidance:

$$\mathcal{L}_{\text{gen}} = \mathcal{L}_{\text{LPIPS}}(\hat{J}_{\text{unseen}}, \tilde{J}_{\text{refine}}). \tag{21}$$

Total Loss. During training, we jointly optimize the Gaussian primitives and model parameters to ensure faithful and physically plausible rendering. The overall objective is formulated as:

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \lambda_{\text{depth}}\mathcal{L}_{\text{depth}} + \lambda_{\text{bri}}\mathcal{L}_{\text{bri}} + \lambda_{\text{gen}}\mathcal{L}_{\text{gen}}, \tag{22}$$

where λ_{depth}, λ_{bri}, and λ_{gen} are weighting coefficients and are set to 5, 0.1, and 0.1, respectively.

4 Experiments

4.1 Experimental Settings

Datasets and Metrics. We evaluate the performance of our method on 12 scenes from three widely used benchmark datasets: SeaThru-NeRF Levy et al. (2023), U-IW Tang et al. (2024), and S-U Yuan et al. (2026a). Specifically, (1) the SeaThru-NeRF dataset consists of real underwater images captured from four marine environments, with each scene containing 18∼29 views and primarily featuring static seabed rocks. (2) The U-IW dataset is constructed from sampled frames of different dynamic underwater videos. Each scene contains 67∼100 views featuring both static and dynamic objects. (3) The S-U dataset is a synthetic scattering dataset based on four scenes from the Mip-NeRF 360 dataset Barron et al. (2022). Each scene contains 125∼240 synthetic views, where underwater effects are simulated following Levy et al. (2023). For evaluation, we conduct quantitative evaluations of the reconstructed results across various underwater scenes and provide qualitative comparisons of the restoration results to assess perceptual visual quality.

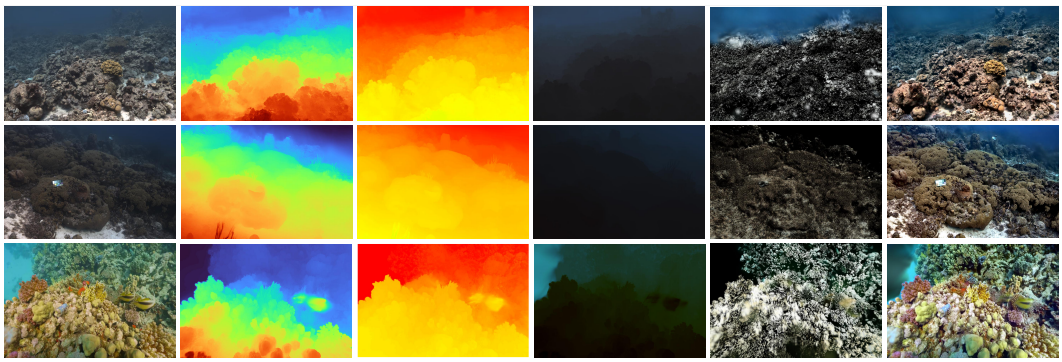
Implementation Details. All experiments are implemented using PyTorch on an NVIDIA RTX 5090 GPU. For the Progressive Gaussian Generation stage, we employ a two-layer spatial MLP to encode spatial information, which is then decoded into Gaussian features using a two-layer feature

Table 2: Quantitative comparison of different methods on the U-IW dataset Tang et al. (2024). The best and second-best scores are marked as red and orange, respectively.

Method	Composite			Coral			Sardine			Turtle		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
SeaThru-NeRF Levy et al. (2023)	23.45	0.795	0.248	25.35	0.855	0.160	13.89	0.383	0.624	13.06	0.542	0.502
3DGS Kerbl et al. (2023)	26.07	0.874	0.158	28.43	0.907	0.114	22.79	0.838	0.247	32.14	0.958	0.112
SeaSplat Yang & Leonard (2024)	25.50	0.864	0.165	27.59	0.895	0.124	22.70	0.839	0.241	32.29	0.957	0.107
RecGS Zhang et al. (2024)	22.59	0.866	0.162	25.97	0.904	0.115	15.34	0.708	0.330	24.83	0.875	0.116
WaterSplatting Li et al. (2024)	25.72	0.873	0.146	28.22	0.902	0.113	22.66	0.843	0.237	26.07	0.912	0.174
Plenodium WU et al. (2026)	25.79	0.870	0.150	28.26	0.901	0.114	23.04	0.850	0.232	31.54	0.942	0.128
3D-USE Yuan et al. (2026b)	25.80	0.871	0.148	28.35	0.904	0.117	24.63	0.861	0.223	31.68	0.946	0.123
3D-UIR Yuan et al. (2026a)	26.54	0.877	0.153	28.73	0.907	0.114	24.97	0.867	0.218	32.53	0.959	0.117
UniOcean (Ours)	27.24	0.885	0.121	29.32	0.915	0.096	25.39	0.876	0.197	32.70	0.961	0.105

Table 3: Quantitative comparison of different methods on the S-U dataset Yuan et al. (2026a).The best and second-best scores are marked as red and orange, respectively.

Method	Garden			Stump			Bicycle			Counter		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
SeaThru-NeRF Levy et al. (2023)	26.02	0.884	0.188	25.11	0.738	0.364	22.39	0.730	0.361	24.22	0.867	0.259
3DGS Kerbl et al. (2023)	32.51	0.922	0.191	25.49	0.808	0.320	28.13	0.856	0.242	30.94	0.937	0.178
SeaSplat Yang & Leonard (2024)	30.52	0.896	0.209	26.53	0.800	0.320	27.89	0.836	0.258	29.31	0.911	0.225
RecGS Zhang et al. (2024)	27.41	0.899	0.207	22.64	0.745	0.335	26.14	0.831	0.237	22.27	0.826	0.256
WaterSplatting Li et al. (2024)	31.89	0.922	0.175	29.09	0.837	0.279	28.38	0.829	0.280	30.51	0.925	0.208
Plenodium WU et al. (2026)	31.59	0.908	0.172	28.83	0.830	0.264	28.49	0.838	0.241	31.15	0.928	0.197
3D-USE Yuan et al. (2026b)	31.44	0.906	0.193	28.76	0.828	0.283	28.15	0.812	0.276	30.24	0.926	0.210
3D-UIR Yuan et al. (2026a)	33.20	0.923	0.142	28.97	0.832	0.255	30.25	0.875	0.181	32.47	0.934	0.171
UniOcean (Ours)	33.68	0.926	0.140	29.22	0.835	0.248	30.64	0.878	0.176	32.72	0.939	0.171



(a) Reconstructed (b) Depth Map (c) Absorption Map (d) Scattering Map (e) Texture Map (f) Clear

Figure 5: Physical visualization. Our method accurately estimates individual physical components from underwater views, thereby enabling effective particle modeling and Gaussian optimization.

MLP with four heads. We first optimize the coarse Gaussians for 7,000 iterations and then introduce the texture Gaussians for joint optimization. For the Adaptive Particle Modeling stage, we employ a three-layer CNN to separately estimate the absorption and scattering components. Our UniOcean is optimized for 30,000 iterations on each scene. More details are presented in the Appendix.

4.2 Comparison with SOTAs

Quantitative Result. Tables 1 and 2 show the quantitative results on different underwater scenes. For the SeaThru-NeRF dataset, our method achieves a higher average PSNR than 3D-USE and 3D-UIR, indicating superior reconstruction quality. For the U-IW dataset, despite the presence of challenging dynamic objects, our UniOcean achieves the best performance across all metrics. Additionally, Table 3 presents the quantitative scores across different synthetic underwater scenes. It can be observed that UniOcean consistently outperforms the previous best-performing method across the evaluated metrics. These results demonstrate its ability to adapt to varying underwater conditions and highlight the advantages of our unified generative framework over existing approaches.

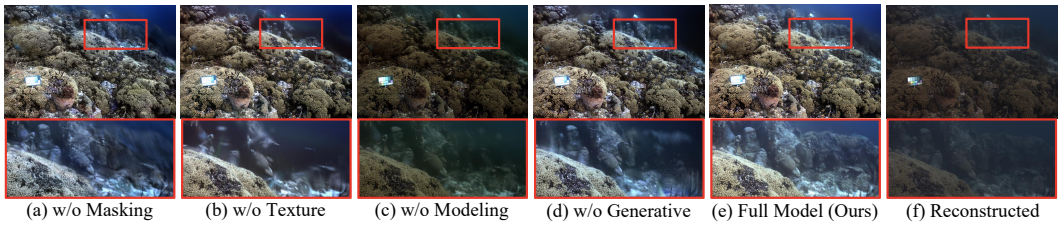
Visual Comparison. We further provide visual comparisons against competing methods. As shown in Fig. 4, our approach significantly enhances rendering quality on the SeaThru-NeRF dataset, especially in distant regions and when preserving fine-grained textural details of scene objects. For the Curacao scene, although previous methods can alleviate scattering effects caused by the underwater medium, their results typically suffer from overly dark and unrealistic appearances. In contrast, our method effectively rectifies the scene appearance via a physically-based modeling mechanism.

Table 4: Ablation results for each components on the SeaThru-NeRF dataset. The first and second best scores are marked in each cell.

Ablation	PSNR ↑	SSIM ↑	LPIPS ↓
w/o Gaussian Masking	30.16	0.923	0.121
w/o Texture Gaussian	29.97	0.921	0.125
w/o Particle Modeling	27.76	0.879	0.183
w/o Generative Guidance	30.22	0.925	0.120
Full Model (Ours)	30.48	0.928	0.117

Table 5: Ablation results of individual loss on the SeaThru-NeRF dataset. The first and second best scores are marked in each cell.

Ablation	PSNR ↑	SSIM ↑	LPIPS ↓
Reconstruction ($\mathcal{L}_{\text{rec}}$)	29.16	0.913	0.134
+ Depth_Global ($\mathcal{L}_{\text{depth}}^{\text{global}}$)	30.04	0.922	0.125
+ Depth_Local ($\mathcal{L}_{\text{depth}}^{\text{local}}$)	30.17	0.925	0.121
+ Generative ($\mathcal{L}_{\text{gen}}$)	30.26	0.927	0.119
+ Brightness ($\mathcal{L}_{\text{bri}}$)	30.48	0.928	0.117



(a) w/o Masking (b) w/o Texture (c) w/o Modeling (d) w/o Generative (e) Full Model (Ours) (f) Reconstructed

Figure 6: Visual results for each ablation study. The full model achieves superior visual quality.

Similar advantages can also be observed on the U-IW and S-U datasets. Fig. 5 visualizes the different physical components estimated by our method. As observed, the depth map clearly reflects the spatial distribution and depth hierarchy of objects within the scene. The absorption and scattering maps accurately capture the variation of light as it propagates through the underwater medium, particularly the depth-dependent attenuation and scattering effects. The texture map further represents high-frequency details that are often overlooked in original rendering processes.

4.3 Ablation Studies

Effectiveness of Individual Components. We evaluate the effectiveness of individual components of our UniOcean in Table 4 and Fig. 6. Specifically, w/o Gaussian Masking removes the masking strategy and directly optimizes the coarse Gaussians, making it difficult to fully capture the target appearance. w/o Texture Gaussian removes the Gaussian feature generator used to predict texture attributes, resulting in insufficient recovery of high-frequency scene textures. Meanwhile, w/o Particle Modeling discards the estimation of absorption and scattering maps, and instead directly learns the scene mapping using the neural network, leading to a notable performance drop. In addition, w/o Generative Guidance removes the generative model used for texture refinement and pseudo view rendering, thus yielding suboptimal results. In contrast, our full model attains superior quantitative performance and delivers better visual quality.

Impact of Each Loss Function. In Table 5, We conduct ablation studies to investigate the impact of each loss function. We start with the reconstruction loss as the baseline and progressively add each loss to evaluate its contribution to the overall performance. In particular, integrating the global depth loss brings prominent performance improvements, as depth information supplies vital geometric cues to facilitate accurate scene optimization. Additional performance gains are observed upon incorporating other loss components, validating their respective benefits.

5 Conclusion

In this paper, we propose UniOcean, a novel generative Gaussian splatting framework for unified underwater 3D reconstruction. Unlike existing methods that rely on particle-specific modeling paradigms, UniOcean develops a unified generative framework to adaptively handle diverse underwater conditions. Specifically, we propose an Adaptive Particle Modeling strategy that estimates depth-dependent absorption and scattering components to adaptively model global light attenuation, thereby uniformly modeling diverse particle-induced effects. Furthermore, we design a Progressive Gaussian Generation strategy that leverages optimized coarse Gaussians to generate fine-grained texture Gaussians guided by generative models, thereby composing refined Gaussians to capture both global appearance and local textures. Extensive experiments demonstrate that our UniOcean significantly outperforms existing methods both qualitatively and quantitatively.

REFERENCES

Derya Akkaynak and Tali Treibitz. A revised underwater image formation model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6723–6732, 2018.

Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5470–5479, 2022.

Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Zip-nerf: Anti-aliased grid-based neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19697–19705, 2023.

Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *European Conference on Computer Vision*, pp. 333–350, 2022.

Jiahao Chen, Yipeng Qin, Lingjie Liu, Jiangbo Lu, and Guanbin Li. Nerf-hugs: Improved neural radiance fields in non-static scenes using heuristics-guided segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19436–19446, 2024.

Zhiqin Chen, Thomas Funkhouser, Peter Hedman, and Andrea Tagliasacchi. Mobilenerf: Exploiting the polygon rasterization pipeline for efficient neural field rendering on mobile architectures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16569–16578, 2023.

Xiaofeng Cong, Yu Zhao, Jie Gui, Junming Hou, and Dacheng Tao. A comprehensive survey on underwater image enhancement based on deep learning. *arXiv preprint arXiv:2405.19684*, 2024.

Howard R Gordon. Can the lambert-beer law be applied to the diffuse attenuation coefficient of ocean water? *Limnology and Oceanography*, 34(8):1389–1409, 1989.

Qiuping Jiang, Yuese Gu, Chongyi Li, Runmin Cong, and Feng Shao. Underwater image enhancement quality evaluation: Benchmark dataset and objective metric. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(9):5959–5974, 2022.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):139–1, 2023.

Andreas Kurz, Thomas Neff, Zhaoyang Lv, Michael Zollhöfer, and Markus Steinberger. Adanerf: Adaptive sampling for real-time rendering of neural radiance fields. In *European Conference on Computer Vision*, pp. 254–270, 2022.

Sangwoon Kwak, Joonsoo Kim, Jun Young Jeong, Won-Sik Cheong, Jihyong Oh, and Munchurl Kim. Modec-gs: Global-to-local motion decomposition and temporal interval adjustment for compact dynamic 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11338–11348, 2025.

Minseong Kweon and Jinsun Park. Oceansplat: Object-aware gaussian splatting with trinocular view consistency for underwater scene reconstruction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 5773–5781, 2026.

Deborah Levy, Amit Peleg, Naama Pearl, Dan Rosenbaum, Derya Akkaynak, Simon Korman, and Tali Treibitz. Seathru-nerf: Neural radiance fields in scattering media. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 56–65, 2023.

Huapeng Li, Wenxuan Song, Tianao Xu, Alexandre Elsig, and Jonas Kulhanek. Watersplatting: Fast underwater 3d scene reconstruction using gaussian splatting. *arXiv preprint arXiv:2408.08206*, 2024.

Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *Advances in Neural Information Processing Systems*, 33:15651–15663, 2020.

Shaohua Liu, Junzhe Lu, Zuoya Gu, Jiajun Li, and Yue Deng. Aquatic-gs: A hybrid 3d representation for underwater scenes. *arXiv preprint arXiv:2411.00239*, 2024a.

Shuhong Liu, Lin Gu, Ziteng Cui, Xuangeng Chu, and Tatsuya Harada. I2-nerf: Learning neural radiance fields under physically-grounded media interactions. *arXiv preprint arXiv:2510.22161*, 2025.

Xiangyue Liu, Han Xue, Kunming Luo, Ping Tan, and Li Yi. Genn2n: Generative nerf2nerf translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5105–5114, 2024b.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (TOG)*, 41(4):1–15, 2022.

Hyunwoo Park, Gun Ryu, and Wonjun Kim. Dropgaussian: Structural regularization for sparse-view gaussian splatting. *arXiv preprint arXiv:2504.00773*, 2025a.

Jongmin Park, Minh-Quan Viet Bui, Juan Luis Gonzalez Bello, Jaeho Moon, Jihyong Oh, and Munchurl Kim. Splinegs: Robust motion-adaptive spline for real-time dynamic 3d gaussians from monocular video. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 26866–26875, 2025b.

Gaurav Parmar, Taesung Park, Srinivasa Narasimhan, and Jun-Yan Zhu. One-step image translation with text-to-image models. *arXiv preprint arXiv:2403.12036*, 2024.

Yuanjian Qiao, Mingwen Shao, Lingzhuang Meng, and Kai Xu. Restorgs: Depth-aware gaussian splatting for efficient 3d scene restoration. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 11177–11186, 2025.

Fitsum Reda, Janne Kontkanen, Eric Tabellion, Deqing Sun, Caroline Pantofaru, and Brian Curless. Film: Frame interpolation for large motion. In *European Conference on Computer Vision*, pp. 250–266, 2022.

Christian Reiser, Songyou Peng, Yiyi Liao, and Andreas Geiger. Kilonerf: Speeding up neural radiance fields with thousands of tiny mlps. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14335–14345, 2021.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pp. 87–103, 2024.

Yunkai Tang, Chengxuan Zhu, Renjie Wan, Chao Xu, and Boxin Shi. Neural underwater scene representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11780–11789, 2024.

Peihao Wang, Yuehao Wang, Dilin Wang, Sreyas Mohan, Zhiwen Fan, Lemeng Wu, Ruisi Cai, Yu-Ying Yeh, Zhangyang Wang, Qiang Liu, et al. Steepest descent density control for compact 3d gaussian splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 26663–26672, 2025.

Changguang WU, Jiangxin Dong, Chengjian Li, and Jinhui Tang. Plenodium: Underwater 3d scene reconstruction with plenoptic medium representation. *Advances in Neural Information Processing Systems*, 38:155502–155531, 2026.

Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20310–20320, 2024.

Jay Zhangjie Wu, Yuxuan Zhang, Haithem Turki, Xuanchi Ren, Jun Gao, Mike Zheng Shou, Sanja Fidler, Zan Gojcic, and Huan Ling. Difix3d+: Improving 3d reconstructions with single-step diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26024–26035, 2025a.

Jiang Wu, Rui Li, Yu Zhu, Rong Guo, Jinqiu Sun, and Yanning Zhang. Sparse2dgs: Geometry-prioritized gaussian splatting for surface reconstruction from sparse views. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 11307–11316, 2025b.

Tao Xie, Xi Chen, Zhen Xu, Yiman Xie, Yudong Jin, Yujun Shen, Sida Peng, Hujun Bao, and Xiaowei Zhou. Envgs: Modeling view-dependent appearance with environment gaussian. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5742–5751, 2025.

Zhiwen Yan, Weng Fei Low, Yu Chen, and Gim Hee Lee. Multi-scale 3d gaussian splatting for anti-aliased rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20923–20931, 2024.

Daniel Yang and John Leonard. Seasplat: Representing underwater scenes with 3d gaussian splatting and a physically grounded image formation model. *arXiv preprint arXiv:2409.17345*, 2024.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37:21875–21911, 2024a.

Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20331–20341, 2024b.

Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19447–19456, 2024.

Jieyu Yuan, Yujun Li, Yuanlin Zhang, Chunle Guo, Xiongxin Tang, Ruixing Wang, and Chongyi Li. 3d-uir: 3d gaussian for underwater 3d scene reconstruction via physics-based appearance-medium decouplin. *arXiv preprint arXiv:2505.21238*, 2025.

Jieyu Yuan, Yujun Li, Yuanlin Zhang, Chunle Guo, Xiongxin Tang, Ruixing Wang, and Chongyi Li. 3d-uir: 3d gaussian for underwater 3d scene reconstruction via physics-based appearance–medium decoupling. *IEEE Transactions on Image Processing*, 35:5452–5465, 2026a.

Jieyu Yuan, Yuanlin Zhang, Jihong Li, Chunle Guo, Huimin Lu, and Chongyi Li. 3d-use: From image-level to scene-level underwater enhancement. *arXiv preprint arXiv:2608.28020*, 2026b.

Tianyi Zhang, Weiming Zhi, Braden Meyers, Nelson Durrant, Kaining Huang, Joshua Mangelson, Corina Barbalata, and Matthew Johnson-Roberson. Recgs: Removing water caustic with recurrent gaussian splatting. *IEEE Robotics and Automation Letters*, 2024.

Yulong Zheng, Zicheng Jiang, Shengfeng He, Yandu Sun, Junyu Dong, Huaidong Zhang, and Yong Du. Nexusgs: Sparse view synthesis with epipolar depth priors in 3d gaussian splatting. *arXiv preprint arXiv:2503.18794*, 2025.

Jingchun Zhou, Tianyu Liang, Dehuan Zhang, Siyuan Liu, Junsheng Wang, and Edmond Q Wu. Waterhe-nerf: Water-ray matching neural radiance fields for underwater scene reconstruction. *Information Fusion*, 115:102770, 2025.

Zehao Zhu, Zhiwen Fan, Yifan Jiang, and Zhangyang Wang. Fsgs: Real-time few-shot view synthesis using gaussian splatting. In *European Conference on Computer Vision*, pp. 145–163, 2024.

A APPENDIX

A.1 ADDITIONAL DETAILS

Implementation Details. We provide additional implementation details of the proposed UniOcean. Specifically, following the standard 3DGS framework, we implement depth rendering as a dedicated second pass, where the pixel-wise depth is computed by weighted summation of the depth values of all 3D Gaussians projected onto each pixel. The Gaussian densification and pruning are performed every 100 iterations via adaptive density control. We employ Depth Anything V2 Yang et al. (2024a) to estimate pseudo-depth maps. Meanwhile, we adopt a fine-tuned one-step diffusion model from DIFIX3D+ Wu et al. (2025a) to refine the rendered views, as it effectively alleviates visual artifacts caused by under-reconstruction while enhancing high-frequency details. We provide the detailed training procedure of UniOcean in Algorithm 1.

Algorithm 1 Detailed training procedure of UniOcean

Input: $I(x)$: input underwater image; $\mathcal{G}_I$: initialized Gaussian parameter $\{\mu_i, \Sigma_i, \alpha_i, c_i\}$; ζ: camera extrinsic parameters; $f_{\xi 1}$: spatial MLP; $f_{\xi 2}$: feature MLP; $f_{\xi 3}$: decode MLP; F_θ: neural attenuation function; r_0: initial masking;

Output: $\hat{J}(x)$: rendered chear image; $\hat{I}(x)$: reconstructed image; θ_{UniOcean}: trained model;

Initialization parameters: λ_{depth} = 5; λ_{bri} = 0.1; λ_{gen} = 0.1; I_{total} = 30,000;
for $i = I_0$ to I_{total} **do**;
    Progressive Gaussian Generation:
    $r \leftarrow r_0 \cdot (I_{cur}/I_{total})$;
    $\tilde{\alpha_i} \leftarrow M_i(r) \odot \alpha_i$;
    $\mathcal{G}_c \leftarrow \{\mu_i, r_i, s_i, c_i, \tilde{\alpha_i}\}$; ▷ Coarse Gaussian
    $F_s \leftarrow f_{\xi 1}(\phi_{hash}(x_i, y_i, z_i))$;
    $F_h \leftarrow \sigma(f_{\xi 2}(Concat[p_i, r_i, s_i])) \odot F_s$;
    $\hat{r}_i, \hat{s}_i, \hat{c}_i, \hat{\alpha}_i \leftarrow f_{\xi 3}(F_h)$;
    $\mathcal{G}_\xi \leftarrow \{\hat{\mu}_i, \hat{r}_i, \hat{s}_i, \hat{c}_i, \hat{\alpha}_i\}$; ▷ Texture Gaussian
    $\mathcal{G}_{refined} = \mathcal{G}_c \cup \mathcal{G}_\xi$; ▷ Refined Gaussian
    Physical Appearance Modeling:
    $\hat{J}(x), \hat{D}(x) \leftarrow render(\mathcal{G}_{refined})$;
    $\hat{A}(x), \hat{S}(x) \leftarrow F_\theta(Norm\hat{D}(x), \gamma(\zeta))$; ▷ Physical Estimation
    $\hat{I}(x) \leftarrow \hat{J}(x) \cdot \hat{A}(x) + \hat{S}(x)$; ▷ Unified modeling
end for
$\mathcal{L}_{total} \leftarrow \mathcal{L}_{rec} + \lambda_{depth}\mathcal{L}_{depth} + \lambda_{bri}\mathcal{L}_{bri} + \lambda_{gen}\mathcal{L}_{gen}$;
Optimize the UniOcean with $\mathcal{L}_{total}$;
return θ_{UniOcean}.

Detailed Architecture of CNN. Furthermore, we present the detailed architecture of the designed CNN for estimating the absorption and scattering physical components, as shown in Fig. 7. Specifically, the network consists of two branches. In the absorption branch, the camera poses, appearance embeddings from degraded views, and the rendered depth map are first concatenated and then calibrated by a 1×1 convolution layer. The calibrated features are subsequently divided into three channel groups, each undergoing adaptive weighting and feature learning to capture complementary representations. These processed features are finally aggregated to produce the absorption map. In parallel, the scattering branch leverages the depth map to predict two scattering coefficients, which are further refined through modulation with learnable weights. The resulting components are then effectively fused to generate the final scattering map, thereby enabling a more accurate modeling of the underlying scattering effects.

Details of the Generative Model. The detailed architecture of the generative model is presented in Fig. 8. Given a rendered novel view that may contain artifacts introduced by the 3D representation, the model aims to produce a refined novel view with improved visual quality. The framework is built upon SD-Turbo Sauer et al. (2024), a single-step diffusion model well-suited for efficient image-to-image translation, thereby enabling real-time processing. The model is fine-tuned following a strategy similar to Pix2Pix-Turbo Parmar et al. (2024), where the variational autoencoder encoder

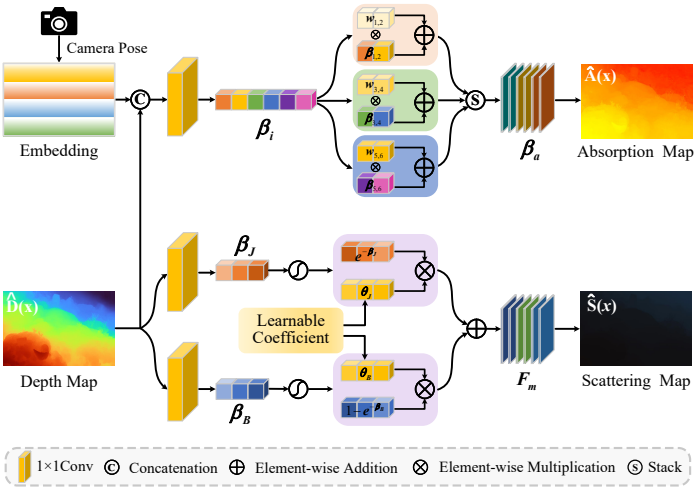


Figure 7: Detailed architecture of the designed neural attenuation function. It mainly consists of two branches: one branch takes the depth map and the camera extrinsics to estimate the absorption map, while the other branch models the scattering map using the depth map together with learnable parameters. Through iterative optimization, the model yields physically plausible distributions.

is kept frozen and the decoder is adapted using LoRA. Instead of sampling from random Gaussian noise, the network directly takes the degraded rendered image as input with a reduced noise level, which facilitates more stable and efficient training. Supervision is provided through a combination of 2D losses, including a reconstruction loss computed as the L2 distance between the predicted output and the corresponding ground-truth image, a perceptual LPIPS loss, and a style loss that encourages sharper and more detailed results. The style loss is implemented via a Gram matrix formulation, defined as the L2 norm of the auto-correlation features extracted from VGG-16 Reda et al. (2022), promoting high-frequency detail preservation and overall visual fidelity:

$$L_{\text{Gram}} = \frac{1}{L}\sum_{l=1}^{L}\beta_l \left\| G_l(\hat{I}) - G_l(I) \right\|_2, \tag{23}$$

where $\hat{I}$ and I denote the rendered view and the ground-truth view, respectively. Meanwhile, G_l denotes the Gram matrix computed from the l-th convolutional layer feature map to characterize the texture statistics. The scalar β_l serves as a weighting factor for the l-th layer, allowing the loss to balance the contributions of multi-scale features.

Details of Hash Encoding. As shown in Fig. 9, we adopt a multi-resolution hash encoding strategy to enhance the spatial representation capability of Gaussian primitives. Specifically, the 3D coordinates of each Gaussian are encoded into hierarchical feature embeddings across multiple resolutions, where coarse levels capture global spatial structures and fine levels preserve high-frequency local details. The encoded multi-scale features are concatenated and fed into the subsequent MLPs to predict Gaussian attributes, enabling more expressive modeling of complex spatial variations. The grid voxel resolution N_l at each level l can be formulated as:

$$N_l = \lfloor N_{\min} \cdot b^l \rfloor, \tag{24}$$

$$b = \exp\left(\frac{\ln N_{\max} - \ln N_{\min}}{L-1}\right), \tag{25}$$

where N_{max} and N_{min} denotes the finest and coarsest resolutions, respectively. L represents the total number of levels. For an input coordinate p, a spatial hash function is employed to map each grid position to its corresponding entry in the hash table, thereby enabling efficient retrieval of the associated voxel value. This approach allows for compact storage of voxel data and significantly reduces memory overhead, while preserving fast access for high-resolution 3D representations. Formally, the retrieved voxel value can be expressed as:

$$h(p) = \left(\oplus_{i=1}^{D} p_i \pi_i\right) \mod T, \tag{26}$$

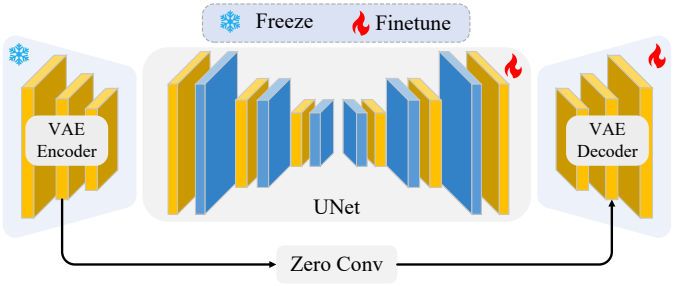


Figure 8: Detailed architecture of the generative model. Given a rendered image degraded by artifacts, the generator aims to produce an enhanced output with improved visual quality. Built upon a U-Net backbone, the network incorporates a cross-view mixing layer to preserve multi-view consistency. The model is fine-tuned from SD-Turbo Sauer et al. (2024), with a frozen VAE encoder and a LoRA-adapted VAE decoder.

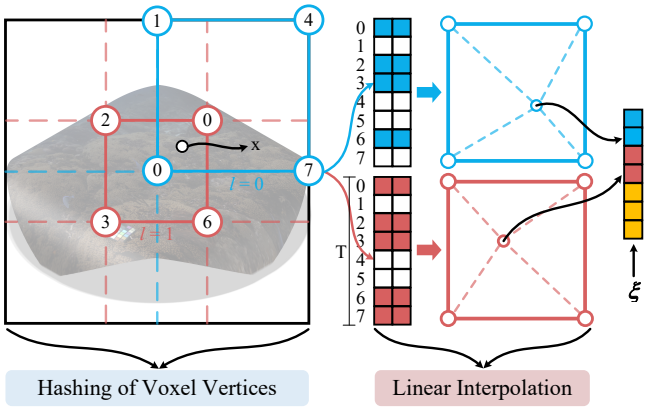


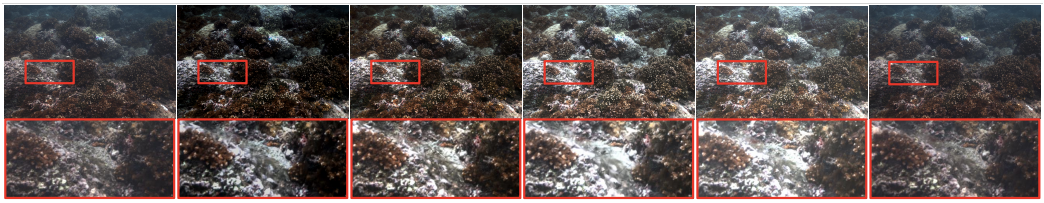
Figure 9: Illustration of multi-resolution hash encoding. The input coordinates are mapped to multiple hash tables at different resolutions to capture both coarse and fine-grained spatial features. The encoded features from different levels are then concatenated and fed into a neural network, enabling effective representation of complex spatial structures.

where D and T denote the input dimension and the size of the hash table, respectively. $\oplus$ represents the bit-wise XOR operation, and π_i are dimension-wise large prime numbers.

A.2 Additional Experimental Results

Effects of Different Exposure Values. As illustrated in Table 6 and Fig. 10, with the exposure value increases progressively, the rendered clear views exhibit a rise trend in overall brightness. This behavior primarily stems from the global modulation effect of the exposure parameter on the pixel intensity distribution. Under low-exposure conditions, the overall image response is limited, and the signal-to-noise ratio in dark regions remains low, causing structural and textural details to be severely suppressed. As the exposure value increases, the proposed method becomes more responsive to incident light, effectively enhancing the visibility of low-intensity regions and progressively revealing underlying geometric structures as well as high-frequency texture details. Furthermore, a moderate increase in exposure can partially compensate for depth-dependent energy attenuation caused by medium absorption and scattering, thereby improving global contrast and perceptual clarity. However, excessively high exposure values may lead to overexposure in bright regions, resulting in information compression and loss of fine details, and potentially degrading the fidelity of structural representation. Therefore, in practical rendering pipelines, it is essential to carefully balance brightness enhancement, contrast preservation, and detail fidelity to achieve an optimal trade-off between perceptual quality and physical consistency. In our experiments, we empirically set the exposure value to approximately 0.4 to adjust the overall brightness of the corrected results.

Ablation of Other Components. We further conduct ablation studies on other components, As shown in Table 7. Specifically, $w/$ Traditional Modeling denotes that we adopt the conventional



(a) Input (b) E = 0.2 (c) E = 0.4 (d) E = 0.6 (e) E = 0.8 (f) Reconstructed

Figure 10: Visual comparison of different exposure values.

Table 6: Quantitative comparison of different exposure values on the SeaThru-NeRF dataset Levy et al. (2023). The first and second best scores are marked in each cell.

No.	Exposure Values	PSNR↑	SSIM↑	LPIPS↓
(a)	E = 0.2	29.52	0.887	0.129
(b)	E = 0.4	30.48	0.928	0.117
(c)	E = 0.6	30.11	0.926	0.116
(d)	E = 0.8	28.26	0.864	0.138

Table 7: Ablation of other components. The first and second best scores are marked in each cell.

No.	Ablation	PSNR↑	SSIM↑	LPIPS↓
(a)	$w/$ Traditional Modeling	29.68	0.917	0.134
(b)	$w/$ More Gaussians	30.06	0.922	0.125
(c)	$w/$ 3DGS + Diffusion	30.24	0.924	0.120
(d)	Ours	30.48	0.928	0.117

Table 8: Quantitative comparison of model efficiency. Our method achieves competitive results and strikes a better balance between model efficiency and rendering quality.

Methods	Training Time↓	FPS↑	Memory (GB)↓
SeaThru-NeRF	3.5 h	0.4	15.1
3DGS	22 m	150.1	3.7
SeaSplat	54 m	80	6.1
WaterSplatting	12 m	78	5.9
3D-UIR	35 m	83	8.1
UniOcean (Ours)	48 m	72	8.8

underwater imaging model to model underwater particles. However, the parameters in such models typically need to be manually specified, making them difficult to adapt to diverse underwater scenes. $w/$ More Gaussians denotes directly splitting each Gaussian into additional Gaussians to increase the total number of Gaussians. simply increasing the number of Gaussians does not improve performance, since a larger number of Gaussians does not necessarily enhance their capacity to represent high-frequency textures. Finally, $w/$ 3DGS + Diffusion denotes using a diffusion model as a post-processing step after 3DGS reconstruction. Since the diffusion model operates independently on each rendered view, the generated details cannot be fed back to the underlying 3D representation, which can easily lead to inconsistencies across different views.

Model Efficiency. We present a comparison of model efficiency across different methods in Table 8. All experiments are conducted on a single NVIDIA RTX 5090 GPU to ensure a fair and consistent evaluation setting. The results show that vanilla 3DGS achieves the best performance in terms of training time, rendering speed (FPS), and storage consumption, demonstrating its clear advantage in computational efficiency. However, it lacks explicit modeling of the underwater imaging process and thus fails to effectively capture the global light attenuation induced by absorption and scattering effects, which limits its performance in complex underwater environments. In contrast, our UniOcean achieves highly competitive efficiency while uniformly adapting to diverse underwater conditions. Therefore, UniOcean exhibits clear advantages by jointly considering both rendering quality and computational overhead, striking a better balance between rendering accuracy and efficiency.

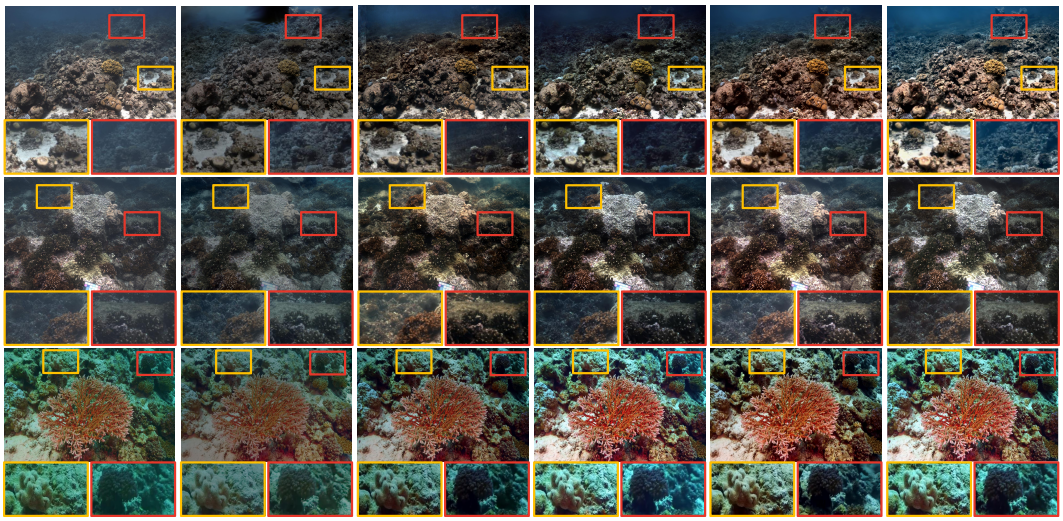


Figure 11: visual comparisons of different methods across diverse underwater scenes.

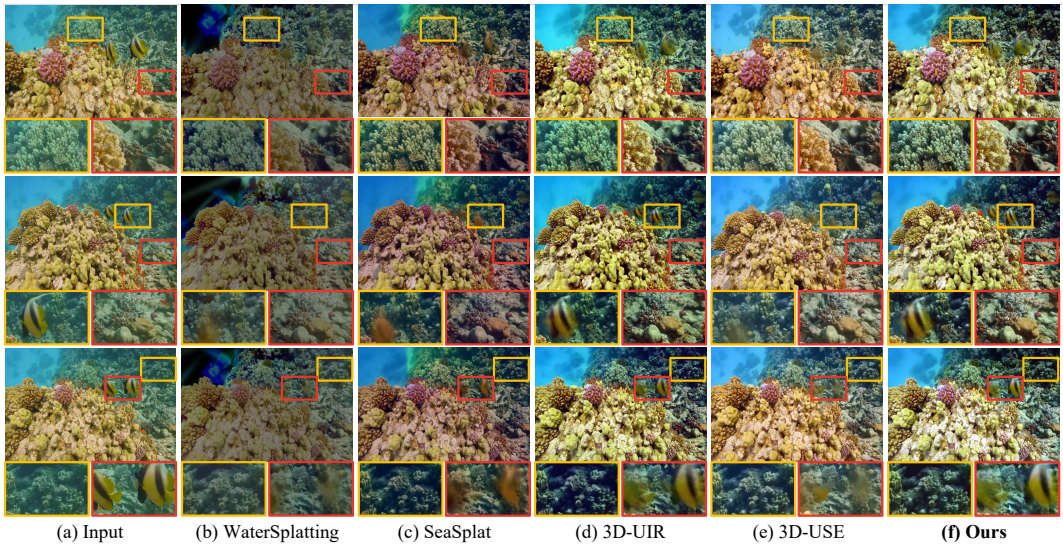


Figure 12: Visual comparison of different methods on the composite scene.

Visual Comparison. Fig. 11 and Fig. 12 present more comprehensive visual comparisons of different methods. As can be observed, WaterSplatting Li et al. (2024) and SeaSplat Yang & Leonard (2024) exhibit noticeable deficiencies in brightness correction, often resulting in inaccurate illumination estimation and low contrast, which hinders the clear representation of scene details. Although 3D-UIR Yuan et al. (2025) achieves relatively satisfactory brightness restoration, it struggles to handle color distortions in distant regions caused by medium scattering. Meanwhile, 3D-USE tends to suffer from color shift and inconsistent dynamic modeling, which yields unnatural appearances and missing dynamic objects in the reconstructed scenes. Similar observations can also be found on the MipNeRF-360 dataset Barron et al. (2022). As illustrated in the garden scene in Fig. 13, existing methods either fail to effectively remove the bluish medium caused by scattering effects or cannot properly correct the overall brightness, thereby limiting the final visual performance. In contrast, our method demonstrates stronger robustness across different datasets and diverse scenes, achieving effective medium removal along with accurate brightness and color correction, and thus producing more natural and visually consistent rendering results.

Physical Visualization. In Fig. 14, we provide additional visualizations of decoupled physical components in underwater scenes, including depth maps, absorption maps, scattering maps, tex-



Figure 13: Visual comparison of different methods on the garden scene.

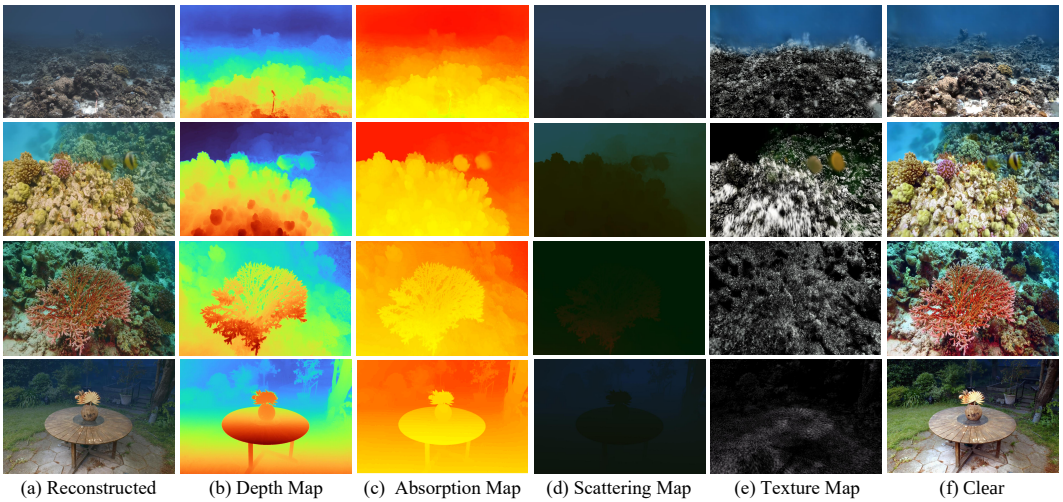


Figure 14: More visualizations of estimated physical components across diverse underwater scenes.

ture maps, and the latent water-free clear component. From these results, it can be observed that each component exhibits depth-dependent and physically plausible distributions. For instance, absorption and attenuation effects become more pronounced with increasing depth, while the texture component primarily captures the intrinsic appearance of the scene. Moreover, the recovered clear component effectively removes the influence of the underwater medium while preserving structural details and color consistency. These observations demonstrate that our method can effectively decouple the underlying physical components in underwater scenes and enable interpretable visual analysis, thereby providing valuable insights into the complex underwater imaging process.

A.3 Additional Discussion

Limitations and Future Work. We further discuss the limitations of our method. In scenarios with severely limited visibility, where the water medium is highly turbid or light penetration is minimal, the model often struggles to accurately disentangle medium-induced effects from the underlying scene, resulting in incomplete restoration or biased appearance estimates. Similarly, in regions subject to pronounced color distortion due to strong scattering or spatially heterogeneous absorption, the recovered colors and brightness may deviate from the true scene values, leading to perceptual inconsistencies and local artifacts. These failure cases underscore the limitations of the current

physical modeling formulation and optimization strategy when confronted with highly challenging underwater environments. They also motivate several promising directions for future work, such as incorporating more robust priors to handle highly turbid media, leveraging temporal consistency for dynamic scene reconstruction, and exploring hybrid approaches that combine LLM learning with physics-based modeling to further enhance visual fidelity under extreme conditions.

Discussion on Diffusion Hallucination. Diffusion models may introduce hallucinated details when restoring severely degraded regions, particularly when the original visual evidence is insufficient. To mitigate this issue, we do not treat the diffusion-refined views as absolute pixel-level supervision. Instead, we employ a perceptual LPIPS loss to provide a soft constraint on the generated views. This allows the diffusion prior to primarily guide the recovery of perceptually meaningful high-frequency textures, while avoiding excessive enforcement of potentially hallucinated pixel-level details. Meanwhile, the generated views are jointly optimized with the underlying 3D Gaussian representation, such that the recovered details are constrained by the scene geometry and multi-view observations. Therefore, the diffusion model serves as a perceptual prior for enhancing missing textures rather than directly determining the final scene appearance.

Social Impact. The proposed UniOcean exerts a significant positive social impact by effectively enhancing the visualization and rendering performance of 3D scenes in complex underwater environments. Beyond its technical merits, this framework holds great potential for practical applications in socially and environmentally relevant fields, such as underwater agent navigation and intelligent marine exploration, where reliable visual perception under underwater media is critical to operational safety and efficiency. By improving scene clarity and structural fidelity in these underwater environments, UniOcean directly contributes to enhanced operational safety, advanced assistive intelligence, and more accurate decision-making in marine-related activities, which in turn supports the sustainable development of marine industries and the protection of marine ecosystems. Notably, we also recognize the importance of responsible technology deployment, particularly in privacy-sensitive scenarios and marine protected areas. We therefore advocate for the integration of ethical guidelines into the application of UniOcean, ensuring that this technology is utilized in a manner that is beneficial, equitable, and respectful of both human interests and the natural environment across diverse social and ecological settings.

A.4 LLM USAGE STATEMENT

In this work, a LLM was only adopted to assist with linguistic revisions. Its usage was limited to grammatical correction, enhancement of text fluency, and unification of writing style. Notably, we did not rely on the LLM to produce new research concepts, perform data analysis, or create any of the substantive scientific content. All study design, code implementation, experimental results and conclusions belong exclusively to the original contributions of the authors.