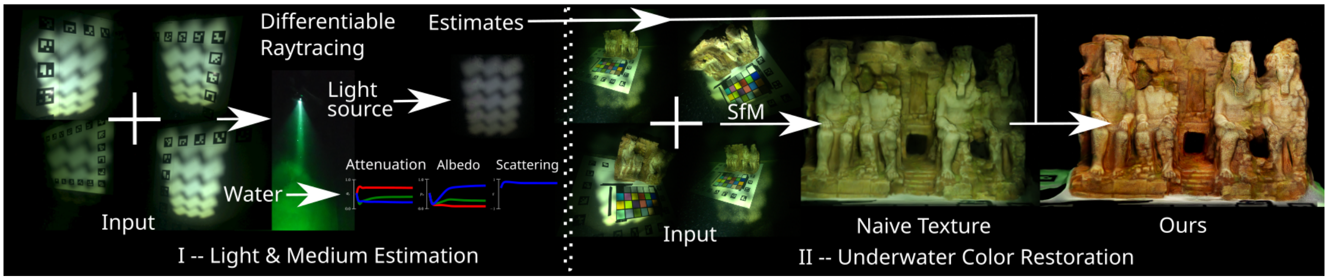
In-Situ Joint Light and Medium Estimation for Underwater Color Restoration

David Nakath Mengkun She Yifan Song Kevin Köser
GEOMAR Helmholtz Centre for Ocean Research Kiel
dnakath@geomar.de



Abstract

The majority of Earth's surface is situated in the deep sea and thus remains deprived of natural light. Such adverse underwater environments have to be explored with powerful camera-light systems. In order to restore the colors in images taken by such systems, we need to jointly estimate physically-meaningful optical parameters of the light as well as the water column. We thus propose an integrated in-situ estimation approach and a complementary surface texture recovery strategy, which also removes shadows as a by-product. As we operate in a scattering medium under inhomogeneous lighting conditions, the volumetric effects are difficult to capture in closed-form solutions. Hence, we leverage the latest progress in Monte Carlo-based differentiable ray tracing that becomes tractable through recent GPU RTX-hardware acceleration. Evaluations on synthetic data and in a water tank show that we can estimate physically meaningful parameters, which enables color restoration. The approaches could also be employed to other camera-light systems (AUV, robot, car, endoscope) operating either in the dark, in fog – or – underwater.

1. Introduction

More than half of Earth's surface is covered by the deep ocean[12]. No sunlight reaches the waters below 200m depth or the seafloor underneath. This rather adversarial visual environment is – in addition to the lack of natural illumination – governed by volumetric attenuation and scattering effects that depend on the local water composition. To explore this – yet mainly uncharted – area, we need to employ powerful camera-light systems and corresponding calibration procedures to extract physically-meaningful information from the data captured in a scattering medium. Here, image based-methods for water property estimation can save the deployment of delicate and costly instruments. Properly estimated optical water parameters can be used to infer physical or chemical ocean properties, provide information about processes or the state of the ocean (e.g. essential ocean variables[26]), or to directly analyse or detect events like algal bloom[11] or sediment suspension[37] from eddies. Significantly different from imaging with isotropic illumination, actual light sources exhibit an angular characteristic depending on the illuminant, the shape of its reflector and refraction at light housing interfaces to the water (see Fig. 2). This even holds for natural illumination 1-2m below the surface but especially for exploration in the dark with artificial light sources co-moving with the observer. Exact knowledge of the light, the medium prop-

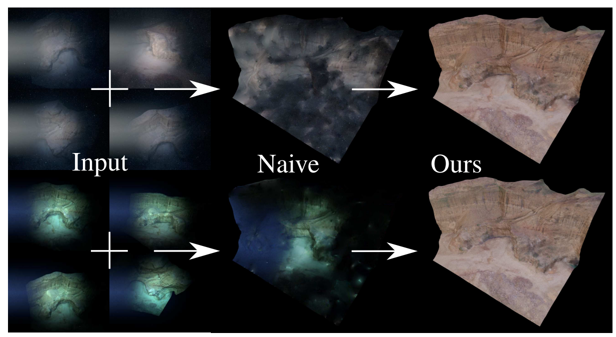


Figure 1. Underwater color restoration. The two rows are captured under different light and medium conditions, the corrected models are comparable, which is beneficial for monitoring operations.

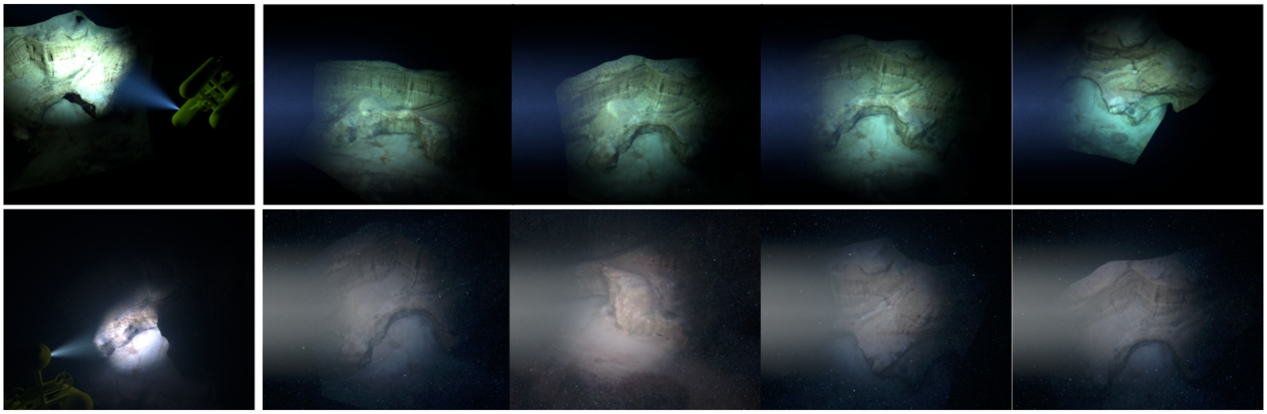


Figure 2. **Left**: Envisioned application and **right**: resulting views through the camera with different poses. The images are taken with different light/water parameters per row of the *same* 3D model. The latter has been published by GSXNet under a CC BY-NC 4.0–license.

erties, and the camera's response function would allow for a physically faithful recovery of the colors of objects with known distance, like shown in the teaser and Fig. 1. Dynamic illumination occurs in the deep sea scenario or when divers use torch lights at night or in caves, but also in endoscopic imaging, when driving at night or when robots inspect disasters, installations or tunnels with co-moving light sources. All these scenarios suffer from varying appearance, making faithful or even only consistent large scale mapping difficult. Cameras are well-suited for estimating the angular characteristics and relative orientation of lights, but underwater these parameters intertwine with volumetric scattering and attenuation which makes closed form solutions challenging. Instead, we leverage recent breakthroughs in differentiable raytracing and GPU acceleration, which allows to obtain the desired parameters conceptually similar to training a neural network, i.e. by minimizing a loss function until raytraced images look like the captured real images. While this has been used previously in computer graphics and for simulations, we apply this method directly to a real light source and water.

In this work we strive to integrate the aforementioned achievements ranging over different fields into the following contributions: Specifically, we provide **(I)** a differentiable non-parametric light model, capable of capturing arbitrary point light shapes (and, implicitly, light orientations), **(II)** a fully differentiable piece of water, which can be optimized for its attenuation and scattering properties in order to facilitate **(III)** a color-restoration approach, which removes shadows on 3D models. The latter is also used as a verification step for the estimation of the former two models' parameters. The approaches were tested in a synthetic scenario with known ground truth and in a water tank.

This work is the first step of showcasing differential raytracing underwater, we believe it opens up a new way of underwater computer vision, where (volumetric) light effects are currently blocking many vision applications.

2. Related Work

Differentiable Rendering Recently, there has been a lot of progress in differentiable rendering research, see e.g. [22] for a comprehensive overview. This spawned several applications like self-supervision for monocular 3D object detection[5]. In [16], a differentiable renderer is employed – and even learned – to predict geometric correspondence fields to refine pose estimates of 3D objects. These approaches use rasterized rendering-schemes tightly integrated into a neural network. However, in the underwater domain such an approach is not sufficient to deal with the dominant effects like attenuation and scattering.

Ray tracing, which recently matured significantly in the theoretical [36] as well as in the technical domain (e.g. through NVIDIA's OptiX leveraging the RTX-features of the GPUs), is better suited for volumes. In addition, the first differentiable approach has recently been proposed by [24], who rely on edge sampling to handle visibility changes. [9] presented a differentiable version of Mitsuba, used to train so-called inverse transport networks. They can be used to find initial parameter values for a scattering medium, which are subsequently refined through a direct application of the differentiable renderer. However, the evaluations were only conducted on artificial scenes. Recently, Mitsuba2 [30] has enabled differentiable ray tracing by leveraging an underlying jet-based auto-differentiation structure based on Enoki [20]. In this paper, we investigate differentiable ray tracing directly applied to real images using a pixel-wise photometric error function, which – to the best knowledge of the authors – has never been attempted before.

Approaches to light and medium property estimation Macroscopic models to underwater lighting were proposed by McGlamery[25] and Jaffe[19]. Based on Preisendorfers work [38], Mobley[27] discusses light propagation at the particle level, but the full physical model is difficult to invert and computationally expensive, for instance, in real ocean waters light is not scattered isotropically, but the volume scattering function depends on (and reveals!) the local composition of the water [35]. For "more uniform" illumination scenarios such as fog, Nayar and Narasimhan[29] have proposed a much simpler model that then became popular also for shallow water applications. An essential weakness, when using RGB images, is that absorption and scattering coefficients in water[3] vary strongly with wavelength (also inside the color channels of images), while they less do so in fog[1], making it difficult to obtain a single attenuation coefficient (e.g. for all "red" parts). Akkaynak and Treibitz[2] have addressed some weaknesses of the fog model by a modified backscatter estimation and an extra step for considering ambient light, which however does not account for artificial light cones and therefore holds only for shallow water. In addition, both models are tailored to horizontal imaging, where the Radiative Transfer Equation has a simple solution, whereas raytracing can cope with arbitrary directions/angles of light propagation. [42] synthesizes images based on the direct and the backscatter component of a signal in a scattering medium to infer the next-best underwater view based on and information gain criterion, i.e., differential entropy.

Bryson et al.[6] work in the artificial light scenario, and have used a pre-calibrated Gaussian light source model to correct for inhomogeneous illumination, an approach also later taken by [45]. Because of computational complexity all above approaches significantly simplify or omit the complicated volumetric scattering effects. In contrast, by employing differentiable raytracing, we will show that we can even estimate the anisotropic shape of the volume scattering. Until now, directly measuring the phase function in images required a highly elaborated setup [15], the medium to be diluted, such that only a single scattering event is likely [28], or has only been applied to synthetic imagery [9].

In terms of texture restoration, many useful image enhancement approaches such as using a "gray world" world assumption [7], but also learning and other statistical methods have been proposed. However, as argued in [2], such enhancement approaches heavily depend on assumptions and data and do not reveal the physical properties. As refraction at the light source housing interface changes the light cone (similar as for cameras) in air and in water, we aim for a complete underwater solution. We could also apply our approach in air, to infer some start values for the light characteristics in air. In this case, outside the underwater world, Park et al.[32] have used a board to calibrate a light source, which also inspired our underwater approach. However, we use a non-parameteric representation and therefore we can generalize to non-symmetric point-lights with arbitrary patterns and colors. Also in air, Azinovic et al.[4] infer light source positions in a rasterization based inverse rendering scheme, which however does not extend to volumetric effects such as attenuation and scattering. Also the general concept of analysis-through-synthesis, or inverse rendering, for obtaining optical parameters has been proposed before in air using rasterization techniques and several simulation studies have suggested that raytracing-based solutions could be useful for volumetric parameters as well [10, 34, 31].

3. Approach

It is relatively straight-forward to measure the distance and rough directions of each light wrt. a rigidly-coupled camera, but obtaining the detailed light orientation and the specific underwater light pattern is challenging. Consequently, we initially seek to estimate the light and medium parameters in a calibration scenario (see teaser image, **left**). We assume to observe a known calibration board, with a rigidly-coupled camera-light system but no other illumination, e.g. from the Sun (See Fig. 3). Camera, light and board reside in a world system (which coincides with the camera coordinate system), for which we use T_{c2w}, T_{l2w} and T_{b2w} as the respective (pose) transformations. We assume that all light rays from the light source intersect in one point (point light), and that the distance to the camera center is known. The radiant intensity distribution (RID) of the light source can be arbitrary, but for practical reasons we will assume it is confined to less than a half sphere and pointed roughly in the viewing volume of the camera. The camera is assumed to be a geometrically calibrated camera with linear (or known) radiometric response. For our experiments we use a carefully centered dome port[41, 40] to avoid refraction, but this is just for convenience and no limitation of the method.[1] For the actual calibration, a white board with AruCo markers [14] is then presented in different distances and orientations and its pose T_{b2w} is estimated from the images. Together with the water body, this provides a complete geometric model for the "calibration scene". Thus, we can synthesize images using assumed light and water parameters and compare rendered images to the captured reference (calibration) images in a multi-view analysis-by-synthesis-approach.

In a second – color-restoration – step, we use the obtained water and light parameters in another multi-view-approach to restore textures of underwater scenes captured in the same water, by the same camera-light system (see **right** part of the teaser image and Fig. 1). In principle, this could be extended also to recovering the object's BRDF or other parameters, but for this contribution we assume that the object is Lambertian.

[1] See [21] for the complications when using flat ports in 3D vision.

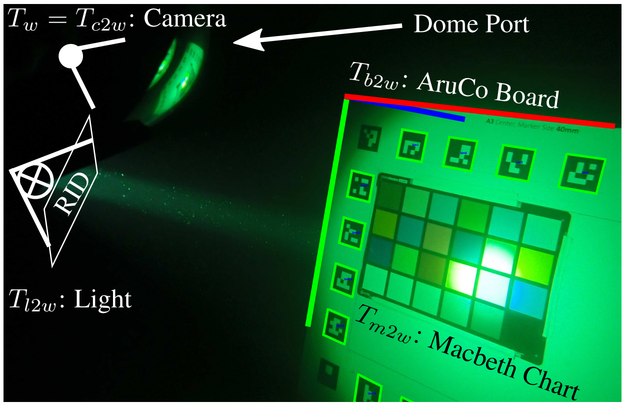


Figure 3. Real system setup and corresponding transformations.

3.1. Differentiable Ray Tracing

Instead of rendering an image I from a scene description X, i.e, $f(X) = I$, as common in computer graphics, we would like to invert this process to extract scene parameters from an image, hence $X = f^{-1}(I)$. As closed-form inversions are intractable for this kind of setting, we will use the autodiff-mode of a modified version of Mitsuba2 [30] throughout this paper. Specifically, we added a differentiable version of the phase function to yield a fully differentiable *piece of water*, i.e., a scattering homogenoeus medium, which can be optimized wrt. its parametrization. Furthermore, we derive this as a multi-view approach capable of handling many images at the same time, for smoothness within the autodiff framework we simultaneously employ closed-form gradients.

Non-Parametric Projector Light Model We define the light source as a projector texture of an inverse RGB-camera, see Fig. 3. Each pixel in the projector texture encodes the irradiance (from the point light) on the virtual image plane of the projector, which we refer to as $x^{RID} \in \mathbb{R}^{N\times M\times 3}$. We use 3 channels here to represent red, green and blue components, thus supporting also light sources that are more cold white for the inner angles but more reddish for outer angles. The advantage of this parameterization is that we can represent arbitrary patterns (e.g. square LEDs, or inner/outer cones) and it also implicitly encodes the light orientation inside the texture. Compared to explicitly parameterizing the exact light orientation using a quaternion or Euler angles, rotationally symmetric lights will not lead to degenerate parameter settings during estimation (in case the roll angle of a symmetric light is unobservable). Certain expected shapes or properties of the light can still be encouraged by imposing smoothness or priors onto the projector texture variables as we will show later.

Medium Model To obtain a solution of the volume rendering equation, we use the `volpathmis`-integrator of Mitsuba2. It uses unidirectional and next-event-estimation path-sampling strategies to evaluate a path integral-formulation[33, 44] in a Monte Carlo-scheme.[2]

Throughout this paper, we assume an isotropic homogeneous medium, which can be exhaustively described by its attenuation, albedo and scattering properties. The attenuation combines the loss of radiance due to out-scattering σ_s and absorption σ_a. In an isotropic homogeneous medium, the behavior is independent of the incoming direction and the interaction point, hence we simply have

$$\sigma_t = \sigma_a + \sigma_s. \tag{1}$$

The albedo further describes the composition of the sum given above by the probability of scattering versus absorption

$$p_a = \frac{\sigma_s}{\sigma_t}. \tag{2}$$

Due to our assumptions, σ_t and p_a are simply constant throughout the medium. We call the respective optimization parameters $x^{\sigma_t} \in \mathbb{R}^3$ and $x^{p_a} \in \mathbb{R}^3$. Photons that are absorbed simply *disappear* from the renderer, whereas scattered photons change their direction and can later be sensed by the camera as a “disturbance” (contributing to unsharpness, for small forward scattering angles, or to a foggy appearance, for larger scattering angles). The scattering direction has to be sampled and is parameterized using the Henyey-Greenstein phase function [17], which gives the probability for a photon to be scattered into an angle of θ, defined between the ingoing and outgoing direction:

$$p_{HG} = (cos\theta)\frac{1}{4\pi}\frac{1-g^2}{(1+g^2+2g(\cos\theta))^{3/2}}. \tag{3}$$

Again, this definition becomes possible because of the medium definition, where interactions at a certain position are independent of the incoming direction of a ray. The parameter g in this function controls the overall shape of the scattering, i.e. whether the light is scattered predominantly in forward or backward direction, or more uniformly in all directions. More complicated volume scattering functions exist for particular ocean waters (see [35, 13]) allowing for a more appropriate description[43], but for this presentation we stick to the simple model, which has just one parameter to optimize: $x^g \in \mathbb{R}$.

3.2. Robust Multiview Estimation

Using the above modifications and settings, we can render an image for a given set of parameters or define an objective function on that image, e.g. the squared difference between the rendered images and the real images, summed over all pixels. The differentiable raytracer will then provide the partial derivatives with respect to the light and

[2] Due to space limitations, we cannot discuss the entire physical model for raytracing. We refer the interested reader to the literature on physically-based rendering, in particular [36, p.888], which is also available online.

water parameters as described in the previous section. To minimize the objective function, the Adam optimizer [23] is used for a stochastic gradient descent approach. When working with real data, outliers and extreme noise may occur and have to be taken into account, therefore we have integrated the following extensions:

Huber Loss We replace the quadratic loss using a Huber loss function [18], denoted by $\rho_\delta(x)$, to improve stability by reducing the weight of outliers, especially caused by non-modeled objects such as floating particles in the scene.

Binary Mask We define also a mask as indicator function $M(\cdot)$ defined on an image I and providing a boolean array indicating for each pixel $I_{i,j}$ whether it should be considered in the cost function. When capturing calibration images, we can mask obvious outliers (such as fish swimming through the image).

Smoothness Constraint Since the light emitted from the light source is usually smooth, neighboring pixels in the RID should have similar values. Such a prior can be encouraged by adding a smoothness term

$$R(x^{RID}) = \sum_{i,j} (x^{RID}_{i,j} - x^{RID}_{i,j+1})^2 + (x^{RID}_{i,j} - x^{RID}_{i+1,j})^2 \quad (4)$$

to the data term used in optimization.

Objective Function and Calibration For calibration we minimize the overall objective

$$F(X) = G(X) + \alpha_s R\left(x^{RID}\right), \quad (5)$$

with $X = x^{RID}, x^{\sigma_t}, x^{p_a}, x^g$ and α_s being a scale factor for the smoothness term. To obtain the gradients for the first term of the objective, we use an autodiff step on the following function function across multiple reference images ref^v and their respective rendered counterparts img^v

$$G(X) = \underbrace{\sum_v \frac{1}{|M(ref^v)|_1} \sum_{i,j} M(ref^v_{i,j}) \rho_\delta(ref^v_{i,j} - img^v_{i,j}(X)),}_{\frac{\partial G}{\partial X} \text{ through backpropagation}} \quad (6)$$

which integrates the binary mask and the Huber norm. Then, we compute the second part in closed-form

$$\frac{\partial R}{\partial x^{RID}} = \left[\cdots, 2(x^{RID}_{i,j} - x^{RID}_{i,j+1}) + 2(x^{RID}_{i,j} - x^{RID}_{i+1,j}), \cdots\right]^T$$

We can now simply obtain the gradient of the overall objective function, due to linearity we have:

$$\frac{\partial F}{\partial X} = \frac{\partial G}{\partial X} + \alpha_s \frac{\partial R}{\partial X}. \quad (7)$$

Finally, we use this gradient to update X in an Adam update-step [23], and keep on iterating until the best parameters are found.

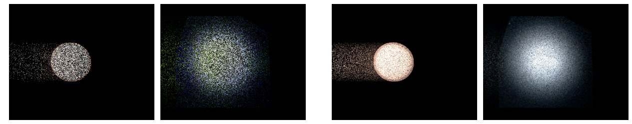


Figure 4. **Left** rendering with 1 SPP is faster but more noisy compared to 6 SPP, **right**. Every **first** image: first iteration (same start parameters) for light/water optimization. Every **second** image: after 1000 iterations the optimizations have diverged. The parameters obtained by 1SPP are significantly worse than those of 6SPP.

3.3. Restoration

The restoration pipeline is structured similar to the calibration. However, we now fix the water and light parameters to the previously estimated ones and turn to the problem of removing water and light effects from underwater images. For this, we now optimize a texture in a multi-view-analysis-by-synthetis approach. For surface color restoration of an underwater scene or object, the 3D geometry of the object is required. This can be obtained using a Structure-from-Motion pipeline[39]. Here we collect all surface colors in a texture that is attached to the object surface, e.g. represented as a triangle mesh, using OpenMVS [8] (see teaser figure). From there, and given the light and water calibration parameters, we minimize again the objective function, but this time the reference images do not constitute calibration images but are photos of a 3D scene. As initial values we can either use the surface color reconstruction from non-underwater methods, or simply initialize a black texture (c.f. teaser image and Fig. 1). Since the light source and the 3D geometry is contained in the scene, this approach for texture estimation automatically removes shadows explainable by the light source from the object texture.

4. Evaluation

The evaluation for this contribution is based on synthetic images and images taken in a water tank, mimicking greenish coastal water. Initially, we take a brief look at the samples per pixel (SPP) parameter. Depending on the number of SPP noise can be reduced, at the cost of longer runtime. The same holds true during estimation as we show in Fig. 4, where we compare optimization using 1SPP to 6SPP. In air it is often possible to run the differentiable ray racing-based optimization using 1SPP, since each image is just an intermediate step of the stochastic gradient-based optimization, and only the overall gradient has to be approximately correct. However, for the complex volumetric underwater effects we found that 1SPP often has bad convergence behavior and that it is better to render using at least 6SPP. Finally, to demonstrate the power of the approach, we are using extremely challenging light source patterns.

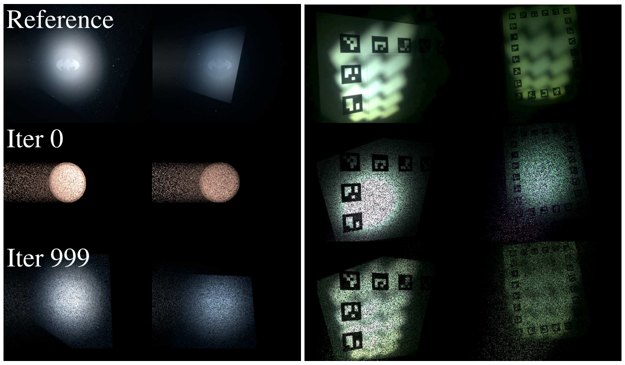


Figure 5. Examples of calibration images and development of their synthesized counterparts – **left**: synthetic, **right**: tank.

MSE	RID	σ_t	p_a	g
AB1 - single view	414.9	0.0275	0.0611	0.0001
AB2 - squared err.	137.1	0.0005	0.0002	0.0008
AB3 - Huber	135.9	0.0002	0.0002	0.0016
AB4 - smoothness	129.6	0.0005	0.0003	0.0008
AB5 - Hub.+smooth	125.2	0.0001	0.0002	0.0018

Table 1. Results of **AB1-5** shown as the mean squared error (MSE) after 500 iterations. The displayed MSE RID is the squared error between ground truth RID and estimated RID (in the range $[0; 1]$), summed over 250×250 projector pixels and 3 color channels. The other parameters are defined according to eqs. (1), (2) and (3).

4.1. Raytraced Data with Known Ground Truth

To validate the calibration step of our approach we set up a virtual scene with a light, a camera, an enclosing water body, as well as a virtual board that is presented to the camera in different poses. We introduced a multiview approach (presenting boards at different distances), based on the hypothesis that only spatially differing measurements will enable the algorithm to tell apart water properties and RID-color. This reasoning is reflected in the comparison of a single view run **AB1** with all other multiview runs **AB2-5**: Except for the surprisingly well scatter estimate, the other estimates are significantly worse (c.f. Tab. 2), since from a single view errors in water color can compensate for errors in the RID. We can – even on synthetic data – observe a positive effect on the RID-MSE between not using Huber-Loss **AB2** and and using it in **AB3**. As another aspect, a smoothness-constraint term is explicitly applied to the RID. When comparing the run **AB2** not using Huber-Loss with a run using the smoothness constraint **AB4**, we can observe a mildly positive effect on the RID, while the other estimates remain stable. Finally, **AB5** is the full system.

Detailed results of the latter are also displayed in Tab. 2, **left**, showing that all parameters were estimated accurately. Two sample calibration images are shown in Fig. 5, **left**. The estimated RID of the light source can be seen in Fig. 7, **left**, alongside the ground truth **middle**. Though the result is somewhat noisy, it can be seen that the general shape is recovered, including even the bat imprint in the center. Fig. 6 **upper row** displays the evolvement of the parameters (all initialized at 0.5) over the iterations. We can clearly see, that the estimation is quite reproducible, despite the substantial noise that is included in 6 SPP differentiable renderings (as shown in Fig. 4). δ was set to 0.3, learning rate to 0.02. In Fig. 6 **bottom row** we display the mean error and standard deviation of the estimated RID. The errors are very low, but show a slight overestimate of the light power. This is probably compensated (or caused) by a minimal overestimate of the water attenuation on the other hand.

Finally, we used the estimated parameters to restore the underlying texture of a 3D model. Additionally, we estimated a second set of parameters and used them for the same task. Our method is clearly capable of producing comparable models in different light and water settings, which is beneficial for underwater monitoring: c.f. Figs 1, 2.

Color Consistency We virtually equip the calibration board with a Macbeth color chart and render images in different poses. Afterwards we use the estimated water and light parameters to restore the texture of the chart as can be seen in Fig. 8 for two example poses. The different color fields of the board are enumerated as A1 to F4, and for each of them the central patch's mean color is compared to all other, independently restored boards by the patch error

$$p_{err} = \sum_{c \in RGB} \sum_{(p1,p2) \in P} \frac{1}{|P|} (\mu_{p1}^c - \mu_{p2}^c)^2. \qquad (8)$$

It is is computed by summing up the mean squared error between the mean value μ of a patch pair $(p1, p2)$ over all channels c, and is presented in Tab. 2. Overall, it is very small. The largest errors occur at the boundary of particular boards, where sometimes only very little light reaches, which makes the texture restoration challenging and increases the consistency error. For the restoration we used 64 SPP, $\delta = 0.5$ and a learning rate of 0.02.

The evaluation methods applied to the synthetic data (apart from the unknown ground truth) were also used in the subsequent real data evaluations.

4.2. Real Data

Real data was captured using a GoPro Hero 9 with a dome port, for which we have performed camera calibration underwater. We stored the images in raw format and convert them to float with linear response using *rawpy*/*LibRaw*. We have rigidly attached an *Aigend* - IPX8 18000lm 500M scuba diving light to the camera. We have modified it to exhibit a very demanding, high-frequency light pattern and a very narrow light cone for the tank experiment (see Fig. 5, **right** for sample calibration images).

We added green and blue bathtub colorants to the water, giving it a greenish hue with strong attenuation and mild

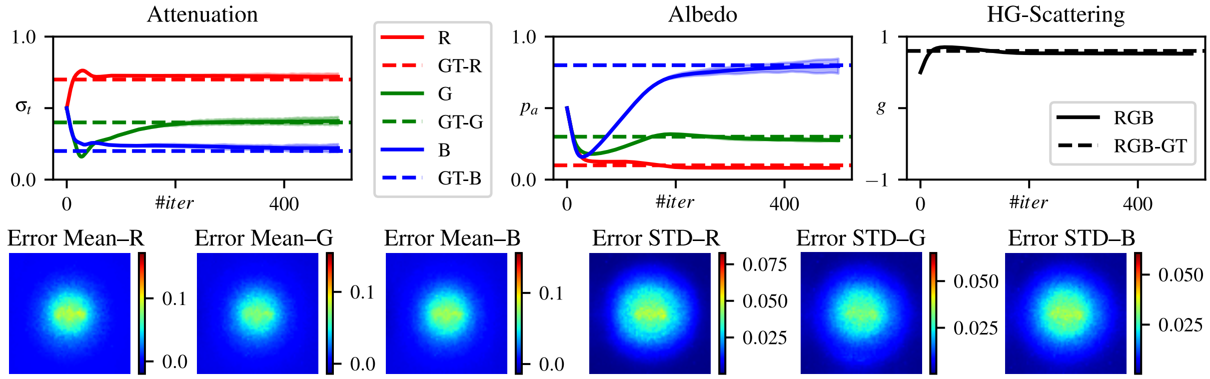


Figure 6. Synthetic parameter estimation experiment **EST1** avg'd over ten runs. The **upper row** shows the mean results of the parameter estimation with additional transparent fill of 3 STD. The **lower row** shows the RID estimation error, splitted into **left:** the mean error and **right:** the STD of the error. Please note: the results in the lower row are normalized to [0,1].

ES1	RID	σ_t	p_a	g	p_{err}	**A**	**B**	**C**	**D**	**E**	**F**
R	159.8	0.0001	0.0003	–	**1**	3.5191e-5	0.0005	0.0001	4.6867e-5	0.0015	0.1034
G	116.8	1.6543e-8	0.0004	–	**2**	0.0003	0.0006	8.5967e-5	0.0003	0.0007	0.1226
B	98.6	0.0002	9.2320e-5	–	**3**	0.0005	0.0003	0.0001	0.0003	0.0010	0.0340
RGB	125.2	0.0001	0.0002	0.0018	**4**	0.0117	0.0026	0.0002	0.0003	0.0013	0.0030

Table 2. **Left**: MSEs for all estimates in the final iteration step of **ES1**. **Right**: result of **EV1** on Macbeth chart, shown as the pairwise mean squared error between the evaluation patches in the final iteration. Please see Fig. 8 for patch encodings.

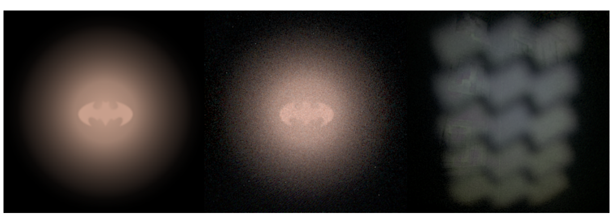


Figure 7. RIDs. **Left**: GT, **middle**: **ES1**, **right**: **ES2**.

	Estimates	p_{err}	**A**	**B**	**C**	**D**	**E**	**F**
σ_t	0.53, 0.17, 0.63	**1**	0.0091	0.0281	0.0336	0.0079	0.0163	0.0875
p_a	0.13, 0.59, 0.09	**2**	0.0056	0.0010	0.0118	0.0014	0.0225	0.1837
g	-0.53	**3**	0.0037	0.0120	0.0208	0.1244	0.0281	0.2082
–	–	**4**	0.0716	0.0077	0.0136	0.0015	0.0018	0.0002

Table 3. **Left:** tank estimation **ES2** and **right:** tank evaluation **EV2**. Please see Fig. 8 for the measurement patch encodings.

scattering. We have switched off all other illumination for the tank, which is black from the inside, providing a reasonably good simulation of the ocean at night or at depth. Then, we perform light and water optimization as described before. The estimated RID is displayed in Fig. 7 **right**, and qualitatively fits well to the modified light source. The estimated parameters are displayed in Tab. 3, **left**. We also submerged real Macbeth color charts and took images in various poses. Again, we restored each texture separately and computed the consistency among the corresponding patches (see Fig. 9). The colors of the Macbeth board are consistent though not as good as for the synthetic data (c.f. Tabs. 2, 3, **right**). However, note that we used an extremely challenging light source and that the greenish parts around the color boards are regions with very low light, where most red and blue information was lost. As shown in the teaser image, we can conduct a multi-view reconstruction of the colors of a 3D structure submerged in the test tank, even in this very challenging condition.

5. Discussion

The synthetic evaluation shows that differentiable raytracing can be used to simulate complicated scattering phenomena, jointly with attenuation and light cones, in a parameter estimation scenario. When images are presented in a good range, the algorithm can recover the light RID as well as the water parameters. It is even possible that light outside the field of view of the camera is scattered into the viewing volume, and constrains the RID further. As compared to in-air rendering, we observed that using a significantly higher SPP number is advisable. The restoration technique automatically recovers the surface colors or texture. The results become noisy where only little light reaches, but the examples shown are also already extreme cases to really demonstrate the method's capabilities. However, we have also encountered a few limitations and things to consider:

White Balance and Water-Light Ambiguity When following a ray from the light source into the camera, the sensed intensity is a product of the directed intensity emitted

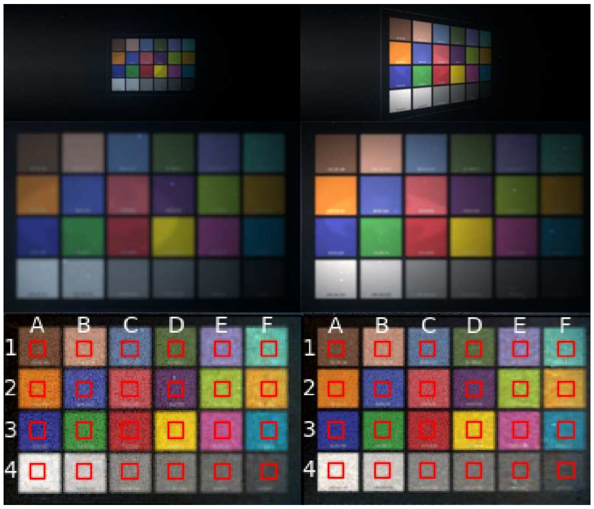


Figure 8. Results of **EVA1**. **Upper row** original views, **middle row** initial state, **lower row** final state of the color restoration. The red boxes indicate measurement areas for the errors in Tab. 2.

by the source, attenuation of the water, the albedo of the calibration target, again water attenuation, and the sensitivity of the camera system (plus scatter, which we neglect here, to make the ambiguity clear), all of which can vary with wavelength. We assume that camera and calibration target are known, such that the "unknowns" in this product are the light source and the water properties. Water attenuation depends on distance, which disambiguates the factorization into light and water color when using multiple object distances. However, if we use a single distance, errors in light intensity can be compensated by errors in water properties. This can only be resolved by distance variation. Since they are only factors, wrong assumptions about camera white balance or calibration target color will be absorbed into the light calibration, not influencing water property estimation.

Wide-band Coefficients As noted in [1] attenuation coefficients can vary significantly inside the visual spectrum. and the traditional way of parameter extraction is highly camera and scene-dependent. Consequently, it is better to use a spectral approach (coefficients for smaller wavelength intervals) as compared to wide-band coefficients (RGB). Currently, we use wide-band coefficients, which gave promising results in this work, still we are investigating the possibility using spectral raytracing for future work.

Calibration Objects So far, we rely on known calibration objects, which is a common approach. However, it is possible to replace this by using structure from motion, object detection and a catalog of known colors. As mentioned above, wrong calibration target colors would not influence the water-property estimation.

Sufficient Energy per Channel We should shoot enough light to be able to observe sufficient variance in each of the single color-channels. Although this is a generic observation, it should especially be kept in mind for a precise estimation of the water parameters.

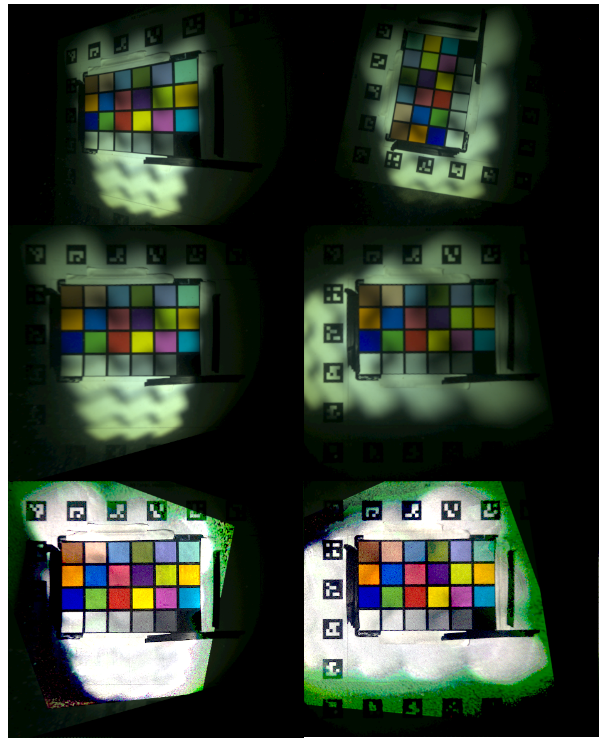


Figure 9. Results of **EV2**. **Upper row**: original views, **middle row** shows the inital, **lower row**: final state of the color restoration.

6. Conclusion

In this paper, we presented a joint light and medium calibration procedure based on differential ray tracing that has been successfully applied to the problem of water parameter estimation using a camera-light system. A corresponding technique processes the thus-estimated parameters for color-restoration. In general, this method can – by its nature – also be used in different settings involving joint light and medium estimation like e.g., calibrating headlights of cars, maybe for endoscopes or robots exploring caves, tunnels or other dark or foggy scenarios. We believe that for the underwater scenarios differentiable raytracing is a big step forward and future work should explore also other opportunities, e.g. more tight coupling with structure-from-motion systems or neural networks.

Acknowledgements

This publication has been funded by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) Projektnummer 396311425, through the Emmy Noether Programme. We are also grateful for support from the Chinese Scholarship Council (CSC) for M. She (202006050015) and Y. Song (201608080215).

References

[1] D. Akkaynak and T. Treibitz. A revised underwater image formation model. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6723–6732, 2018. 3, 8

[2] Derya Akkaynak and Tali Treibitz. Sea-thru: A method for removing water from underwater images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1682–1691, 2019. 3

[3] Derya Akkaynak, Tali Treibitz, Tom Shlesinger, Yossi Loya, Raz Tamir, and David Iluz. What is the space of attenuation coefficients in underwater computer vision? In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 568–577. IEEE, 2017. 3

[4] Dejan Azinovic, Tzu-Mao Li, Anton Kaplanyan, and Matthias Nießner. Inverse path tracing for joint material and lighting estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2447–2456, 2019. 3

[5] Deniz Beker, Hiroharu Kato, Mihai Adrian Morariu, Takahiro Ando, Toru Matsuoka, Wadim Kehl, and Adrien Gaidon. Monocular differentiable rendering for self-supervised 3d object detection. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 514–529, Cham, 2020. Springer International Publishing. 2

[6] Mitch Bryson, Matthew Johnson-Roberson, Oscar Pizarro, and Stefan B Williams. True color correction of autonomous underwater vehicle imagery. *Journal of Field Robotics*, 33(6):853–874, 2016. 3

[7] Gershon Buchsbaum. A spatial processor model for object colour perception. *Journal of the Franklin institute*, 310(1):1–26, 1980. 3

[8] Dan Cernea. OpenMVS: Multi-view stereo reconstruction library. 2020. 5

[9] Chengqian Che, Fujun Luan, Shuang Zhao, Kavita Bala, and Ioannis Gkioulekas. Inverse transport networks. *arXiv preprint arXiv:1809.10820*, 2018. 2, 3

[10] António Correia, Peter Hanselaer, Hugo Cornelissen, and Youri Meuret. Radiance based method for accurate determination of volume scattering parameters using gpu-accelerated monte carlo. *Opt. Express*, 25(19):22575–22586, Sep 2017. 3

[11] John J. Cullen, Áurea M. Ciotti, Richard F. Davis, and Marlon R. Lewis. Optical detection and assessment of algal blooms. *Limnology and Oceanography*, 42(5part2):1223–1239, 1997. 1

[12] BW Eakins and GF Sharman. Hypsographic curve of earth's surface from etopo1. *NOAA National Geophysical Data Center, Boulder, CO*, 5, 2012. 1

[13] Georges R. Fournier and Miroslaw Jonasz. Computer-based underwater imaging analysis. In Gary D. Gilbert, editor, *Airborne and In-Water Underwater Imaging*, volume 3761, pages 62 – 70. International Society for Optics and Photonics, SPIE, 1999. 4

[14] Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco José Madrid-Cuevas, and Manuel Jesús Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014. 3

[15] Ioannis Gkioulekas, Shuang Zhao, Kavita Bala, Todd Zickler, and Anat Levin. Inverse volume rendering with material dictionaries. *ACM Transactions on Graphics (TOG)*, 32(6):1–13, 2013. 3

[16] Alexander Grabner, Yaming Wang, Peizhao Zhang, Peihong Guo, Tong Xiao, Peter Vajda, Peter M Roth, and Vincent Lepetit. Geometric correspondence fields: Learned differentiable rendering for 3d pose refinement in the wild. In *European Conference on Computer Vision*, pages 102–119. Springer, 2020. 2

[17] Louis G Henyey and Jesse Leonard Greenstein. Diffuse radiation in the galaxy. *The Astrophysical Journal*, 93:70–83, 1941. 4

[18] Peter J Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics*, pages 492–518. Springer, 1992. 5

[19] Jules S Jaffe. Computer modeling and the design of optimal underwater imaging systems. *IEEE Journal of Oceanic Engineering*, 15(2):101–111, 1990. 2

[20] Wenzel Jakob. Enoki: structured vectorization and differentiation on modern processor architectures, 2019. https://github.com/mitsuba-renderer/enoki. 2

[21] Anne Jordt, Kevin Köser, and Reinhard Koch. Refractive 3d reconstruction on underwater images. *Methods in Oceanography*, 15-16:90–113, 2016. Computer Vision in Oceanography. 3

[22] Hiroharu Kato, Deniz Beker, Mihai Morariu, Takahiro Ando, Toru Matsuoka, Wadim Kehl, and Adrien Gaidon. Differentiable rendering: A survey. *arXiv preprint arXiv:2006.12057*, 2020. 2

[23] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. 5

[24] Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. Differentiable monte carlo ray tracing through edge sampling. *ACM Transactions on Graphics (TOG)*, 37(6):1–11, 2018. 2

[25] B. L. McGlamery. Computer analysis and simulation of underwater camera system performance. Technical report, Visibility Laboratory, Scripps Institution of Oceanography, University of California in San Diego, 1975. 2

[26] Patricia Miloslavich, Nicholas J. Bax, Samantha E. Simmons, Eduardo Klein, Ward Appeltans, Octavio Aburto-Oropeza, Melissa Andersen Garcia, Sonia D. Batten, Lisandro Benedetti-Cecchi, David M. Checkley Jr., Sanae Chiba, J. Emmett Duffy, Daniel C. Dunn, Albert Fischer, John Gunn, Raphael Kudela, Francis Marsac, Frank E. Muller-Karger, David Obura, and Yunne-Jai Shin. Essential ocean variables for global sustained observations of biodiversity and ecosystem changes. *Global Change Biology*, 24(6):2416–2433, 2018. 1

[27] Curtis D Mobley. *Light and water: radiative transfer in natural waters*. Academic press, 1994. 2

[28] Srinivasa G Narasimhan, Mohit Gupta, Craig Donner, Ravi Ramamoorthi, Shree K Nayar, and Henrik Wann Jensen. Ac-

quiring scattering properties of participating media by dilution. In *ACM SIGGRAPH 2006 Papers*, pages 1003–1012. 2006. 3

[29] S. K. Nayar and S. G. Narasimhan. Vision in bad weather. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 820–827 vol.2, 1999. 3

[30] Merlin Nimier-David, Delio Vicini, Tizian Zeltner, and Wenzel Jakob. Mitsuba 2: A retargetable forward and inverse renderer. *ACM Transactions on Graphics (TOG)*, 38(6):1–17, 2019. 2, 4

[31] Jan Novák, Iliyan Georgiev, Johannes Hanika, and Wojciech Jarosz. Monte carlo methods for volumetric light transport simulation. *Computer Graphics Forum*, 37(2):551–576, 2018. 3

[32] Jaesik Park, Sudipta N Sinha, Yasuyuki Matsushita, Yu-Wing Tai, and In So Kweon. Calibrating a non-isotropic near point light source using a plane. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2259–2266, 2014. 3

[33] Mark Pauly, Thomas Kollig, and Alexander Keller. Metropolis light transport for participating media. In *Eurographics Workshop on Rendering Techniques*, pages 11–22. Springer, 2000. 4

[34] V. Periyasamy and M. Pramanik. Advances in monte carlo simulation for light propagation in tissue. *IEEE Reviews in Biomedical Engineering*, 10:122–135, 2017. 3

[35] Theodore J Petzold. Volume scattering functions for selected ocean waters. Technical report, Scripps Institution of Oceanography La Jolla Ca Visibility Lab, 1972. 3, 4

[36] Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically based rendering: From theory to implementation*. Morgan Kaufmann, 2016. 2, 4

[37] Sylvain Pinet, Jean-Michel Martinez, Sylvain Ouillon, Bruno Lartiges, and Raul Espinoza Villar. Variability of apparent and inherent optical properties of sediment-laden waters in large river basins; lessons from in situ measurements and bio-optical modeling. *Opt. Express*, 25(8):A283–A310, Apr 2017. 1

[38] RW Preisendorfer. Physical aspect of light in the sea. *Univ. Hawai. Pre8s. Honolulu Hawaii 51*, 60, 1964. 2

[39] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 5

[40] Mengkun She, David Nakath, Yifan Song, and Kevin Köser. Refractive geometry for underwater domes. *arXiv preprint*, 2021. 3

[41] Mengkun She, Yifan Song, Jochen Mohrmann, and Kevin Köser. Adjustment and calibration of dome port camera systems for underwater vision. In Gernot A. Fink, Simone Frintrop, and Xiaoyi Jiang, editors, *Pattern Recognition - 41st DAGM German Conference, DAGM GCPR 2019, Dortmund, Germany, September 10-13, 2019, Proceedings*, volume 11824 of *Lecture Notes in Computer Science*, pages 79–92. Springer, 2019. 3

[42] Mark Sheinin and Yoav Y Schechner. The next best underwater view. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3764–3773, 2016. 3

[43] Alberto Tonizzo, Michael Twardowski, Scott McLean, Ken Voss, Marlon Lewis, and Charles Trees. Closure and uncertainty assessment for ocean color reflectance using measured volume scattering functions and reflective tube absorption coefficients with novel correction for scattering. *Applied optics*, 56(1):130–146, 2017. 4

[44] Eric Veach. *Robust Monte Carlo methods for light transport simulation*. Stanford University, 1998. 4

[45] Takaki Yamada, Adam Prügel-Bennett, and Blair Thornton. Learning features from georeferenced seafloor imagery with location guided autoencoders. *Journal of Field Robotics*, 38(1):52–67, 2021. 3