StreamUR: Physics-informed Near Real-Time Underwater Image Restoration

Christos Antoniou, Sotiris Spanos, Simon Vellas, Valsamis Ntouskos, Konstantinos Karantzalos

Remote Sensing Lab, SRSE-GE, National Technical University of Athens, 15773 Zographou, Greece
{chrant.mail,spanossotiris1998}@gmail.com
{svellas,ntouskos}@mail.ntua.gr
karank@central.ntua.gr

Keywords: underwater imaging, color restoration, deep learning, computer vision, embedded systems.

Abstract

The exploration of underwater environments poses significant challenges due to the optical properties of water, leading to color distortion, reduced contrast and blurring in images. This work aims to enhance the clarity and fidelity of underwater images and videos in near real-time. The *SeaThru* physics-based color correction method was suitably adapted for obtaining target images across a diverse collection of underwater datasets considered. Based on these target images, the *MIMO-UNet* model is used to address the processing speed limitations of the physics-based correction methods, enabling near real-time image and video processing without explicit depth information. The proposed method has been integrated into autonomous underwater observation systems and remotely operated vehicle (ROV) cameras, offering enhanced visibility. Additionally, we build a *MIMO-UNet* network for generating realistic synthetic underwater images, valuable for training and simulation. This research advances underwater imaging enhancement and restoration, significantly improving visual data quality and vision-dependent tasks in submerged environments. The public release of the dataset aims to facilitate further research and development in this field.

1. Introduction & Related Work

The exploration of underwater environments presents unique challenges due to the complex optical properties of water, which significantly affect visibility and image quality. Traditional methods have struggled with color distortion, low contrast, and blurring caused by light scattering and absorption. To address these challenges, various image restoration methods have been proposed in the literature.

Underwater image enhancement methods can be broadly categorized into traditional and learning-based approaches. Traditional methods, such as dark channel prior (He *et al.*, 2010) and histogram equalization (Chen *et al.*, 2022), rely on prior assumptions regarding image formation and distortions to obtain clear images. However, these methods may fail when the assumptions do not hold. On the other hand, learning-based methods are typically trained using synthetic underwater images because obtaining clear underwater images is very challenging. Synthetic training data provide clear reference images corresponding to degraded original images, allowing to train models that restore original colors from underwater images (Z. Wang *et al.*, 2023). However, models trained solely on synthetic data may struggle to perform well on real-world underwater images, due to the domain gap. While some learning-based methods developed using Generative Adversarial Networks (GANs) (Li et al., 2017), (Q. Wang *et al.*, 2023) do not require synthetic images and focus on improving perceptual quality, they can sometimes result in significant color artifacts.

Our approach builds on the "SeaThru" method (Akkaynak & Treibitz, 2019), a color correction algorithm in underwater imaging, which employs the revised physics-based model of (Akkaynak & Treibitz, 2018) for removing the effects of water from underwater images. This method diverges from other techniques that prioritize aesthetic enhancement over scientific accuracy, focusing instead on the accurate restoration of true colors by accounting for light's behavior through water in different wavelengths.

As an official implementation of SeaThru was not available, we developed our own to apply and refine the algorithm across our datasets, producing a large collection of reference data suitable for training deep learning methods. Ensuring consistency and robustness through fine-tuning of the hyperparameters and post-processing steps like white balancing was a demanding and time-consuming task.

To supply depth information essential for the SeaThru algorithm, we incorporated outputs from a pre-trained monocular depth estimation approach based on a neural network. Recent advancements in monocular depth estimation, such as (Godard *et al.*, 2019) and (Ranftl *et al.*, 2021), have shown promising results in providing accurate depth cues from single images. We utilize (Ke *et al.*, 2023) in our framework, as it produces the most robust results on our datasets, even if it is not explicitly trained in underwater images.

Aiming to assimilate the application of the reference underwater restoration method using an efficient neural network, we adjusted and trained the MIMO-UNet (Cho *et al.*, 2021), as its architecture is proven to function well on the image restoration task of deblurring, utilizing end-to-end supervised learning and addressing the processing speed limitations inherent in the SeaThru method. The developed model directly transforms underwater RGB images into restored images in real-time, without requiring explicit depth information. This advancement facilitates near real-time processing of images and videos, enabling numerous practical applications involving real-time image streams in dynamic underwater environments.

Additionally, we developed and trained the same neural network architecture to perform the reverse transformation: adding back the effects of water into corrected images. This process is essential for creating realistic synthetic underwater data for simulations and lab testing, where access to actual underwater environments may be limited. Our approach facilitates the generation of synthetic images that are affected by the distortions caused by water, offering a convenient tool for training and refining underwater imaging and analysis algorithms, similarly to the work of (Li *et al.*,2020) and (Wen *et al.*, 2023) where the revised underwater image formation model suggested in (Akkaynak & Treibitz, 2018) is used to produce synthetic underwater images. The proposed method leverages the actual distortions and color changes captured in the original underwater images, offering an efficient and more realistic underwater image simulation.

The deep learning models that are developed are also evaluated on embedded AI computational platforms like the NVIDIA Jetson Nano, ensuring that they operate efficiently even in hardware-constrained environments. This consideration is crucial for deployment in autonomous underwater vehicles, where power and processing capabilities are often limited.

To demonstrate the applicability and versatility of our method, we integrated the proposed efficient correction model into remote sensing applications within autonomous underwater systems, such as an autonomous underwater observation system and the camera of a Remotely Operated Vehicle (ROV). This integration allows for the real-time correction of underwater scenes, greatly enhancing the visibility and color fidelity of the environment. Additionally, this correction aids in improving the functionality of other vision-based algorithms by providing them with clearer and radiometrically accurate visual data. For instance, the application of the proposed method has notably enhanced the performance of an underwater bubble identification neural network for an autonomous underwater observation system (Spanos *et al.*, 2024).

In summary, the proposed StreamUR method introduces several novel contributions to the field of underwater computer vision:

1. Physics-based, robust reference data: Unlike other learning-based methods that rely on synthetically enhanced images for training, our approach uses physics-based corrected images as reference, ensuring higher fidelity.

2. Real-Time Processing: The MIMO-UNet enables near real-time (5-10 fps) correction of underwater images and video streams, a significant improvement over the much less efficient SeaThru implementation, ensuring the feasibility of near real-time visual applications in underwater robotics.

3. Independence from depth information: By relying on depth cues intrinsic in underwater images, the dependency on depth data can be lifted. Based on this, our method simplifies the operational complexity of underwater imaging systems, which often struggle to obtain accurate depth information.

4. Validation: We have successfully integrated and tested our model in real-world applications, including autonomous underwater visual observation systems and remotely operated vehicles (ROVs), demonstrating practical utility beyond in real environments.

5. Synthetic data generation: The development of a neural network capable of generating realistic underwater images from corrected images provides a valuable tool for creating high fidelity synthetic datasets, crucial for training and simulation purposes in the absence of extensive real-world data.

6. Public dataset provision: The release of our underwater imagery dataset will facilitate further research and development in the field, promoting innovation and improvement in underwater imaging technologies.

2. Methodology

2.1 SeaThru physics-based correction

The SeaThru method (Akkaynak & Treibitz, 2019), represents a significant advancement in underwater image processing by addressing the unique challenges posed by the underwater environment. Traditional methods for underwater image enhancement often fail to correct color distortions and image degradation caused by light absorption and scattering in water. SeaThru, however, uses a physics-based approach to recover the true colors of underwater scenes. Underwater image formation is mostly influenced by:

- **Backscatter**: Particles suspended in water scatter light back to the camera, creating a hazy appearance that reduces image clarity and contrast.
- **Attenuation**: Light intensity diminishes as it travels through water due to absorption and scattering. This reduction in light affects different wavelengths to varying degrees, leading to color distortions.

Considering some simplifying assumptions on the model introduced in (Akkaynak & Treibitz, 2018), the model considered in SeaThru is:

$$I_c = J_c\, e^{-\beta_c^D z} + B_\infty (1 - e^{-\beta_c^B z}),$$

with I_c the observed image for wide-band channel c, J_c the corresponding image of the scene unaffected by the water, B_∞ the asymptotic value of the backscatter as depth z increases, and β_c^B /β_c^D the backscatter and attenuation coefficients, respectively.

SeaThru aims to separate and correct the effects of Backscatter and Attenuation using the following key steps:

- **Depth Map Estimation**: Accurate depth information is crucial for modeling light attenuation and backscatter.
- **Backscatter Removal**: The method identifies and removes the backscatter component of the image as a function of depth
- **Attenuation Coefficient Estimation**: The attenuation of light is modeled as a depth-dependent coefficient, which varies with wavelength.
- **Image Reconstruction**: After removing backscatter and correcting for attenuation, the remaining direct signal is adjusted to restore the image's true colors. This step involves applying white balancing techniques to ensure color accuracy.

By implementing these steps, the water's influence from underwater images is effectively removed, restoring their true colors and significantly enhancing their clarity. A detailed description of the steps involved is provided in the Appendix.

2.2 Efficient underwater image correction

The physics-based restoration model described above can correct the images from distortions caused by the water, however it poses several challenges as far as ease of use and efficiency are concerned. This is due to two main factors: i) the necessity of estimating the depth man of the scene for each image/frame and

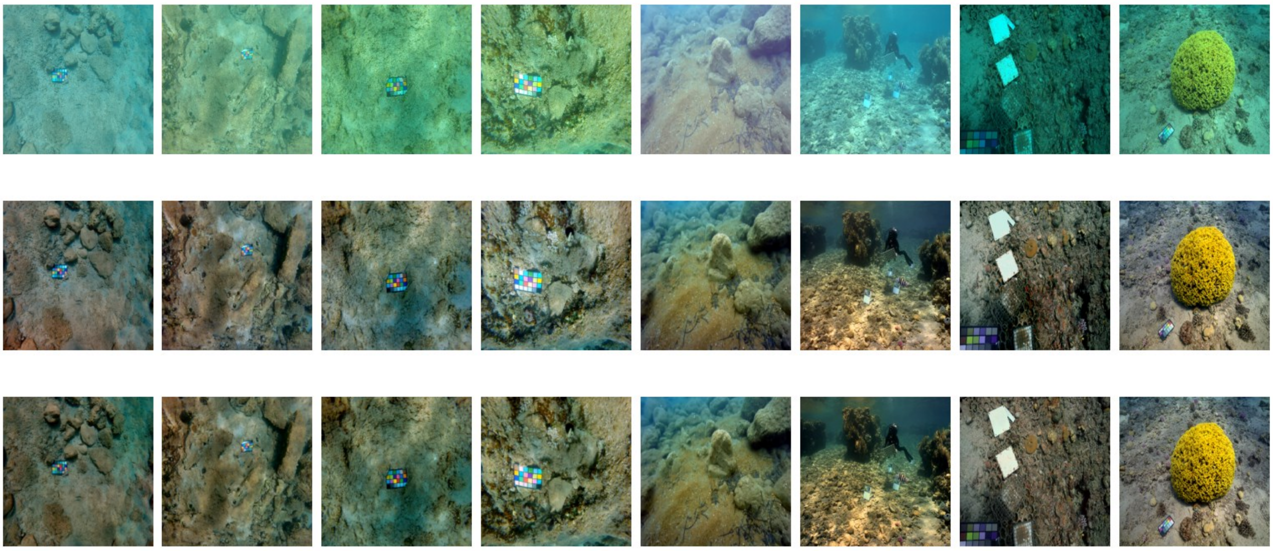

Figure 1: In the first row, seven underwater images are show from the validation split of the dataset we collected as well as from the SeaThru dataset. In the second row, the refence images corrected using the SeaThru physics-based method are shown, while the third row shows the images corrected using the proposed method.

ii) the long processing time of the steps which cannot be easily parallelized for exploiting modern GPU capabilities.

We address these limitations by training an efficient state-of-the-art image restoration deep learning model with the task of directly generating the corrected images given the underwater images. Specifically, we apply the SeaThru physics-based correction method across a wide variety of underwater images, and use the corrected images, coupled with the original ones, as reference data, for training end-to-end a MIMO-UNet (Cho et al., 2021) in a supervised manner. Moreover, we relax the requirement of providing the depth map of the scene and rely on the ability of the model to effectively exploit depth cues that are intrinsic in the underwater images for performing the correction.

The MIMO-UNet, or Multi-Input Multi-Output U-Net, has been introduced for the task of image deblurring and has been shown to handle complex image restoration tasks efficiently. The network is designed to be computationally efficient, making it suitable for near real-time applications and applications involving limited computational resources. By using a single U-Net structure with multi-scale inputs and outputs, the network reduces the computational overhead typically associated with stacked sub-networks. The architecture's ability to handle multi-scale features and its efficient feature fusion mechanism contribute to its superior performance in image restoration tasks.

By utilizing the MIMO-UNet architecture, we were able to train a robust neural network capable of producing high-quality corrected underwater images. This network offers a practical solution for near real-time underwater image enhancement, without requiring costly explicit depth estimation of the scene.

2.3 Underwater image generation

In addition to training the MIMO-UNet for underwater image enhancement, we also developed a reverse model using the same architecture. This reverse model was trained in a similar manner, but with a key difference: this time, the corrected images were used as inputs and the corresponding original underwater images as outputs. This allowed us to build a model capable of generating synthetic underwater data from images taken in the air.

The synthetic data generated by our model faithfully mimics the effects of depth-dependent attenuation and backscatter, resulting in more accurate and useful training data for underwater image processing tasks.

By developing this reverse model, we created a tool for generating high-quality synthetic underwater data from in-air images. This synthetic data can be invaluable for training and validating underwater imaging algorithms, offering a more realistic and effective alternative to traditional methods of synthetic data generation.

2.4 Embedded applications

Autonomous monitoring applications in remote environments require the deployment of refined data processing algorithms on edge devices, utilizing highly efficient processing architectures. Edge computing involves performing data processing near the data source rather than relying on a centralized infrastructure, allowing to extract task-specific information from large volumes of collected data, reducing storage requirements and enabling real-time decision making. Additionally, several tasks rely on the processing of a continuous stream of incoming sensor data, close to the source, for decision-making or navigation.

In this context, we examine the deployment of our developed model on low-power processing architectures designed for high-performance edge computing. Specifically, we target the Nvidia Jetson Nano 4GB development board, hosting an embedded GPU. The deployment on the Jetson GPU clearly benefits from the high performance/power ratio of the device, allowing for near real-time processing of incoming images, while consuming less than 10W of power.

Deploying the model on the Nvidia embedded GPU requires setting up the necessary environment, similar to a desktop GPU. However, to achieve maximum performance, it is essential to convert the model to TensorRT, a proprietary Nvidia framework

https://doi.org/10.5194/isprs-archives-XLVIII-3-2024-1-2024 | © Author(s) 2024. CC BY 4.0 License.

Figure 2: On the top row, video frames collected from the ROV camera with 10 frame intervals are presented. The second row shows the corresponding results produced in real-time by the proposed model.

designed to fully exploit the Nvidia GPU architecture by using a combination of techniques such as layer fusion, precision calibration and dynamic tensor memory management.

3. Datasets

We utilized distinct datasets to ensure a comprehensive and robust evaluation of our underwater image enhancement techniques. The datasets include:

- **Datasets from Previous Studies**: We incorporated datasets from (Akkaynak & Treibitz, 2019) and (Li *et al.*, 2019), which consist of various underwater images captured under different conditions.
- **Proprietary Dataset**: We collected a custom underwater dataset using a GoPro camera. The collected dataset includes a wide range of underwater environments, depths, and lighting conditions, providing a diverse set of images. To enhance the accuracy of our quality assessment, we included a color chart in the images. This allowed for a more precise estimation of the color correction and image restoration quality.
- **ROV Camera Dataset**: We also considered a proprietary dataset comprising video frames captured from the camera of a small Remotely Operated Vehicle (ROV). This dataset offers a dynamic perspective, featuring underwater videos that are crucial for evaluating the performance of our real-time processing algorithms.

The combined dataset consists of 3010 images (2102 from proprietary dataset), showcasing a diverse array of underwater conditions and environments. Initially, we applied the SeaThru method to these datasets for generating a set of corrected images that served as reference data. Subsequently, we used the original datasets along with the SeaThru-corrected images to train the MIMO-UNet image restoration model. This training process enables the model to perform physics-informed near real-time underwater image correction with high accuracy, by mimicking the restoration performed by the SeaThru physics-based model. The diversity of the datasets helps our model to generalize well across various underwater conditions, making it robust and reliable for practical applications. The whole dataset was augmented to 6,000 images by randomly applying various transformations such as rotations and flips to the original images.

The proprietary dataset comprising the images collected using the GoPro camera and the mini ROV, is publicly available at this link.

4. Experimental Evaluation

4.1 Implementation

The MIMO-UNet forward and reversed models were trained on an NVIDIA GeForce GTX 980 Ti, 8GB V-RAM GPU. Training was carried out for 35 epochs, which took about 700 minutes on this hardware. Images are resized to 256×256 size before being fed to the model. Regarding the most important hyperparameters, batch size was set to 2 due to limited V-RAM size. The learning rate was set to $1 \cdot 10^{-4}$. Moreover, ADAM algorithm was used as the optimizer since it was experimentally found to outperform any other optimizer.

4.2 Training Process

The training of the underwater image restoration model was performed using a fully supervised learning approach, leveraging the corrected images produced by the SeaThru physics-based method as target data. This setup allows the MIMO-UNet to learn the complex mappings required to correct underwater images directly in an efficient way. Unlike the original SeaThru algorithm, we did not consider depth maps as inputs for our network, providing only the images. This simplifies the application of the network by eliminating the need for scene depth information. The full augmented dataset of 6,000 images is considered for training, considering a training-validation split of 80%-20%, respectively.

The training of the reverse model, used for conditional underwater image generation, is based on the same full augmented collection of 6,000 images. However, in this case, the training pairs consisted of the corrected images (produced by the SeaThru algorithm) as inputs and the original underwater images as outputs.

This setup ensured that the network learned to map clear, in-air images to their underwater counterparts, capturing the complex distortions and color shifts that occur underwater. As with the enhancement model, the training was conducted using a fully supervised learning approach with the degraded underwater images as targets and the corrected water-free images as inputs.

https://doi.org/10.5194/isprs-archives-XLVIII-3-2024-1-2024

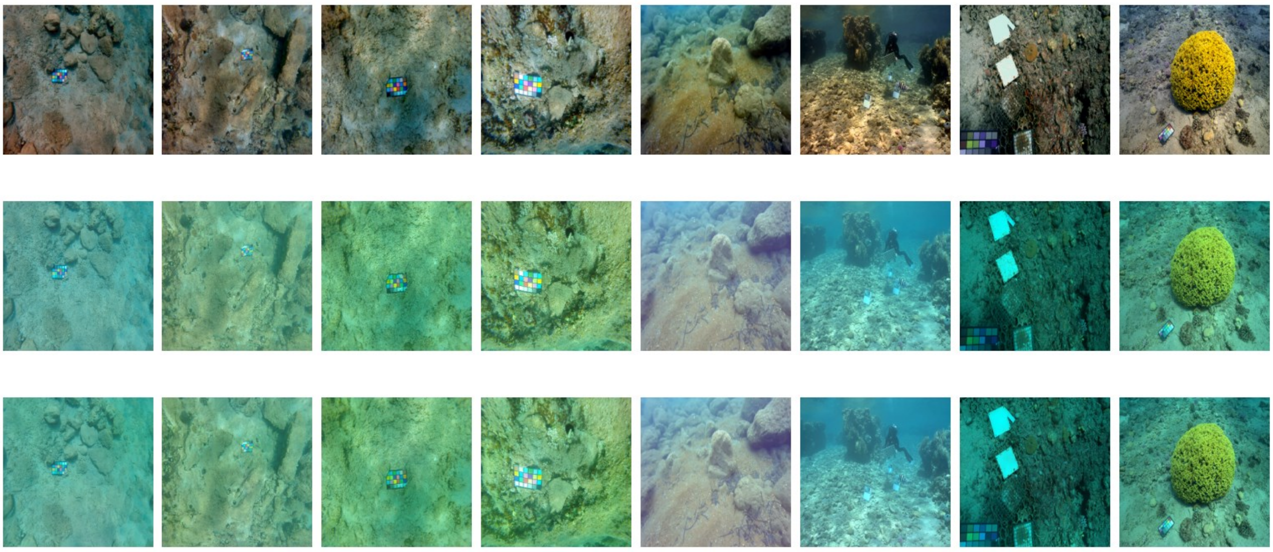

Figure 3: In the first row, the images are corrected using the SeaThru implementation. In the second row there are the initial underwater images. In the third row the images are passed through our reverse model (water additive).

4.3 Metrics

Models that perform color correction (enhancement) on images, especially underwater images, are quite challenging to evaluate, particularly in qualitative terms. To address this, we use two standard quantitative image quality assessment metrics, namely Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) (Z. Wang *et al.*, 2004).

PSNR is a metric used to measure the quality of a reconstructed image compared to its original noise-free version. It is expressed in decibels (dB), with higher values indicating better quality and less noise. SSIM, on the other hand, assesses the structural similarity between two images. It ranges from -1 to 1, where values closer to 1 indicate higher similarity and better preservation of image structure.

4.4 Results

To evaluate our models, we consider two aspects: how close the model outputs are to the ones produced by the physics-based approach, and how realistic they seem.

The first aspect is summarized in the quantitative results presented in Table 1. One can note that both model implementations (forward and reverse) achieved quite satisfying results for these complex tasks. SSIM values in both cases were close to 1 (perfect similarity), and the PSNR exceeded 30, indicating differences that are difficult to notice without carefully observing details on specific parts of the image. An interesting observation is that the reverse model achieved better results which is mostly due to the nature of the task. The forward model performs an enhancement process which must respect the lower frequencies of the image while enhancing the high ones, which is a notably harder task. On the other hand, the reverse model could be considered as a degradation process which is mostly correlated with the reconstruction of the lower frequencies of the images, which constitutes an easier task overall.

Regarding qualitative assessment, a color chart was used in the captured images. This allows us to perform a more detailed analysis as known color components are present in the images.

	Forward Model	Reverse Model
Average PSNR	30.287	32.790
Average SSIM	0.964	0.971

Table 1 Model metrics averaged over the test split of the collected dataset for the forward and reverse models.

As we can observe in Figure 1, most of the images produced by the efficient underwater image restoration model are quite realistic overall. With respect to the color chart, although some minor color inconsistencies are observed which mainly consist of bluish and pink hues, it is reconstructed quite faithfully. An interesting fact is that, even though the initial images had significant illumination differences due to varying depth and time of acquisition (morning to evening), the model outputs tend to be homogenous. Another interesting fact is that although our results are very similar to the physical model, we can observe that they have less color artifacts, which can be attributed to the generalization capabilities of the neural network employed.

The efficient underwater enhancement model achieves an effective processing rate of 5-10 frames per second (FPS) on the hardware mentioned in Section 4.1. Figure 2 presents frames taken from a video stream corrected in real-time using the proposed model, illustrating its capabilities to efficiently correct images while producing homogenous outputs suitable for video enchantment purposes.

Regarding the qualitative evaluation of the reverse model, Figure 3 shows the corresponding results. It is evident that the model performs as intended with only minor hardly noticeable differences between the reference images and the model outputs. Based on this, we performed further trials feeding our reverse model with real-world images, as shown in Figure 4. From these examples we can observe that the reverse model tends to produce images with similar degradation to the one observed in underwater images. This suggests that the reverse model could be used as an effective underwater image generation approach.

https://doi.org/10.5194/isprs-archives-XLVIII-3-2024-1-2024 |

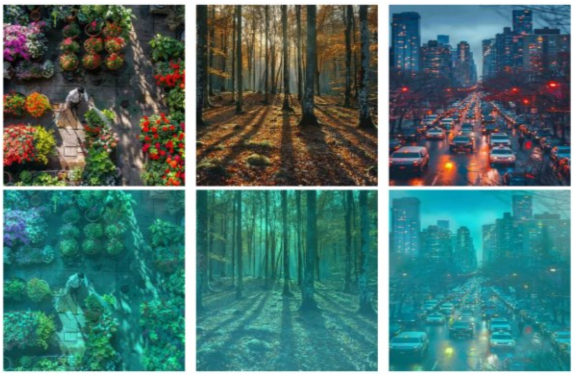

Figure 4: The first row shows images collected from the web, while the second row shows the respective images after the usage of the reverse (water additive) model.

4.4 Edge computing results

For the deployment of a PyTorch model on an Nvidia Jetson device, the architecture does not necessarily require modification or specific adaptation to the embedded GPU. However, converting the model to TensorRT is mandatory to achieve maximum processing performance, as described in Section 2.4.

In our case, the inherent complexity of MIMO-UNet, due to its multi-scale architecture, limits its performance on very small devices, especially those with low memory. A performance of 3-4 FPS can be achieved for the MIMO-UNet model on a 4GB Jetson Nano without resorting to aggressive optimization techniques such as model pruning and quantization which, if required, could help achieve even higher frame rates.

A version of the proposed StreamUR method, suitably adapted for the Jetson Nano embedded AI single-board computer has been tested in the autonomous video capturing system presented in (Spanos *et al.*, 2024) for enhancing the images before performing bubble detection.

5. Conclusions & Future work

In this work, we presented a comprehensive approach for enhancing underwater images by leveraging both traditional physics-based methods and advanced deep learning techniques. Our work builds on the SeaThru algorithm, a SOTA approach for underwater image correction, and extends its capabilities utilizing the established deep neural network architecture, MIMO-UNet. By mimicking the physics-based correction approach of SeaThru, a physics-informed neural network is introduced for efficiently correcting the main distortions present in underwater images, namely the backscatter and the per-channel attenuation. In this way, we achieve comparable image clarity and enhanced color reconstruction, while achieving near real-time processing, which is not feasible with physics-based approaches. Moreover, the usage of the MIMO-UNet network eliminates the need of hard-to-obtain depth information that is used in the physics-based approach. By utilizing a reverse training paradigm, we produced a generator model that provides the ability to create novel datasets depicting underwater scenes synthetically, without having the necessity to perform in situ data acquisition. Our custom-made dataset, consisting of GoPro camera frames and ROV video frames, in combination with publicly available datasets offered higher generalization capabilities and robustness of our approach.

The use of our models in autonomous underwater systems, such as observation platforms and ROVs, demonstrates their practical utility in various underwater applications. The ability to perform real-time image correction significantly enhances the visibility and data quality in underwater exploration and monitoring tasks.

Our work highlights the potential of combining traditional physics-based approaches with modern deep learning techniques to address complex challenges in underwater imaging. The SeaThru algorithm provided a robust foundation for image correction, while the MIMO-UNet and reverse model expanded the capabilities to real-time processing and synthetic data generation. The release of the custom BlueRealm dataset and the transparency of the StreamUR method aims to foster further research and development in underwater computer vision. By providing high-quality training data and validated algorithms, we hope to support the community in advancing the state of the art in this field.

Despite the significant progress made, there are areas for future improvement and exploration. Firstly, the accuracy of depth estimation methods can be improved, further enhancing the effectiveness of image correction algorithms. Additionally, for ensuring broader applicability, the proposed method can be further evaluated in more diverse underwater environments, including turbid and low-light conditions considering also additional quantitative evaluation criteria, as for example numerical deviations of the reconstructed hue values of the color chart.

Acknowledgements

This work was supported in part by the iSEAu EU H2020 MSCA-IF project under Grant 101030367, in part by the RAMONES FET Proactive project under Grant 101017808, and in part by the SANTORY program funded by the Hellenic Foundation for Research and Innovation under Grant 1850.

References

Akkaynak, D., Treibitz, T., 2018: A revised underwater image formation model. *IEEE/CVF International Conference in Computer Vision and Pattern Recognition (CVPR)*, 6723–6732.

Akkaynak, D., Treibitz, T., 2019: Sea-thru: A method for removing water from underwater images. *IEEE/CVF International Conference in Computer Vision and Pattern Recognition (CVPR),* 1682-1691.

Chen, Z., Wang, J., Huang, S., 2022: Underwater image enhancement based on histogram-equalization approximation using physics-based dichromatic modeling. *Sensors, 22*(6), 2168

Cho, S., Ji, S., Hong, J., Jung, S., Ko, S., 2021: Rethinking Coarse-to-Fine Approach in Single Image Deblurring. *IEEE/CVF International Conference in Computer Vision and Pattern Recognition (CVPR),* 4641-4650.

Ebner, M., Hansen, J., 2013: Depth map color constancy. *Bio-Algorithms and Med-Systems*, 9(4):167–177.

Godard, C., Mac Aodha, O., Firman, M., Brostow, G., 2019: Digging into Self-Supervised Monocular Depth Estimation. *IEEE/CVF International Conference on Computer Vision (ICCV),* 3828-3838.

He, K., Sun, J., Tang, X., 2010: Single image haze removal using dark channel prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 33*(12), 2341-2353.

Ke, B., Obukhov, A., Huang, S., Metzger, N., Daudt, R. C., Schindler, K., 2023: Repurposing diffusion-based image generators for monocular depth estimation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9492-9502.

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

Li, C., Guo, C., Ren, W., Cong, R., Hou, J., Kwong, S., 2020: Underwater scene prior inspired deep underwater image and video enhancement. *Pattern Recognition*, 98, 107038.

Li, J., Skinner, K. A., Eustice, R. M., Johnson-Roberson, M., 2017: WaterGAN: Unsupervised generative network to enable real-time color correction of monocular underwater images. *IEEE Robotics and Automation letters*, 3(1), 387-394.

Ranftl, R., Bochkovskiy, A., Koltun, V., 2021: Vision transformers for dense prediction. *IEEE/CVF International Conference on Computer Vision (ICCV)*, 12179-12188

Spanos, S., Ntouskos, V., Karantzalo, K., 2024: Training-Free Multispectral Demosaicing with Deep Image Priors. *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*.

Wang, Q., Xie, C., Li, Y., Chen, S., Guo, J., 2023: PUGAN: Physical Model-Guided Underwater Image Enhancement Using GAN with Dual-Discriminators. *IEEE Transactions on Image Processing*, 32, 2234-2245.

Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E., 2004: Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4), 600–612.

Wang, Z., Shen, L., Xu, M., Yu, M., Wang, K., Lin, Y., 2023: Domain Adaptation for Underwater Image Enhancement. *IEEE Transactions on Image Processing*, 32, 1442-1457.

Wen, J., Cui, J., Zhao, Z., Yan, R., Gao, Z., Dou, L., Chen, B. M., 2023: SyreaNet: A Physically Guided Underwater Image Enhancement Framework Integrating Synthetic and Real Images. *IEEE International Conference on Robotics and Automation (ICRA)*, 5177-5183.

Appendix

We describe in detail here the main steps of our implementation of the SeaThru underwater image correction method, closely following the steps introduced in (Akkaynak & Treibitz, 2019).

Depth Map Estimation

A crucial component of the SeaThru method is the accurate estimation of depth for each pixel in the image, known as the range (or depth) map. This range map is essential for modeling the attenuation and backscatter of light in water, which are dependent on the distance between the camera and the objects in the scene. To generate these range maps, we utilized a neural network for monocular depth estimation as described in (Ke *et al.*, 2023). The neural network offers a method to estimate depth from a single image, which is particularly advantageous because obtaining depth information about a scene typically requires specialized equipment or algorithms such as Structure-from-Motion (SfM), both of which are generally difficult to acquire and implement. This neural network is designed to predict relative depth values within an image, rather than absolute depth measurements in meters. The steps involved in generating the range map include:

- **Image Preprocessing**: Each underwater image is preprocessed to enhance features that are critical for depth estimation. This includes techniques such as adaptive histogram equalization to improve contrast.
- **Depth Estimation**: The preprocessed images are fed into the neural network, which outputs a relative depth map. This map indicates the relative distances of objects in the scene but lacks absolute scale.
- **Scaling Depth Values**: To obtain an absolute depth map, we manually specify the minimum and maximum distances in the scene. These values are used to scale the relative depth map, providing an estimation of depth in meters.
- **Range Map Refinement**: The initial depth estimates are refined using additional techniques such as bilateral filtering to smooth out noise while preserving edges. This ensures that the range map is accurate and suitable for further processing in the SeaThru method.
- **Further Refinement:** We noticed that the generated range maps are more accurate if we use the image with the backscatter removed, so we generated a range map for each image, we utilized this range map to calculate the backscatter and remove it and then, we fed the monocular depth estimation network with the backscatter removed image, produced a range map again and used this new range map to recalculate the backscatter and to the rest of our algorithm.

Backscatter Removal

Backscatter removal is a critical step in the SeaThru method, aiming to eliminate the light scattered by particles in the water column that degrades the quality of underwater images. This scattered light, or backscatter, significantly reduces the contrast and color fidelity of underwater photographs.

The backscatter component B_c can be modeled as:

$$I_c = D_c + B_c,$$

where I_c is the observed image for the color channel c (R, G, or B), D_c represents the direct signal from the scene that has undergone attenuation and B_c is the backscatter component.

The SeaThru method estimates the backscatter component by leveraging the darkest pixels in the image, which are assumed to be areas with minimal direct signal. These pixels are used to infer the backscatter as follows:

Dark Pixel Identification: The algorithm searches for the darkest pixels in the image, which are likely to be areas with little to no direct signal (e.g., shadows or very dark regions). This is inspired by the Dark Channel Prior (DCP) used for haze removal in atmospheric images (He *et al.*, 201) but adapted to utilize known depth information from the range map.

Depth Interval Partitioning: The range map is divided into 5 evenly spaced depth intervals. Within each interval, the darkest 1% of the RGB triplets are identified. The values of these dark pixels are considered to provide an estimate of the backscatter for that specific depth range.

Backscatter Model: The relationship between backscatter and depth is modeled using the equation:

$$B_c = B_\infty(1 - e^{-\beta_c^B z}) + J'_c\, e^{-\beta_c^D z}.$$

Non-linear Least Squares Fitting: The algorithm employs non-linear least squares fitting to estimate the parameters. This fitting process minimizes the difference between the observed backscatter and the model's prediction. The parameters are subject to specific bounds to ensure physical plausibility: B_∞ , J' are in the range [0, 1] and β_c^B, β_c^D are in the range [0, 5]

Backscatter Removal: Once the backscatter component B_c is estimated, it is subtracted from the observed image I_c to obtain the direct signal D_c:

$$D_c = I_c - B_c.$$

Attenuation Coefficient Estimation

Estimating attenuation coefficients is a key step in the SeaThru method for accurately correcting underwater images. This involves modeling how light is absorbed and scattered as it travels through water, which varies with depth and wavelength.

The direct signal D_c for a given color channel c (R, G, or B) can be described by the equation:

$$D_c = J_c\, e^{-\beta_c^D(z) z},$$

where J_c is the true scene radiance without attenuation, β_c^D(z) is the depth-dependent attenuation coefficient and z is the depth.

Empirical results suggest modeling β_c^D(z) as a two-term exponential function, namely:

$$\beta_c^D(z) = ae^{bz} + ce^{dz},$$

where *a, b, c, d* are parameters to be estimated.

Estimating an Illumination Map

The true image J_c is related to the direct signal Dc through the equation:

$$J_c = D_c\, e^{\beta_c^D(z) z}.$$

Therefore, the depth-dependent attenuation coefficient can be estimated as: $\frac{log(J_c) - log(D_c)}{z} = \beta_c^D(z)$.

Using the estimated local illumination $E_c = \frac{J_c}{D_c}$, the attenuation coefficient can be approximated as: $\beta_c^D(z) = -\frac{log(E_c)}{z}$.

The SeaThru method uses Local Space Average Color (LSAC) to estimate the illumination map as described in (Ebner &Hansen, 2013):

$$a'_c(x,y) = \frac{1}{|N_e(x,y)|} \sum_{(x',y') \in N_e(x,y)} a_c(x',y'), \text{ with}$$

$$a_c(x,y) = D_c(x,y)p + a'_c(x,y)(1-p),$$

where p controls the locality of the neighborhood. $N_e(x',y')$ is the neighbourhood of (x,y) of 4-connected pixels neighbouring the pixel such that $|\,|z(x,y) - z(x',y')|\,| \leq \epsilon$ and $\hat{E}_c = fa_c$ where f is a constant controlling global illumination.

To refine the estimate, the algorithm minimizes the reconstruction error of depths: $\hat{z} = -\frac{log(\hat{E}_c)}{\beta_c^D(z)}$ by finding the values of a,b,c,d that minimize:

$$min_{\beta_c^D(z)} \parallel z - \hat{z} \parallel.$$

The algorithm employs non-linear least squares fitting with parameter constraints to ensure decaying exponentials: a, c are in the range [0, ∞] and b, d are in the range [−∞, 0].

To counteract overestimation from the illumination map, a multiplier l is used.

Image Reconstruction

The final step in the SeaThru method is image reconstruction, which aims to restore the true colors and clarity of the underwater images. This involves several processes, including color correction, white balancing, and ensuring the uniformity of the results. The final reconstructed image Jc is obtained by combining the corrected direct signal and the estimated attenuation model:

$$J_c = (I_c - B_c)e^{\beta_c^D(z) z},$$

where I_c is the observed image, B_c is the backscatter component, and $\beta_c^D(z)$ is the depth-dependent attenuation coefficient.

White Balancing – Image Finishing:

White balancing is crucial for ensuring that the colors in the reconstructed image are accurate and consistent with what would be seen in air. The SeaThru method originally used the Gray World Hypothesis for white balancing, which assumes that the average color in the scene is grey. However, to ensure the uniformity and consistency of the reconstructed images, we experimented with different white balancing methods and selected the final images based on their visual quality and uniformity. This involved comparing the results from different methods and choosing the one that provided the best balance between color accuracy and visual appeal, as well as the consistency between images of the same dataset.

The final reconstructed images are validated qualitatively by comparing them with reference images that contain a McBeth color chart and adjusting the parameters as needed to improve accuracy. This iterative process ensures that the final images are of high quality and suitable for further analysis and processing.