Learning from the Field: Physically-Based Deep Learning to Advance Robot Vision in Underwater Environments

by

Katherine A. Skinner

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Robotics)
in the University of Michigan
2019

Doctoral Committee:

Associate Professor Matthew Johnson-Roberson, Chair
Professor Ryan M. Eustice
Associate Professor Brian Hopkinson, University of Georgia
Associate Professor Emily Mower Provost
Assistant Professor Ram Vasudevan

"The living ocean drives planetary chemistry, governs climate and weather, and otherwise provides the cornerstone of the life-support system for all creatures on our planet, from deep-sea starfish to desert sagebrush. That's why the ocean matters. If the sea is sick, we'll feel it. If it dies, we die. Our future and the state of the oceans are one."

— Dr. Sylvia Earle

Katherine A. Skinner
kskin@umich.edu
ORCID iD: 0000-0003-4775-5040

To my parents

ACKNOWLEDGEMENTS

I have many people to thank for their generous support and guidance throughout my graduate research career. First, thank you to my Ph.D. advisor, Professor Matt Johnson-Roberson, for providing me with many years of mentorship and support. I appreciate your constant encouragement and all of the advice you have given me. Thank you to the members of my doctoral committee – Professor Ryan Eustice, Professor Brian Hopkinson, Professor Emily Provost, and Professor Ram Vasudevan – for your guidance and insightful comments. I appreciate all of your time and effort throughout this process.

Thank you to my labmates in the Deep Robot Optical Perception (DROP) lab. We have had some great adventures over the years. I am grateful for all of your help in making field work go as smoothly as possible. Beyond that, I am thankful to have colleagues that I can also call my friends. I would also like to thank the members of the Ford Center for Autonomous Vehicles (FCAV). I have really enjoyed my collaborations with FCAV and have learned a lot from working with you.

I feel very lucky to have had such a unique and wonderful experience in the first cohort of Robotics students at the University of Michigan. Thank you to the Robotics faculty and staff. You have created an amazing program and environment for learning and research. None of this would be possible without you. Thank you also to the Robotics students. I am so thankful for the wonderful community we have created. It has been an incredible privilege to be a part of the growth of the Robotics Institute at the University of Michigan.

Finally to my friends and family – thank you for everything that you do for me. I could not have gotten through this journey without all of you.

This work was supported in part by a fellowship from the Robotics Institute at the University of Michigan, by the Ford Motor Company via the Ford-UM Alliance under Award N022884, and by the National Science Foundation under Award 1452793.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

LIST OF APPENDICES

Appendix

LIST OF ABBREVIATIONS

3D	three-dimensional
ACFR	Australian Centre for Field Robotics
AUV	autonomous underwater vehicle
BCC	brightness constancy constraint
DNN	deep neural network
DOF	depth of field
DROP	Deep Robot Optical Perception Laboratory
DVL	Doppler velocity log
GAN	generative adversarial network
GPS	global positioning systems
GPU	graphics processing unit
HIMB	Hawaii Institute of Marine Biology
ICP	Iterated Closest Point
IMU	inertial measurement unit
LIDAR	Light Detection and Ranging
LReLU	leaky rectified linear unit
MHL	Marine Hydrodynamics Laboratory
MLA	microlens array
NCEI	National Centers for Environmental Information
ORB	Oriented FAST and Rotated BRIEF
PSP	Pyramid Scene Parsing

RGB red, green, and blue color

RGB-D color and depth

RMSE root mean square error

ROV remotely operated vehicle

RTE radiance transfer equation

SAS synthetic aperture sonar

SGBM Semi-Global Block Matching

SIFT scale invariant feature transform

SIMS Sydney Institute of Marine Science

SLAM simultaneous localization and mapping

SURF speeded up robust features

LIST OF SYMBOLS

α	hyperparameter
β	hyperparameter
δ	threshold
η	beam attenuation coefficient
γ	hyperparameter
κ	scaling parameter
λ	wavelength
$\mathcal{D}$	discriminator
$\mathcal{G}$	generator
$\mathcal{L}$	loss
$\mathcal{M}$	surfel
ν	vignetting model parameter
Φ	structural similarity
ϕ	hyperparameter
ψ	hyperparameter
ρ	water depth from surface
θ	viewing direction
ξ	3D ray direction
ζ	noise vector
B	backscattered light
B_∞	veiling light

C	corrected color image
c	color channel
D	disparity image
d	distance function
E	error function
I	image
j	index
k	camera viewpoint
K_d	attenuation coefficient due to diffuse downwelling
L	left view
L_*	radiance gained from random scattering events
L_0	true scene radiance
L_r	received radiance
M	image mask
P	3D point
p	2D point
q_a	acutance constraint
q_c	contrast constraint
R	right view
r	normalized radius per pixel
S	surface
T	direct transmission
V	vignetting
w	hyperparameter
x	image pixel
z	range

ABSTRACT

Field robotics refers to the deployment of robots and autonomous systems in unstructured or dynamic environments across air, land, sea, and space. Robust sensing and perception can enable these systems to perform tasks such as long-term environmental monitoring, mapping of unexplored terrain, and safe operation in remote or hazardous environments. In recent years, deep learning has led to impressive advances in robotic perception. However, state-of-the-art methods still rely on gathering large datasets with hand-annotated labels for network training. For many applications across field robotics, dynamic environmental conditions or operational challenges hinder efforts to collect and manually label large training sets that are representative of all possible environmental conditions a robot might encounter. This limits the performance and generalizability of existing learning-based approaches for robot vision in field applications.

This thesis focuses on developing approaches for unsupervised learning to advance perceptual capabilities of robots in underwater environments. The underwater domain presents unique environmental conditions to robotic systems that exacerbate the challenges to perception for field robotics. To address these challenges, the proposed approaches leverage physics-based models and cross-disciplinary knowledge about the physical environment and the data collection process to provide constraints that relax the need for ground truth labels in learning-based frameworks. This leads to a hybrid model-based, data-driven solution. Experiments are carried out using data from real field experiments to validate the proposed approaches. Although this work specifically focuses on the underwater domain, its aim is to present novel frameworks for using cross-disciplinary knowledge and physics-based models of environmental conditions in computer vision and unsupervised learning contexts to work towards robust perception systems for deploying robotic platforms in natural and unstructured environments.

CHAPTER 1

Introduction

1.1 Motivation

Humans have sailed the sea and exploited its resources for thousands of years. However, the first scientific expedition to methodically explore the oceans began in 1872 aboard the HMS *Challenger*. Over the course of 1000 days, the *Challenger* sailed around the world, collecting observations of marine life, measurements of water properties, and samples of seafloor sediment [1]. Since then, scientific research vessels have covered much more of our oceans and waterways (Fig. 1.1).

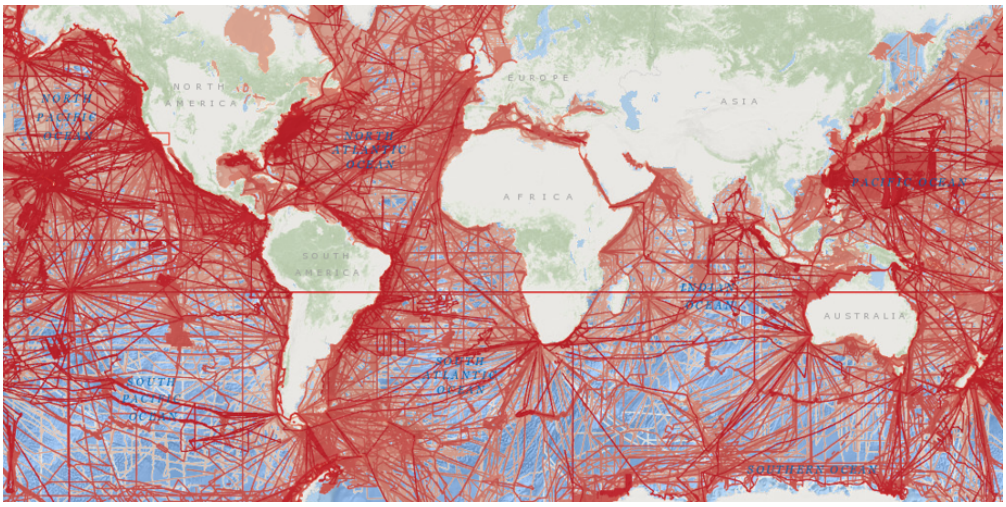


Figure 1.1: Map of tracklines from the National Centers for Environmental Information (NCEI) Marine Trackline Geophysical database showing coverage of ocean expeditions to collect geophysical observations of the sea between 1939-2018 (present) [2].

Throughout this time, techniques and technology for collecting observations of marine environments have drastically improved. Figure 1.2 shows two bathymetric maps of the

Campeche Escarpment in the Gulf of Mexico. The base map was compiled from hydrographic charts and satellite altimetry and, until 2013, was the highest resolution bathymetric map available for the site. The top map was collected with a multibeam sonar during a survey from the R/V *Falkor* in 2013. The depths recorded range from $400m$ (red) to $3700m$ (blue) [3].

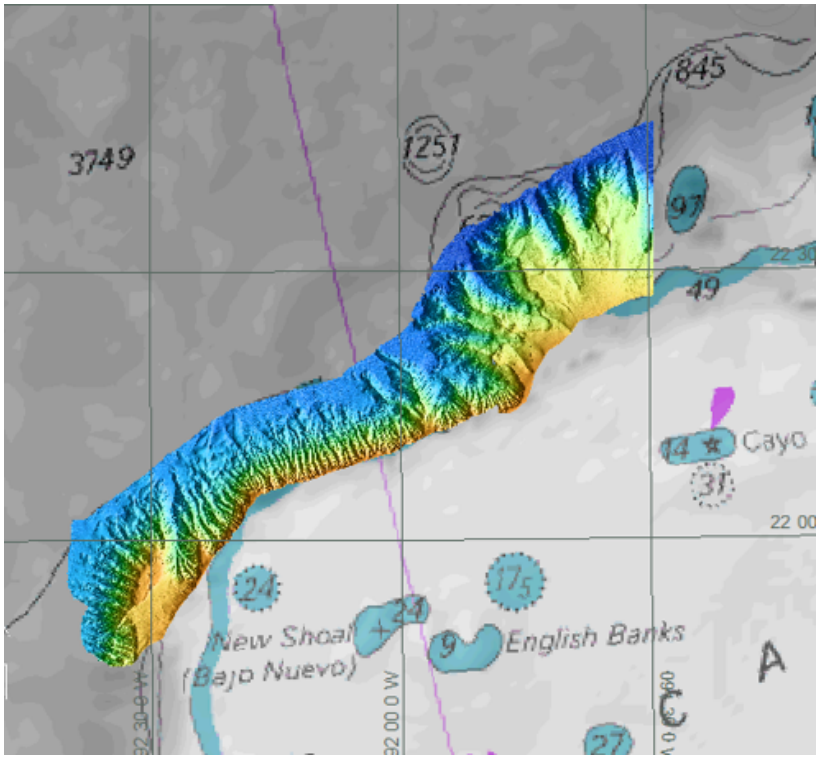


Figure 1.2: Map of the Campeche Escarpment in the Gulf of Mexico. The bottom layer is the best bathymetric map available for the area pre-2013. The top layer shows a multibeam survey gathered during a 2013 expedition from R/V *Falkor*. The depths of the multibeam data range from $400m$ (red) to $3700m$ (blue) [3].

Acoustic sensors such as multibeam echosounders are now commonly used for mapping seafloor bathymetry. Due to their long range underwater, sonar instruments can be mounted on ships to conduct surveys from the sea surface. The resolution of sonar systems is determined by the frequency and range to the seafloor [4] [5]. Side scan sonars operating at higher frequencies can achieve higher resolution, but their operational range is limited due to attenuation through the water column [6] [7]. Synthetic aperture sonars (SASs) use state-of-the-art post-processing techniques to achieve up to $4cm$ resolution at a range of over $100m$. However, these systems must be used from a moving platform over a static scene, and their current cost inhibits widespread application [8].

Still, sonar technology cannot gather color information of a scene, which is important for

many applications. For example, color can be used as an indicator of coral reef health [9]. Figure 1.3 shows a side-by-side comparison of a coral reef site near Lizard Island, Australia before and after a bleaching event [10]. Healthy coral (right) depends on a symbiotic relationship with algae that lives within the coral structure. The algae protects the coral and contributes to the variety of colors that coral reef systems can exhibit. Environmental stressors, such as changes in sea surface temperature, can disrupt this symbiotic relationship, causing the coral to expel the algae. This process leaves behind the "bleached" bone white structure of the coral itself (left), which is now more fragile and vulnerable to disease. While there can be structural differences between healthy and bleached coral, the most drastic difference is the change in color from the absence of the algae [9]. Thus, in order to better monitor coral reef systems, it is important to gather observations of both structure and color of these sites.

Figure 1.3: Images taken from the same site in Lizard Island, Australia in March 2016 (right) and May 2016 (left) show changes in structure and color of coral reefs before and after a bleaching event [10].

For applications such as this where color is important, optical sensors can be used to obtain high resolution color imagery of subsea environments. However, optical sensors are subject to the attenuation of electromagnetic signals through the water column, which limits their ideal operational range. Generally, maximum visibility range is less than $25m$, with the world record range recorded at $79m$ [11]. In practice, the ideal range for imaging surveys is approximately $2 - 4m$ due to visibility constraints. Furthermore, the aperture is limited by practical size, which also limits the field-of-view, or the observable area per frame. Thus, imaging surveys of the seafloor cannot be carried out from surface vessels in deeper waters.

(a) Diver rig survey at Hog Reef off the coast of Bermuda, conducted as a collaboration between Woods Hole Oceanographic Institution (WHOI), University of Michigan (UM), and University of Georgia (UGA).

(b) Deployment of the Deep Robot Optical Perception Laboratory (DROP) Lab's Iver autonomous underwater vehicle (AUV) during a research cruise aboard the R/V *Falkor*.

Figure 1.4: Robotic platforms equipped with high resolution, color cameras can be deployed to conduct systematic imaging surveys of the seafloor.

Instead, many fields in marine science and engineering rely on underwater robotic platforms equipped with imaging sensors to provide high resolution, colored views of the seafloor. At depths less than $40m$, divers can carry diver rigs, or platforms equipped with cameras and navigation sensors to gather imagery of the seafloor (Fig. 1.4a). Remotely operated vehicles (ROVs) and autonomous underwater vehicles (AUVs) can be deployed in depths up to $11000m$ (Fig. 1.4b). Robotic systems are capable of carrying out large-scale surveys efficiently and effectively, gathering terabytes of imagery during a single mission.

1.2 Underwater Image Formation

While recent decades have seen great advances in perceptual capabilities of robotic systems, the subsea environment presents unique challenges to optical imaging that are not present on land.

Figure 1.5 shows an abstraction of underwater light propagation between a camera system and a target scene. As a photon of light travels through the water column, it interacts with surrounding water molecules and particulate matter and can be either scattered or completely absorbed. Absorption is wavelength-dependent and significantly contributes to wavelength-dependent attenuation of light underwater [12]. The rate of attenuation also depends on the properties of the water; it varies between fresh and salt water, coastal and ocean water, and even seasonally [13]. Backscattering occurs when the light is scattered back towards the camera before it reaches the scene. This causes a haze effect, often referred to

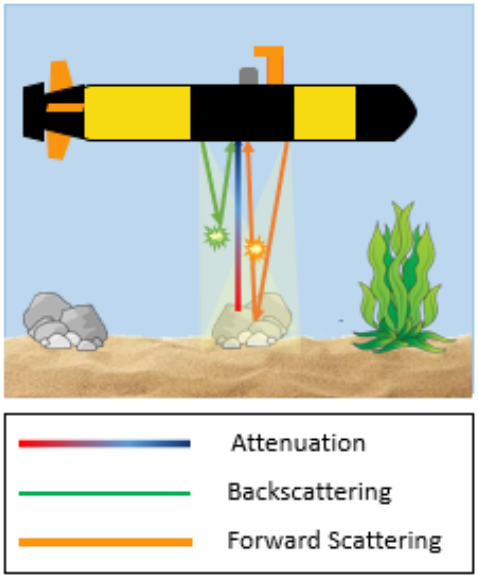


Figure 1.5: Abstraction of an underwater imaging survey subject to several water column effects, including attenuation, backscattering, and forward scattering.

as veiling light. The backscattered signal is the main contributor to image degradation of underwater images [14] [15]. Forward scattering also occurs when light that is scattered away from the camera gets rescattered along the line of sight, leading to a blurring of the scene. However, its contribution to attenuation and image degradation is negligible compared to absorption and backscattering, so it is frequently omitted [14]. Figure 1.6 provides examples of raw underwater images in different bodies of water to demonstrate the range of colors and quality that result from water column effects on underwater light propagation.

(a) Hog Reef, Bermuda (b) Port Royal, Jamaica (c) Pure water tank

Figure 1.6: Sample underwater images from various test sites.

Light propagation through a scattering media can be described by the radiance transfer equation (RTE) [16] [17]. For a homogeneous water column, assuming no inelastic scattering or emission, a compact form of the RTE is given by [17]:

$$L_r(\rho;\xi;\lambda) = L_0(\rho;\xi;\lambda)e^{-\eta(\lambda)z} + \frac{L_*(\rho;\xi;\lambda)e^{-K_d(\lambda)z\cos\theta}}{\eta(\lambda) - K_d(\lambda)\cos\theta}[1 - e^{-[\eta(\lambda)-K_d(\lambda)\cos\theta]z}] \tag{1.1}$$

L_0 is true radiance of the target scene, L_* describes radiance gained from random scattering events, and L_r is the received radiance subject to water column effects. ξ is the three-dimensional (3D) ray direction, λ is wavelength, z is range between the camera and the scene, ρ is water depth from surface (vertical range only), K_d is the attenuation coefficient due to diffuse downwelling, η is the beam attenuation coefficient representing loss of photons due to absorption and scattering, and θ denotes viewing direction.

More intuitively, the above equation describes the received radiance, L_r, as the combination of directly transmitted light, T, subject to attenuation, and backscattered light, B, which carries no information about the scene:

$$L_r = T + B \tag{1.2}$$

This model for underwater light propagation and its application to perception of underwater robotic systems will be further explored throughout this work.

1.3 Computer Vision Challenges

The complex, nonlinear process of underwater image formation presents several challenges for the field of computer vision. On land, real-time depth sensing has been an incredible boon to perception of mobile robots [18], with advances in sensor modalities such as stereo cameras, Light Detection and Ranging (LIDAR), and more recently color and depth (RGB-D) sensors [19]. Unfortunately, the latter two – which include time-of-flight range sensing and pattern projection structured light approaches – are still quite limited in their success underwater due to attenuation of electromagnetic signals through the aqueous medium [20].

In recent decades, stereo cameras have been popular sensing systems for underwater robots. With calibrated stereo pairs, high resolution images can be aligned with depth information to compute large-scale photomosaic maps or metrically accurate 3D reconstructions [21]. However, degradation of images due to range-dependent underwater lighting effects can hinder these approaches. The first step in traditional stereo vision pipelines is to detect distinct feature patches within an image. Common feature descriptors, including scale invariant feature transform (SIFT) [22], speeded up robust features (SURF) [23], and Oriented FAST and Rotated BRIEF (ORB) [24], rely on image contrast, which is reduced in

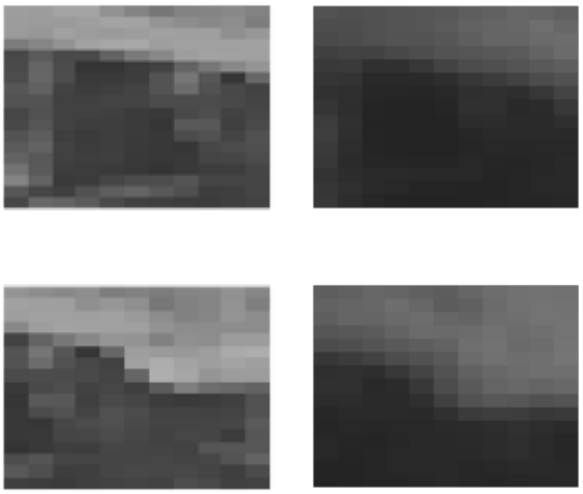
underwater images due to attenuation of light. This makes it challenging to detect distinct features in raw underwater imagery. Figure 1.7 shows sample feature patches from in air images (left) and corresponding patches of the same features imaged underwater (right).

Once features are detected, feature matching is attempted across stereo image pairs (e.g. left-right images). With matched features and calibration, it is then possible to compute disparity, or depth maps, for each view. As more stereo views are gathered across a vehicle trajectory, feature matches can also be made across multiple viewpoints, such as when the vehicle crosses back over a feature it has seen before. These loop closures can provide strong constraints on the relative pose of the vehicle throughout its trajectory, as well as on the geometry of the scene. One assumption that enables loop closures on land is the assumption that one feature imaged from different viewpoints will appear with the same intensity across each image. This is known as the brightness constancy constraint (BCC) [25]. Although this constraint is violated under several circumstances – e.g. shadowing, specular objects – in practice, it is sufficient to enable loop closures even across large changes in vehicle trajectories. Similarly, state-of-the-art methods for real-time 3D reconstruction also rely on photometric consistency as a cue for tracking and fusing image patches from view-to-view.

These assumptions do not hold underwater and, in practice, may not be sufficient for loop closure detection. Figure 1.8 shows an illustration of a case where assumptions of photometric consistency fail: If the same scene point $P(j)$ is imaged from different viewpoints k and $(k+1)$ with a large range disparity between viewpoints, the same feature will appear with different radiance in the resulting images, $I_{k,L}$ and $I_{k+1,L}$, due to range-dependent attenuation.

One approach to restoring underwater images is to use image processing techniques, such as histogram equalization, to stretch the effective contrast of the image. This can improve feature detection and matching and often results in visually appealing images. However, image processing techniques have no knowledge of the physical process of underwater image formation; thus they do not account for range-dependent effects to restore photometric consistency across different viewpoints.

Instead, this thesis proposes novel methods for underwater image restoration that incorporate the physics-based model of underwater light propagation to improve accuracy and consistency of corrected underwater images. Still, this is an ill-posed problem. Scene geometry and object radiance are interdependent, and automating solutions for these parameters is further confounded by other factors such as the imaging sensor and water properties.



(a) In air (b) Underwater

Figure 1.7: Sample image patches of the same features imaged in air (left) and underwater (right). Note the reduced contrast in the feature patches taken from underwater images.

1.4 Operational Challenges

It is also important to acknowledge challenges that arise in practice when working in marine environments. Marine operations are expensive, time-consuming, and sometimes dangerous to carry out. When using diver rigs, diver safety is a critical consideration for system design and mission planning. For boat operations, hiring a boat and crew with all necessary equipment can be expensive. Deep ocean expeditions can require being at sea for weeks at a time. Unpredictable weather can make conditions unsafe for sailing or vehicle deployment. Wildlife may also interfere with divers or vehicles. In practice, these challenges leave a limited window of opportunity for data collection, ultimately limiting the amount of data that can be collected.

Another factor to consider is that WiFi and global positioning systems (GPS) do not operate underwater due to attenuation of electromagnetic signals. GPS is commonly used for localization on land as it provides highly accurate absolute position. Underwater robots must rely on other sensors – such as Doppler velocity logs (DVLs), inertial measurement units (IMUs), or camera systems – to determine relative position and orientation. Furthermore, ground truth position is only available when the vehicle is on the surface, where it is possible to get GPS data.

In general, ground truth data for other measurements is difficult to gather in subsea environments. Ground truth is important for validating developed methods or for supervised training for deep learning approaches. As mentioned previously, high resolution depth sensors used on land do not operate well underwater. Thus it is difficult to gather metrically

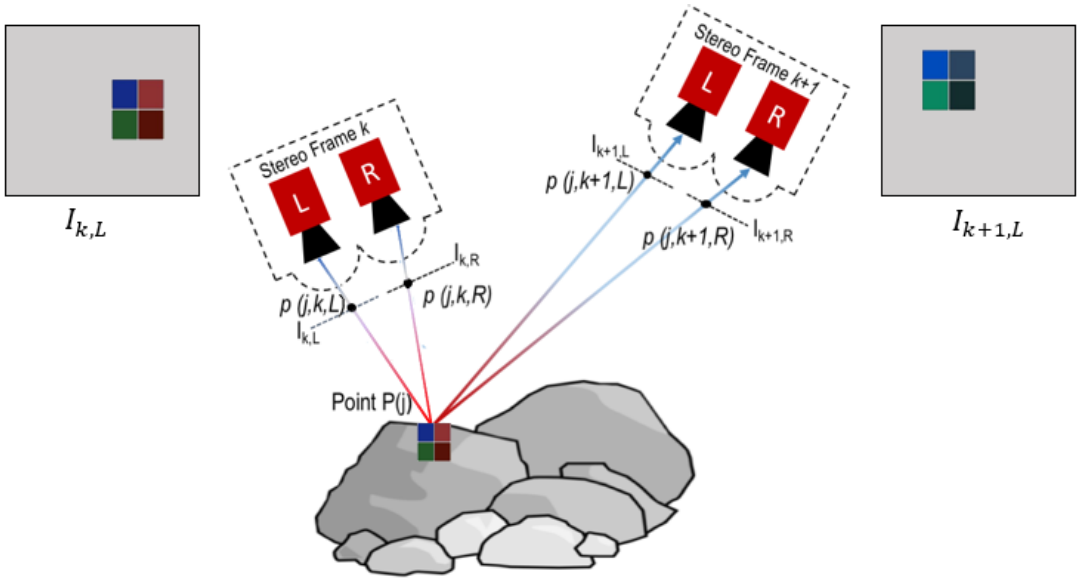


Figure 1.8: Point $P(j)$ is observed in each of the stereo camera pairs, k and $(k+1)$, both in the left and right cameras.

accurate ground truth of scene geometry. Ground truth color of submerged structures is also impractical to gather, unless the structure can be removed from the marine environment for evaluation on land.

1.5 Model-based vs. Data-driven

Two key approaches have been developed to address challenges to robotic perception: model-based approaches and data-driven approaches.

Traditional model-based approaches have provided key insights and robust solutions for many important problems in robotics. These approaches rely on developing rules and structure to describe complex phenomena in natural environments. Well-formulated models limit the complexity of a problem, leading to an efficient, interpretable solution. However, it is often not possible to model every complex process in the real world. Thus, model-based approaches can incur error in assumptions made to reduce the complexity of the modeled space.

In recent years, deep learning, a data-driven approach, has led to many advances in perceptual capabilities of robotic systems. A deep neural network (DNN) can learn to model complex functions with high accuracy. Currently, state-of-the-art methods are supervised, relying on hand annotated labels for network training. When developed and trained on labeled benchmark datasets, DNNs work exceptionally well, outperforming traditional hand-designed and model-based approaches by wide margins on tasks including object de-

tection, semantic segmentation, and instance segmentation. However, many open problems in field robotics are currently ill-suited for supervised learning solutions. Due to operational challenges and the dynamic nature of natural environments, it is impossible to collect and annotate a single training dataset that is representative of all possible environmental conditions a robot might encounter. This limits the performance and generalizability of existing learning-based approaches for many scenarios across field robotics.

This thesis explores the integration of traditional model-based approaches within deep learning frameworks to enable unsupervised learning for robotic perception in underwater environments. Incorporating physics-based models and information from cross-disciplinary fields into data-driven frameworks can provide structure and constraints that relax the need for ground truth labels. In particular, methods developed throughout this thesis integrate knowledge of physics-based models of underwater light propagation, stereo camera geometry and image processing with a focus on designing unsupervised and self-supervised learning approaches that do not require ground truth. Ultimately, this leads to a hybrid model-based, data-driven solution to address challenges to perception for underwater robotic systems.

1.6 Problem Statement

This thesis seeks solutions to the following challenges to perception in underwater robotics:

1. Assumptions used in state-of-the-art computer vision algorithms for terrestrial applications break down underwater due to water column effects.

2. The physics-based model for underwater image formation has been well-studied but there are challenges to implementing an automated solution to account for water column effects in computer vision pipelines. Underwater image restoration is an ill-posed problem that involves reversing effects of a complex physical process with prior knowledge of water column characteristics for a specific survey site, as well as information about scene geometry.

3. Operational challenges in underwater robotics limit the amount and quality of data that can be collected, in practice. Additionally, ground truth of both geometry and color are particularly difficult to gather.

1.7 Contributions

The main objective of this dissertation is to improve perception for underwater robotic systems. To achieve this objective, this dissertation presents the following contributions:

1. Development of a deep learning framework that learns the underlying physics-based model of underwater light propagation in order to simulate and correct for water column effects on color in monocular underwater imagery. (Chapter 2)

2. Development of a method to learn from both structure and color of stereo underwater imagery to perform online depth estimation and color correction of raw underwater stereo images. (Chapter 3)

3. Development of a framework to perform state-of-the-art real-time 3D reconstruction using passive optical sensors in underwater environments. (Chapter 4)

Appendices A, B, and C provide further details of datasets, software, and robotic and sensing systems developed in support of this dissertation, respectively.

Work presented throughout this thesis, and related work, has been published in the following publications:

Katherine A. Skinner, Junming Zhang, Elizabeth Olson and Matthew Johnson-Roberson, "UWStereoNet: Unsupervised learning for depth estimation and color correction of underwater stereo imagery." In Proceedings of the *IEEE International Conference on Robotics and Automation (ICRA)*, Montreal, Canada, 2019.

Elizabeth Olson, Corina Barbalata, **Katherine A. Skinner** and Matthew Johnson-Roberson, "Deep learning for disparity estimation of underwater images with synthetic data." In Proceedings of the *IEEE/MTS OCEANS Conference and Exhibition*, Charleston, USA, 2018.

Jie Li*, **Katherine A. Skinner***, Ryan Eustice and Matthew Johnson-Roberson, "WaterGAN: Unsupervised generative network to enable real-time color correction of monocular underwater images." In *IEEE Robotics and Automation Letters (RA-L)*, 2017. *The authors contributed equally to this work.

Eduardo Iscar, **Katherine A. Skinner** and Matthew Johnson-Roberson, "Multi-view 3D reconstruction in underwater environments: evaluation and benchmark." In Proceedings of the *IEEE/MTS OCEANS Conference and Exhibition*, Anchorage, USA, September 2017.

Katherine A. Skinner and Matthew Johnson-Roberson, "Underwater image dehazing with a light field camera." In Proceedings of the *IEEE Conference on Computer Vision and*

Pattern Recognition – Workshops (CVPR-W), Honolulu, USA, 2017.

Katherine A. Skinner, Eduardo Iscar Ruland and Matthew Johnson-Roberson, "Automatic color correction for 3D reconstruction of underwater scenes." In Proceedings of the *IEEE International Conference on Robotics and Automation (ICRA)*, Singapore, 2017.

Katherine A. Skinner and Matthew Johnson-Roberson, "Towards real-time underwater 3D reconstruction with plenoptic cameras." In Proceedings of the *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, Korea, 2016.

CHAPTER 2

Unsupervised Generative Network to Enable Real-time Color Correction of Monocular Underwater Images

2.1 Introduction

While recent decades have seen great advancements in vision capabilities of underwater platforms, the subsea environment presents unique challenges to perception that are not present on land. As described previously, range-dependent lighting effects such as attenuation cause exponential decay of light between the imaged scene and the camera. This attenuation acts at different rates across wavelengths and is strongest for the red channel in oceanic environments. As a result, raw underwater images appear relatively blue or green compared to the true color of the scene as it would be imaged in air. Simultaneously, light is added back to the sensor through scattering effects, causing a haze effect across the scene that reduces the effective resolution. Restoration of underwater images involves reversing effects of a complex physical process with prior knowledge of water column characteristics for a specific survey site. Additionally, image restoration efforts must account for range- and wavelength-dependencies of water column effects in order to restore photometric constistency and brightness constancy.

In recent years, advances in neural networks have enabled end-to-end modeling of complex nonlinear systems. Yet deep learning has not become as commonplace subsea as it has for terrestrial applications. One challenge is that many deep learning structures require large amounts of training data, typically paired with labels or corresponding ground truth sensor measurements. Gathering large sets of underwater data with depth information is challenging in deep sea environments; obtaining ground truth of the true color of a natural subsea scene is also an open problem.

Rather than gathering training data, this chapter presents a novel approach, WaterGAN, a generative adversarial network (GAN) [26] that uses real unlabeled underwater images to

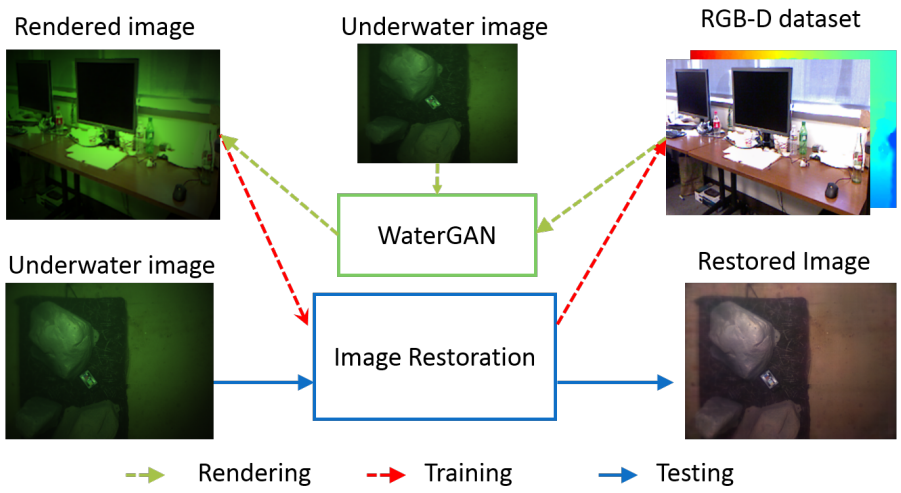


Figure 2.1: Flowchart displaying both the WaterGAN and color correction networks. WaterGAN takes input in-air color and depth (RGB-D) data and a sample set of underwater images and outputs synthetic underwater images aligned with the in-air RGB-D data. The color correction network uses this aligned data for training. For testing, a real monocular underwater image is input and a corrected image and relative depth map are output.

learn a realistic representation of water column properties of a particular survey site. WaterGAN takes in-air images and depth maps as input and generates corresponding synthetic underwater images as output. This dataset with corresponding depth data, in-air color, and synthetic underwater color can then supplant the need for real ground truth depth and color in the training of a color correction network. This chapter also presents a novel network to perform monocular color correction based on generated data. The color correction network takes as input raw unlabeled underwater images and outputs restored images that appear as if they were taken in air.

This chapter is organized as follows: §2.2 presents relevant prior work; §2.3 gives a detailed description of the technical approach; §2.4 presents an experimental setup to validate our proposed approach; §2.5 provides results and a discussion of these results; lastly, §2.6 concludes the chapter.

2.2 Background

Prior work on compensating for effects of underwater image formation has focused on explicitly modeling this physical process to restore underwater images to their true color. Jordt et al. used a modified Jaffe-McGlamery model with parameters obtained through prior experiments [27] [28]. However, attenuation parameters vary for each survey site depending on water composition and quality. Bryson et al. used an optimization approach to estimate

water column and lighting parameters of an underwater survey to restore the true color of underwater scenes [29]. However, this method requires detailed knowledge of vehicle configuration and the camera pose relative to the scene. Our method instead learns to model these effects using a deep learning framework without explicitly encoding vehicle configuration parameters.

Approaches that make use of the gray world assumption [30] or histogram equalization are common preprocessing steps for underwater images and may result in improved image quality and appearance. However, as such methods have no knowledge of range-dependent effects, resulting images of the same object viewed from different viewpoints may appear with different colors. Work has been done to enforce the consistency of restored images across a scene [31], but these methods require dense depth maps. Our preliminary work aimed to relax this requirement using an underwater bundle adjustment formulation to estimate the parameters of a fixed attenuation model and the 3D structure simultaneously [32], but such approaches require a fixed image formation model and handle unmodeled effects poorly. Our novel approach presented here can perform restoration with individual monocular images as input, and learns the relative structure of the scene as it corrects for the effects of range-dependent attenuation.

Several methods have addressed range-dependent image dehazing by estimating depth through developed or statistical priors on attenuation effects [33]–[35]. More recent work has focused on leveraging the success of deep learning techniques to estimate parameters of the complex physical model. Shin et al. [36] developed a deep learning pipeline that achieves state-of-the-art performance in underwater image dehazing using simulated data with a regression network structure to estimate parameters for a fixed restoration model. Our method incorporates real field data in a generative network to learn a realistic representation of environmental conditions for raw underwater images of a specific survey site.

WaterGAN is structured as a GAN. GANs have shown success in generating realistic images in an unsupervised pipeline that only relies on an unlabeled set of images of a desired representation [26]. A standard GAN generator receives a noise vector as input and generates a synthetic image from this noise through a series of convolutional and deconvolutional layers [37]. Recent work has shown improved results by providing an input image to the generator network, rather than just a noise vector. Shrivastava et al. provided a simulated image as input to their network, SimGAN, and then used a refiner network to generate a more realistic image from this simulated input [38]. To extend this idea to the domain of underwater image restoration, we also incorporate easy-to-gather in-air color and depth (RGB-D) data into the generator network since underwater image formation is range-dependent. Sixt et al. proposed a related approach in RenderGAN, a framework for

generating training data for the task of tag recognition in cluttered images [39]. RenderGAN uses an augmented generator structure with augment functions modeling known characteristics of their desired images, including blur and lighting effects. RenderGAN focuses on a finite set of tags and classification as opposed to a generalizable transmission function and image-to-image mapping.

2.3 Methodology

This chapter presents a two-part technical approach to produce a pipeline for image restoration of monocular underwater images. Figure 2.1 shows an overview of the full pipeline. WaterGAN is the first component of this pipeline, taking as input in-air RGB-D images and a sample set of underwater images to train a generative network adversarially. This training procedure uses unlabeled raw underwater images of a specific survey site, assuming that water column effects are mostly uniform within a local area. This process produces rendered underwater images from in-air RGB-D images that conform to the characteristics of the real underwater data at that site. These synthetic underwater images can then be used to train the second component of our system, a novel color correction network that can compensate for water column effects in a specific location in real-time.

2.3.1 Generating Realistic Underwater Images

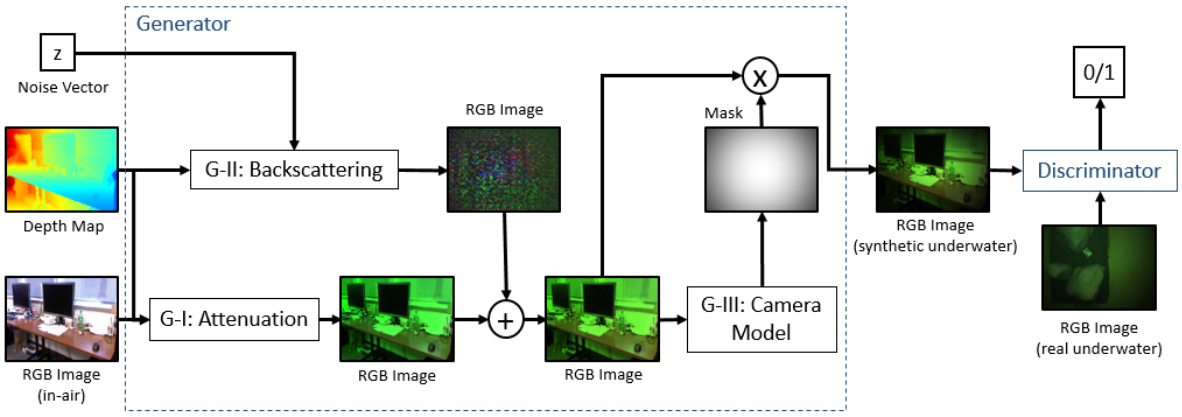


Figure 2.2: WaterGAN: The generative adversarial network (GAN) for generating realistic underwater images with similar image formation properties to those of unlabeled underwater data taken in the field.

WaterGAN is structured as a generative adversarial network, which has two networks training simultaneously: a generator, $\mathcal{G}$, and a discriminator, $\mathcal{D}$ (Fig. 2.2). In a standard GAN [26] [37] the generator input is a noise vector ζ, which is projected, reshaped, and

propagated through a series of convolution and deconvolution layers. The output is a synthetic image, $\mathcal{G}(\zeta)$. The discriminator receives as input the synthetic images and a separate dataset of real images, x, and classifies each sample as real (1) or synthetic (0). The goal of the generator is to output synthetic images that the discriminator classifies as real. Thus in optimizing $\mathcal{G}$, we seek to maximize

$$log(\mathcal{D}(\mathcal{G}(\zeta)). \tag{2.1}$$

The goal of the discriminator is to achieve high accuracy in classification, minimizing the above function, and maximizing $\mathcal{D}(x)$ for a total value function of

$$log(\mathcal{D}(x)) + log(1 - \mathcal{D}(\mathcal{G}(\zeta))). \tag{2.2}$$

The generator of WaterGAN features three main stages, each modeled after a component of underwater image formation: attenuation (G-I), backscattering (G-II), and the camera model (G-III). The purpose of this structure is to ensure that generated images align with the RGB-D input, such that each stage does not alter the underlying structure of the scene itself, only its relative color and intensity. Additionally, this formulation ensures that the network is using depth information in a realistic manner. This is necessary as the discriminator does not have direct knowledge of the depth of the scene. The remainder of this section describes each stage in detail.

G-I: Attenuation

The first stage of the generator, G-I, accounts for range-dependent attenuation of light. The attenuation model is a simplified formulation of the Jaffe-McGlamery model [28] [40],

$$G_1 = I_{air} e^{-\eta(\lambda)z}, \tag{2.3}$$

where I_{air} is the input in-air image, or the initial irradiance before propagation through the water column, z is the range from the camera to the scene, and η is the wavelength-dependent attenuation coefficient estimated by the network. The wavelength, λ, is discretized into three color channels. G_1 is the final output of G-I, the final irradiance subject to attenuation in the water column. Note that the attenuation coefficient is dependent on water composition and quality, and varies across survey sites. To ensure that this stage only attenuates light,

as opposed to adding light, and that the coefficient stays within physical bounds, η is constrained to be greater than 0. All input depth maps and images have dimensions of 48×64 for training model parameters. This training resolution is sufficient for the size of our parameter space and preserves the aspect ratio of the full-size images. Note that the generator can still achieve full resolution output for final data generation, as explained below. Depth maps for in-air training data are normalized to the maximum underwater survey altitude expected. Given the limitation of optical sensors underwater, it is reasonable to assume that this value is available.

G-II: Scattering

As a photon of light travels through the water column, it is also subjected to scattering back towards the image sensor. This creates a characteristic haze effect in underwater images and is modeled by

$$B = B_{\infty}(\lambda)(1 - e^{-\eta(\lambda)z}), \tag{2.4}$$

where B_{∞} is a scalar parameter dependent on wavelength. Stage G-II accounts for scattering through a shallow convolutional network. To capture range-dependency, a 48×64 depth map is input into the generator, along with a 100-length noise vector. The noise vector is projected, reshaped, and concatenated to the depth map as a single channel 48×64 mask. To capture wavelength-dependent effects, this input is copied for three independent convolution layers with kernel size 5×5. This output is batch normalized and put through a final leaky rectified linear unit (LReLU) with a leak rate of 0.2. Each of the three outputs of the distinct convolution layers are concatenated together to create a $48 \times 64 \times 3$ dimension mask. Since backscattering adds light back to the image, and to ensure that the underlying structure of the imaged scene is not distorted from the RGB-D input, we add this mask, M_2, to the output of G-I:

$$G_2 = G_1 + M_2. \tag{2.5}$$

G-III: Camera Model

Lastly, we model vignetting and the sensor response function. Vignetting produces a shading pattern around the borders of an image due to effects from the lens, and it can be modeled by [41]:

$$V = 1 + \nu_1 r^2 + \nu_2 r^4 + \nu_3 r^6, \tag{2.6}$$

where r is the normalized radius per pixel from the center of the image, such that $r = 0$ in the center of the image and $r = 1$ at the boundaries. The constants ν_1, ν_2, and ν_3 are model parameters estimated by the network. The output mask has dimensions of the input images, and G_2 is multiplied by $M_3 = \frac{1}{V}$ to produce a vignetted image G_3,

$$G_3 = M_3 G_2. \tag{2.7}$$

As described in [41], the parameters can be constrained by:

$$(\nu_3 \geq 0) \wedge (4\nu_2^2 - 12\nu_1\nu_3 < 0). \tag{2.8}$$

Finally, we assume a linear sensor response function, which has a single scaling parameter κ [29], with the final output given by

$$G_{out} = \kappa G_3. \tag{2.9}$$

Discriminator

For the discriminator of WaterGAN, we adopt the convolutional network structure used in [37]. The discriminator takes an input image $48 \times 64 \times 3$, real or synthetic. This image is propagated through four convolutional layers with kernel size 5×5 with the image dimension downsampled by a factor of two, and the channel dimension doubled. Each convolutional layer is followed by LReLUs with a leak rate of 0.2. The final layer is a sigmoid function and the discriminator returns a classification label of (0) for synthetic or (1) for a real image.

Generating Image Samples

After training is complete, we use the learned model to generate image samples. For image generation, we input in-air RGB-D data at a resolution of 480×640 and output synthetic underwater images at the same resolution. To maintain resolution and preserve the aspect ratio, the vignetting mask and scattering image are upsampled using bicubic interpolation before applying them to the image. The attenuation model is not specific to

the resolution.

2.3.2 Underwater Image Restoration Network

Note: This subsection was written by Jie Li and is reproduced here for completeness.

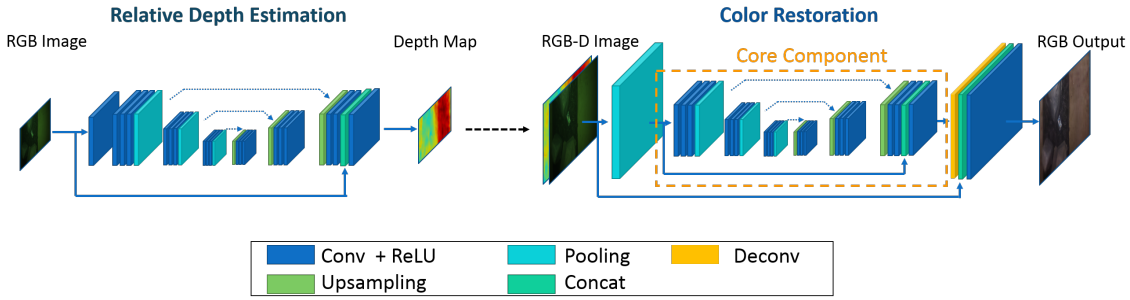


Figure 2.3: Network architecture for color correction. The first stage of the network takes a synthetic (training) or real (testing) underwater image and learns a relative depth map. The image and depth map are then used as input for the second stage to output a restored color image as it would appear in air.

To achieve real-time monocular image color restoration, we propose a two-stage algorithm using two fully convolutional networks that train on the in-air RGB-D data and corresponding rendered underwater images generated by WaterGAN. The architecture of the model is depicted in Fig. 2.3. A depth estimation network first reconstructs a coarse relative depth map from the downsampled synthetic underwater image. Then a color restoration network conducts restoration from the input of both the underwater image and its estimated relative depth map.

We propose the basic architecture of both network modules based on a state-of-the-art fully convolutional encoder-decoder architecture for pixel-wise dense learning, SegNet [42]. A new type of non-parametric upsampling layer is proposed in SegNet that directly uses the index information from corresponding max-pooling layers in the encoder. The resulting encoder-decoder network structure has been shown to be more efficient in terms of training time and memory compared to benchmark architectures that achieve similar performance. SegNet was designed for scene segmentation, so preserving high frequency information of the input image is not a required property. In our application of image restoration, however, it is important to preserve the texture level information for the output so that the corrected image can still be processed or utilized in other applications such as 3D reconstruction or object detection. Inspired by recent work on image restoration and denoising using neural networks [43][44], we incorporate skipping layers on the basic encoder-decoder structure to

compensate for the loss in high frequency components through the network. The skipping layers are able to increase the convergence speed in network training and to improve the fine scale quality of the restored image, as shown in Fig. 2.7. More discussion will be given in §2.5.

As shown in Fig. 2.3, in the depth estimation network, the encoder consists of 10 convolution layers and three levels of downsampling. The decoder is symmetric to the encoder, using non-parametric upsampling layers. Before the final convolution layer, we concatenate the input layer with the feature layers to provide high resolution information to the last convolution layer. The network takes a downsampled underwater image of $56 \times 56 \times 3$ as input and outputs a relative depth map of $56 \times 56 \times 1$. This map is then upsampled to 480×480 and serves as part of the input to the second stage for color correction.

The color correction network module is similar to the depth estimation network. It takes an input RGB-D image at the resolution of 480×480, padded to 512×512 to avoid edge effects. Although the network module is a fully convolutional network and changing the input resolution does not affect the model size itself, increasing input resolution demands larger computational memory to process the intermediate forward and backward propagation between layers. A resolution of 256×256 would reach the upper bound of such an encoder-decoder network trained on a $12GB$ graphics processing unit (GPU). To increase the output resolution of our proposed network, we keep the basic network architecture used in the depth estimation stage as the core processing component of our color restoration net, as depicted in Fig. 2.3. Then we wrap the core component with an extra downsampling and upsampling stage. The input image is downsampled using an averaging pooling layer to a resolution of 128×128 and passed through the core process component. At the end of the core component, the output is then upsampled to 512×512 using a deconvolution layer initialized by a bilinear interpolation filter. Two skipping layers are concatenated to preserve high resolution features. In this way, the main intermediate computation is still done in relatively low resolution. We were able to use a batch size of 15 to train the network on a $12GB$ GPU with this resolution. For both the depth estimation and color correction networks, a Euclidean loss function is used. The pixel values in the images are normalized between 0 to 1.

2.4 Experiments

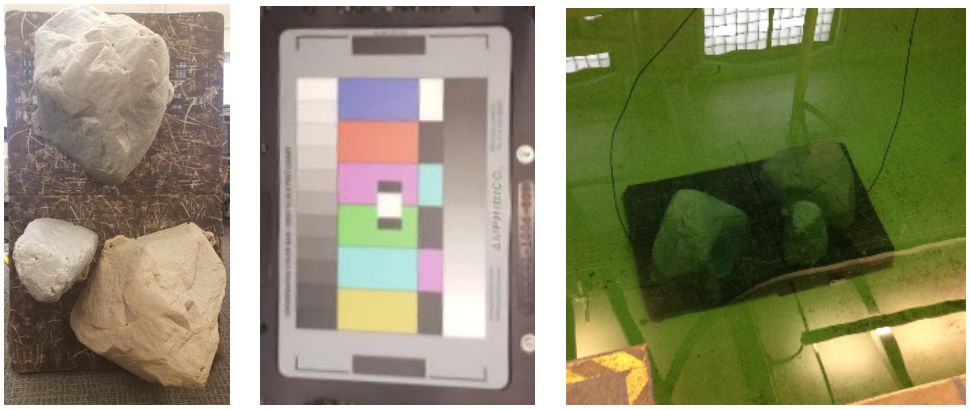
2.4.1 Experimental Setup

We evaluate the proposed method using datasets gathered in both a controlled pure water test tank and from real scientific surveys in the field. As input in-air RGB-D for all

experiments, we compile four indoor Kinect datasets (B3DO [45], UW RGB-D Object [46], NYU Depth [47] and Microsoft 7-scenes [48]) for a total of 15000 RGB-D images. See Appendix A for further details of datasets used throughout this work.

2.4.2 Artificial Testbed

The first survey is done using a 4 ft $\times$ 7 ft man-made rock platform submerged in a pure water test tank at University of Michigan's Marine Hydrodynamics Laboratory (MHL). A color board is attached to the platform for reference (Fig. 2.4). A total of over 7000 underwater images are compiled from this survey.



(a) Rock platform (b) Color board (c) MHL test tank

Figure 2.4: (a) An artificial rock platform and (b) a diving color board are used to provide ground truth for controlled imaging tests in (c) a pure water tank to gather the Marine Hydrodynamics Laboratory (MHL) dataset.

2.4.3 Field Tests

One field dataset was collected in Port Royal, Jamaica, at the site of a submerged city containing both natural and man-made structure. These images were collected with a hand-held diver rig. The water quality across this site was relatively murky. For the experiments, we compile a dataset consisting of 6500 images from a single dive. The maximum depth from the seafloor is approximately $1.5m$. Another field dataset was collected at a coral reef system near Lizard Island, Australia in relatively clear water [49]. The data was gathered with the same diver rig and we assumed a maximum depth of $2.0m$ from the seafloor. We compile a total number of 6083 images from the multi-dive survey within a local area.

2.4.4 Network Training

For each dataset, we train the WaterGAN network to model a realistic representation of raw underwater images from a specific survey site. The real samples are input to WaterGAN's discriminator network during training, with an equal number of in-air RGB-D pairings input to the generator network. Table 2.1 shows training parameters used for training WaterGAN. We train WaterGAN on a Titan X (Pascal) with a batch size of 64 images and a learning rate of 0.0002. The estimated attenuation parameters are initialized by the ideal attenuation coefficients found in prior experimentation [27]. The remaining weights are initialized from a truncated normal distribution. Through experiments, we found 10 epochs to be sufficient to render realistic images for input to the color correction network for the Port Royal and Lizard Island datasets. We trained for 25 epochs for the MHL dataset. Once a model is trained, it can generate an arbitrary amount of synthetic data. For the experiments, we generate a total of 15000 rendered underwater images for each model (MHL, Port Royal, and Lizard Island), which corresponds to the total size of the compiled RGB-D dataset.

Table 2.1: Training parameters for WaterGAN network.

Parameter	**Value**
Momentum	0.5
Batch size	64
Learning rate	0.0002
Training image height	48
Training image width	64

Next, this data is used to train the color correction network with the generated images and corresponding in-air RGB-D images. This set is split into a training set with 12000 images and a validation set with 3000 images. The networks are trained from scratch for both the depth estimation network and image restoration network on a Titan X (Pascal) GPU. Table 2.2 shows training parameters used for training the underwater image restoration network. The depth estimation network is trained for 20 epochs with a batch size of 50, a base learning rate of $1e^{-6}$, and a momentum of 0.9. The color correction network is trained using a two-level training strategy. For the first level, the core component is trained with an input resolution of 128×128, a batch size of 20, and a base learning rate of $1e^{-6}$ for 20 epochs. Then the whole network is trained at a full resolution of 512×512, with the parameters in core components initialized from the first training step. The full resolution model is trained for 10 epochs with a batch size of 15 and a base learning rate of $1e^{-7}$. Results are discussed in §2.5 for all three datasets.

Table 2.2: Training parameters for underwater image restoration network.

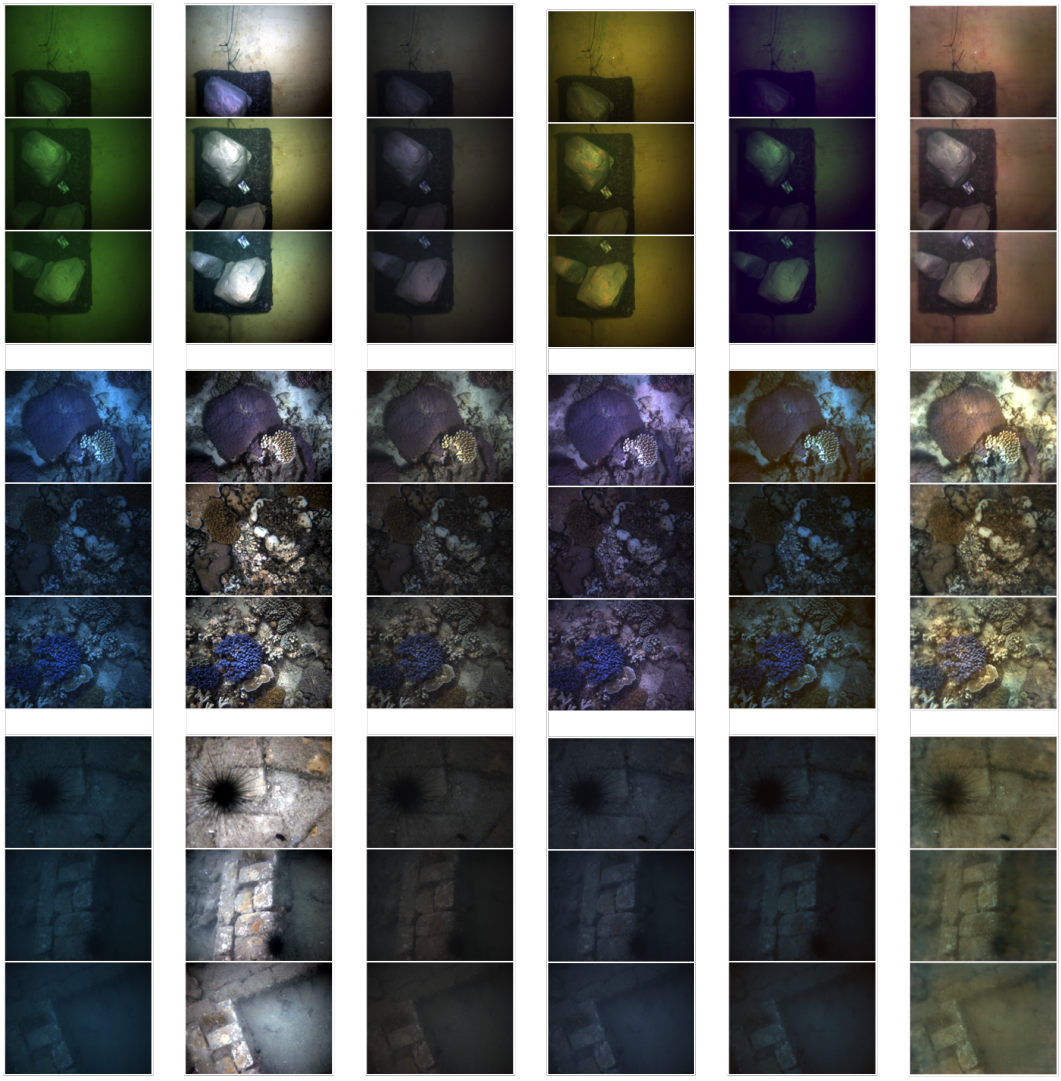
Training Stage	**Parameter**	**Value**
Depth	Momentum	0.9
Depth	Batch size	50
Depth	Epochs	20
Depth	Learning rate	$1e^{-6}$
Color	Batch size	20
Color	Epochs	20
Color	Resolution	128x128
Color	Learning rate	$1e^{-6}$
Full	Batch size	15
Full	Epochs	10
Full	Resolution	512x512
Full	Learning Rate	$1e^{-7}$

2.5 Results & Discussion

To evaluate the image restoration performance in real underwater data, we present both qualitative and quantitative analysis for each dataset. We compare the developed approach to image processing approaches that are not range-dependent, including histogram equalization and normalization with the gray world assumption. We also compare the results to a range-dependent approach based on a physical model, the modified Jaffe-McGlamery model (Eqn. 3) with ideal attenuation coefficients [27]. Lastly, we compare the proposed method to Shin et al.'s deep learning approach [36], which implicitly models range-dependent information in estimating a transmission map.

Qualitative results are given in Fig. 2.5. Histogram equalization looks visually appealing, but it has no knowledge of range-dependent effects so corrected color of the same object viewed from different viewpoints appears with different colors. The proposed method shows more consistent color across varying views, with reduced effects of vignetting and attenuation compared to the other methods. We demonstrate these findings across the full datasets in the following quantitative evaluation.

We present two quantitative metrics for evaluating the performance of our color correction: color accuracy and color consistency. For accuracy, we refer to the color board attached to the submerged rock platform in the MHL dataset. Table 2.3 shows the Euclidean distance of intensity-normalized color in RGB-space for each color patch on the color board compared to an image of the color board in air. These results show that our method has the lowest error for blue, red, and magenta. Histogram equalization has the lowest error for cyan, yellow and green recovery, but our method still outperforms the remaining methods



(a) Raw (b) HE (c) GW (d) JM (e) [36] (f) WG

Figure 2.5: Results showing color correction on the Marine Hydrodynamics Laboratory (MHL), Lizard Island, and Port Royal datasets (from top to bottom). Each column shows (a) raw underwater images, and corrected images using (b) histogram equalization (HE), (c) normalization with the gray world assumption (GW), (d) a modified Jaffe-McGlamery model (Eqn. 3) with ideal attenuation coefficients (JM), (e) Shin et al.'s deep learning approach ([36]), and (f) WaterGAN (WG).

for cyan and yellow.

To analyze color consistency quantitatively, we compute the variance of the intensity-normalized pixel color for each scene point that is viewed across multiple images. In detail, for each scene point, we determine the set of corresponding image features visible across multiple images. We then normalize the pixel values by intensity in RGB-space. Finally, we compute the variance of the normalized pixel colors, where lower variance indicates higher consistency in color across multiple viewpoints of an individual feature. Table 2.4 shows the mean variance of these points. WaterGAN shows the lowest variance across each color channel. This consistency can also be seen qualitatively in Fig. 2.5.

Table 2.3: Color correction accuracy based on Euclidean distance of intensity-normalized color in RGB-space for each method compared to the ground truth in-air color board.

	Raw	**Hist. Eq.**	**Gray World**	**Mod. J-M**	**Shin[36]**	**Prop. Meth.**
Blue	0.3349	0.2247	0.2678	0.2748	0.1933	**0.1431**
Red	0.2812	0.0695	0.1657	0.2249	0.1946	**0.0484**
Mag.	0.3475	0.1140	0.2020	0.2980	0.1579	**0.0580**
Green	0.3332	**0.1158**	0.1836	0.2209	0.2013	0.2132
Cyan	0.3808	**0.0096**	0.1488	0.3340	0.2216	0.0743
Yellow	0.3599	**0.0431**	0.1102	0.2265	0.2323	0.1033

Table 2.4: Variance of intensity-normalized color of single scene points imaged from different viewpoints.

	Raw	**Hist. Eq.**	**Gray World**	**Mod. J-M**	**Shin[36]**	**Prop. Meth.**
Red	0.0073	0.0029	0.0039	0.0014	0.0019	**0.0005**
Green	0.0011	0.0021	0.0053	0.0019	0.0170	**0.0007**
Blue	0.0093	0.0051	0.0042	0.0027	0.0038	**0.0006**

The trained network is also validated on the testing set of synthetic data and the validation results are given in Table 2.5. For both color and depth, root mean square error (RMSE) is used as the error metric. These results show that the trained network is able to invert the model encoded by the generator.

In terms of the computational efficiency, the forward propagation for depth estimation takes $0.007s$ on average and the color correction module takes $0.06s$ on average, which is efficient for real-time applications.

Table 2.5: Validation error in pixel value is given in RMSE in RGB-space. Validation error in depth is given in RMSE (m).

Dataset	Red	Green	Blue	Depth RMSE
Synth. MHL	0.052	0.033	0.055	0.127
Synth. Port Royal	0.060	0.041	0.031	0.122
Synth. Lizard	0.068	0.045	0.035	0.103

It is important to note that the depth estimation network recovers accurate relative depth, not necessarily absolute depth. This is due to the scale ambiguity inherent to the monocular depth estimation problem. Figure 2.6 shows qualitative results of monocular depth estimation using the proposed method. To evaluate the depth estimation in real underwater images, the estimated depth is compared to the depth reconstructed from stereo images available for the MHL dataset in a normalized manner, ignoring the pixels where no depth is recovered from stereo reconstruction due to lack of overlap or feature sparsity. The RMSE of normalized estimated depth and the normalized stereo reconstructed depth is $0.11m$.

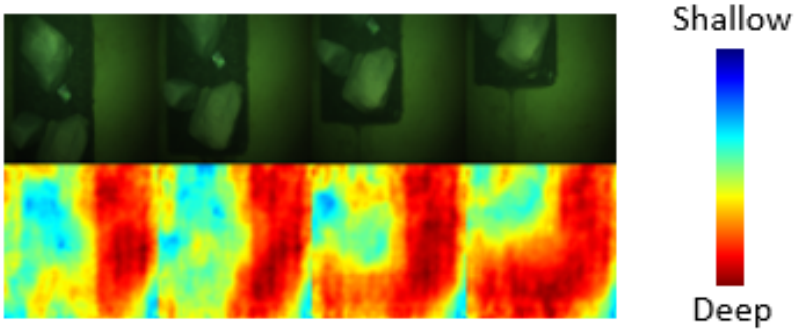


Figure 2.6: Monocular depth estimation output from the image restoration network, where red represents greater depths and blue represents shallower depths.

To evaluate the improvement in image quality due to skipping layers in the color correction network, the network is trained at the same resolution with and without skipping layers. For the first pass of core component training, the network without skipping layers takes around 30 epochs to reach a stable loss, while the proposed network with skipping layers takes around 15 epochs. The same trend holds for full model training, taking 10 and 5 epochs, respectively. Figure 2.7 shows a comparison of image patches recovered from both versions of the network. This demonstrates that using skipping layers helps to preserve high frequency information from the input image.

One limitation of the model is in the parameterization of the vignetting model, which

(a) Raw image patch (b) Restored image without skipping layers (c) Final output

Figure 2.7: Zoomed-in comparison of color correction results of an image with and without skipping layers.

assumes a centered vignetting pattern. This is not a valid assumption for the MHL dataset, so the restored images still show some vignetting though it is partially corrected. These results could be improved by adding a parameter that adjusts the center position of the vignetting pattern over the image. This demonstrates a limitation of augmented generators, more generally. Since they are limited by the choice of augmentation functions, augmented generators may not fully capture all aspects of a complex nonlinear model [39]. We introduce a convolutional layer into the augmented generator that is meant to capture scattering, but additional layers at this stage may capture more complex effects, such as caustic lighting patterns from sunlight in shallow water surveys.

See Appendix B for access to open-source software.

2.6 Conclusion

This chapter presented WaterGAN, a generative network for modeling underwater images from RGB-D in air. We showed a novel generator network structure that incorporates the process of underwater image formation to generate high resolution output images. We then presented a dense pixel-wise model learning pipeline for the task of color correction of monocular underwater images trained on RGB-D pairs and corresponding generated images. We evaluated the method on both controlled and field data to show qualitatively and quantitatively that the output is accurate and consistent across varying viewpoints. There are several promising directions for future work to extend this network. Here we train WaterGAN and the color correction network separately to simplify initial development of our methods. Combining these networks into a single network to allow joint training of image degradation simulation and image restoration would be a more elegant approach.

CHAPTER 3

Unsupervised Learning for Depth Estimation and Color Correction of Underwater Stereo Imagery

3.1 Introduction

Underwater stereo vision is a critical perception component for many marine robotic systems. With calibrated stereo pairs, high resolution images can be aligned with depth information to compute large-scale photomosaic maps or metrically accurate three-dimensional (3D) reconstructions [21]. Relative to other sensors, stereo cameras are compact, inexpensive, and capable of providing high resolution color imagery and depth of subsea scenes. Still, there are many challenges to deploying stereo camera systems in underwater environments due to the degradation of underwater images. As previously described, water column effects are range-dependent and wavelength-dependent, so image degradation increases with range from the camera, and it alters the ratio of color channels in resulting images. This can have severe consequences for traditional stereo vision algorithms, which rely on image contrast and brightness or photometric consistency across views to detect and match image features [25] [50] [22]. In order to develop a robust underwater stereo vision system, we must account for these water column effects to restore photometric consistency, or we must develop methods that are invariant to the inconsistencies seen across varying viewpoints of underwater scenes.

In recent years, deep learning has led to many advances in robotic perception. Applications of deep learning to the underwater domain have shown great potential to solve many open problems in the field [51] [52]. However, once again, there are unique challenges to applying these methods in underwater environments. First, it is particularly challenging to gather large training datasets with ground truth structure and color of underwater scenes. This motivates the development of unsupervised or self-supervised learning approaches. Second, data gathered in underwater environments is highly degraded. Due to underwater

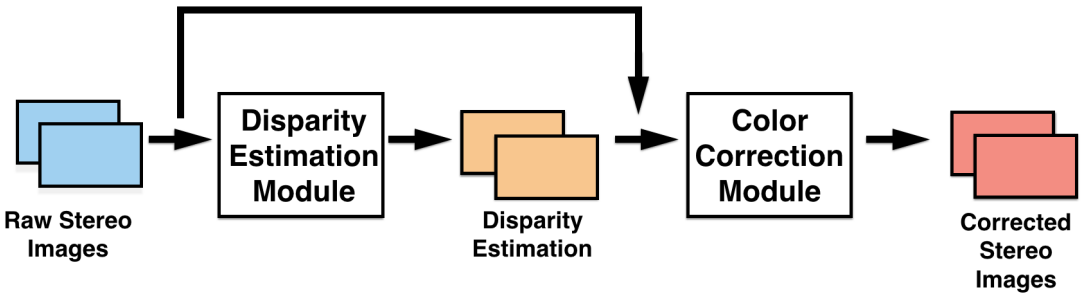


Figure 3.1: Overview of proposed network structure for simultaneous depth estimation and color correction from raw underwater stereo imagery.

lighting effects, images taken underwater can also vary widely depending on factors such as time-of-day and water properties. This makes it difficult to gather representative training data and to develop generalizable networks that work across a range of subsea scenes. One thing we can leverage is prior knowledge from traditional computer vision, image processing and underwater imaging to provide constraints for approaching this problem.

In this chapter, we develop a novel unsupervised network that addresses two challenging problems in underwater stereo vision: dense depth estimation and color correction of raw color underwater images. Our method exploits the process of underwater image formation, insights from image processing, as well as the geometry of stereo camera systems. To our knowledge, this is the first approach to develop a deep learning approach that estimates both dense depth and water column color correction directly from underwater stereo imagery. We present experiments on real underwater data collected at different field sites, with ground truth structure and color to provide both quantitative and qualitative evaluation of results. We show that our method improves upon traditional and state-of-the-art approaches. We also provide a discussion of insights gained on the application of deep learning to underwater stereo vision of robotic systems.

The remainder of this chapter is organized as follows: §3.2 includes relevant prior work, §3.3 details our technical approach, §3.4 presents experiments, §3.5 presents results and discusses these results, and, lastly, §3.6 summarizes conclusions and suggestions for future work.

3.2 Background

3.2.1 Learning from Stereo Imagery

Recent work for terrestrial applications has focused on learning dense depth maps directly from rectified stereo images [53] [54] [55] [56] [57]. These methods require dense ground truth

depth maps, which are difficult to gather underwater. We instead focus on developing an unsupervised network. Prior work on unsupervised disparity estimation has leveraged the constraints of stereo geometry with an image warping loss function [58] [59] [60]. For our disparity estimation network, we leverage a state-of-the-art architecture, DispSegNet [61]. DispSegNet uses a five-dimensional cost volume to estimate a coarse initial disparity. This initial disparity is then refined further using smoothness and semantic segmentation labels. We modify this network so that semantic segmentation is not required.

3.2.2 Underwater Image Restoration

One approach to restore underwater images is to use image processing techniques, such as histogram equalization, to stretch the effective contrast of the image. This can improve feature detection and matching and often results in visually appealing images. However, image processing techniques have no knowledge of the physical process of underwater image formation; thus they do not account for range-dependent effects to restore photometric consistency across different viewpoints. This makes these techniques unreliable for consistent color correction across changing viewpoints.

Our work proposes a network for underwater image restoration that incorporates the physical model of underwater light propagation. Prior work has incorporated knowledge of range-dependent water column effects to perform image restoration of monocular underwater images [33]. Other work has incorporated this model into simultaneous localization and mapping (SLAM) and 3D reconstruction frameworks [31] [29] [27] [32]. These approaches leverage the dense depth output from 3D reconstruction methods as input to the range-dependent model. Bryson et al. also incorporates information such as vehicle lighting configuration [29]. Our approach does not require pre-processing or full 3D reconstruction. We input only unlabeled raw stereo imagery and output dense disparity maps and restored images directly.

Recently, work has been done to provide further insight and experimental validation of traditional models of underwater light propagation that are commonly used for underwater image restoration [17] [62]. This has led to development of an updated model for underwater light propagation, which inspires the structure of our proposed network, discussed in more detail in §3.3.

3.2.3 Learning for Underwater Vision

State-of-the-art methods use deep learning architectures for monocular underwater image restoration. Many methods rely on synthetic underwater datasets that are augmented from

in-air datasets. These training sets are either constructed manually using a known physical model for underwater light propagation [36], or images are augmented within a learned pipeline [51] [52]. We train our network on real underwater images. To our knowledge, our work is the first to develop a learning-based framework for both dense depth estimation and color correction from raw underwater stereo imagery.

3.3 Methodology

Figure 3.1 shows an overview of our proposed network structure. The network is a modular, two-stage network. The first stage performs disparity estimation based on [61]. This stage takes input rectified raw color stereo images, I_L and I_R, and estimates disparity maps for each image, D_L and D_R. The second stage is a color correction module. This module takes in D_L and D_R and converts disparities to metric depth values using the stereo camera calibration. The depth is then concatenated with the raw stereo images and input to the color correction module. The color correction network outputs restored underwater stereo images, C_L and C_R. The following subsections describe further details of each network module.

3.3.1 Disparity Estimation

Note: This subsection was written by Junming Zhang and is reproduced here for completeness.

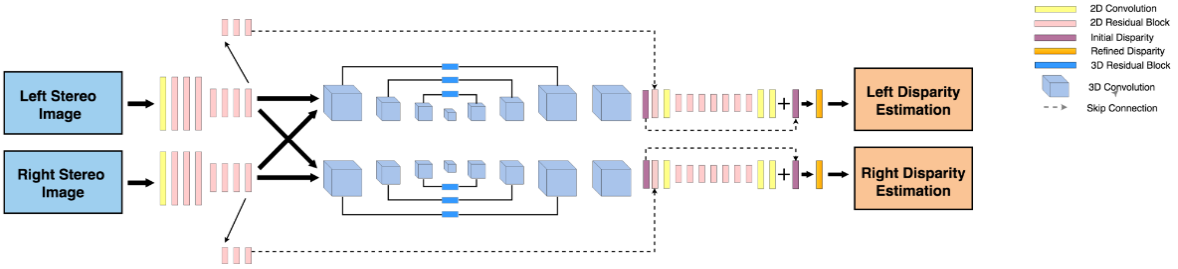


Figure 3.2: Network structure for disparity estimation module [61].

In this work we employ a Siamese network architecture based on DispSegNet [61] to learn dense disparity maps from input left and right images. This network structure was initially developed to refine disparity estimates based on ground truth semantic segmentation [61]. Here we modify it by removing the Pyramid Scene Parsing (PSP) module [63] in order to only perform unsupervised disparity estimation without reliance on ground truth semantic

segmentation. The network structure contains two branches, one for the left image and another for the right image. Each branch has a ResNet [64] structure and weights are shared across both branches. Feature extraction is first performed on the input stereo images to output a feature map 1/4 of the size of input image. The left and right output feature maps are concatenated to form a five-dimensional cost volume, one for each view, with dimensions: $BatchSize \times (MaxDisparity + 1) \times Height \times Width \times FeatureSize$. Three-dimensional convolution can be applied to these cost volumes to output a coarse disparity map as the initial estimate. This coarse estimate is then input into a refinement network to achieve more finegrained disparity estimation.

We use different losses for the initial and refinement stages of the disparity estimation process. The loss for the initial disparity estimate ($\mathcal{L}_{disp\_init}$) forces the model to estimate a coarse disparity map weighting all pixels equally, while the refinement loss ($\mathcal{L}_{disp\_ref}$) forces the model to focus on regions of high difficulty (low texture and high noise). The losses are defined as following:

$$Loss = \alpha_1 \mathcal{L}_{disp\_init} + \alpha_2 \mathcal{L}_{disp\_ref} \tag{3.1}$$

$$\mathcal{L}_{disp\_init} = \beta_1 \mathcal{L}_{disp\_warp} + \beta_2 \mathcal{L}_{consist} + \beta_3 \mathcal{L}_{reg} \tag{3.2}$$

$$\mathcal{L}_{disp\_ref} = \gamma_1 \mathcal{L}_{disp\_warp} + \gamma_2 \mathcal{L}_{consist} + \gamma_3 \mathcal{L}_{smooth} \tag{3.3}$$

where α_1 and α_2 are scalars that trade off between initial and refinement loss and β_1, β_2 and β_3 trade off between photometric loss ($\mathcal{L}_{disp\_warp}$), consistency loss ($\mathcal{L}_{consist}$) and regularization loss ($\mathcal{L}_{reg}$) respectively, each of which are described below. The γ scalars serve the same purpose for each respective loss in the $\mathcal{L}_{disp\_ref}$ refinement loss.

The key component of the total loss during unsupervised disparity estimation is the photometric warping loss. This loss leverages geometric constraints inherent to the stereo vision problem and is enabled by a differentiable bilinear sampler [58] [59]. To construct this loss, we use the estimated disparity map to warp the left image to the right image, and vice versa. After this warping, we use photometric reconstruction error that measures the visual difference between the warped image and the real image. Photometric loss is computed for both sides of the stereo pair. For the left side, it is defined as following:

$$\mathcal{L}_{disp\_warp} = \phi_1 \Phi(I_L, I_L^{'}) + \phi_2 |I_L - I_L^{'}| + \phi_3 |\nabla I_L - \nabla I_L^{'}| \tag{3.4}$$

where I_L is left input image, $I_L^{'}$ is the reconstructed left image from the right input image, ∇ is the first derivative and $\Phi()$ is structural similarity which is used to increase robustness against errors in ill-posed regions of the image. The ϕ scalars were set experimentally. The photometric loss for the right camera is symmetric.

With strict photometric loss during training, the predicted disparity map typically contains a high amount of noise. Regularization loss $\mathcal{L}_{reg}$ is used to enforce locally smooth results. Consistency loss $\mathcal{L}_{consist}$ enforces that given the reconstructed left image $I_L^{'}$, we can also warp it back to the right view $I_R^{''}$ using the right camera disparity map D_R. This additional loss forces the left and right disparity branches of the network to be coupled. This loss minimizes the visual difference between $I_R^{''}$ and I_R. After convergence the initial disparity still contains error in challenging regions. We locate these regions by finding the inconsistencies between the initial disparities of the left and right disparity maps [65]. This informs a smoothness loss that minimizes differences in these regions. This loss is only applied during refinement because it requires initialized disparity estimates. Note that we do not use the supervised segmentation loss. For additional details, please refer to the original paper [61].

3.3.2 Color Correction

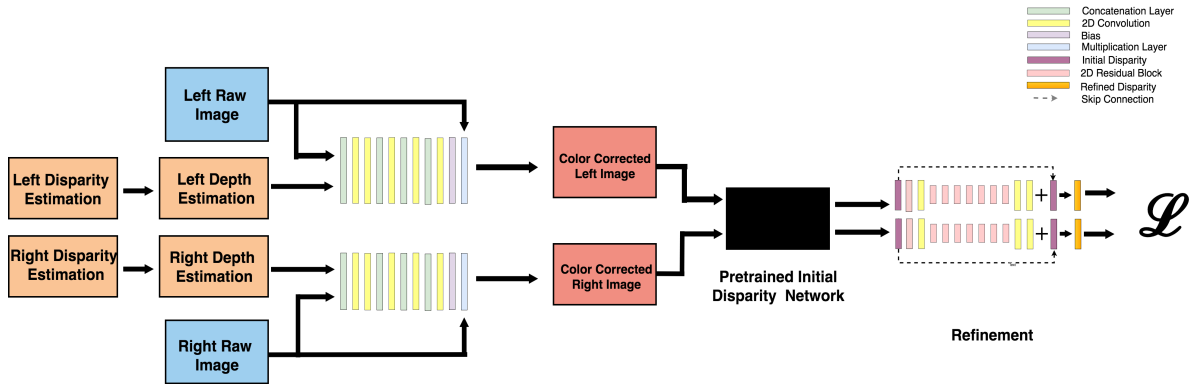


Figure 3.3: Network structure for color correction and disparity refinement module.

Figure 3.3 shows a detailed diagram of the color correction network module. The color correction network structure is motivated by a physics-based model of underwater image formation [28] [40]:

$$I(x) = L_0(x)e^{-\eta z(x)} + B_\infty(1 - e^{-\eta z(x)}) \tag{3.5}$$

where z is the range between the camera and the scene, η is an attenuation coefficient, x are spatial coordinates in the image, B_∞ is the veiling or environmental light, L_0 is the image before attenuation, and I is the resulting underwater image. The first component, $L_0(x)e^{-\eta z(x)}$, is the direct transmission subject to attenuation; the additive component contributes to backscattering, which appears as haze in the resulting image.

Here we focus on modeling the direct transmission component to correct for the effects of attenuation. Future work will address backscattering for hazy scenes. Our approach requires estimation of range, or depth maps, z of the scene, as well as the attenuation coefficient η. In classical models, η is considered to be wavelength-dependent, such that different wavelengths are attenuated at different rates, leading to the characteristic color of underwater imagery. This would mean estimating a scalar η_c per color channel. Recently, Akkaynak et al. presented a novel model of attenuation that suggests that estimating η in this way induces error in color correction algorithms [62]. Instead, [62] proposes, and validates through experimentation, that η is dependent on many factors including sensor characteristics, water properties, and range to the scene. These factors are challenging to measure or estimate for all scenes. Instead, we propose a learning module to estimate and account for attenuation in order to correct color of underwater imagery.

Our color correction network (Fig. 3.3) is based on [66], which was initially developed for terrestrial image restoration. We make several modifications for our underwater image restoration pipeline. Our network features two branches with shared weights. Each branch takes a raw underwater image and its respective disparity map output from the disparity estimation module. Disparity maps are converted to depth maps and scaled to improve numerical stability. The paired depth map and image are concatenated and input into a fully convolutional network. The output of our network is then multiplied by the raw underwater image to output a color corrected image. Relating back to the model of underwater image restoration, our network estimates $e^{\eta z}$, the inverse of the transmission.

We use a two-stage loss for our network. The initial and refinement color correction losses are given by:

$$\mathcal{L}_{color\_init} = \psi_1 \mathcal{L}_{color\_warp} + \psi_2 \mathcal{L}_{color\_cyc} + \psi_3 \mathcal{L}_{gray} + \psi_4 \mathcal{L}_{IQ} \tag{3.6}$$

and

$$\mathcal{L}_{color\_ref} = w_{init} \mathcal{L}_{color\_init} + w_{ref} \mathcal{L}_{disp\_ref} \tag{3.7}$$

where ψ and w are hyperparameters for relative weighting of loss components determined experimentally.

Photometric Warping Loss ($\mathcal{L}_{color\_warp}$)

To ensure that both the left and right images appear with consistent colors after image restoration, we use a photometric warping error similar to that in the disparity estimation network, where the estimated disparity maps are used to warp one corrected image to the

other view. For example, the left corrected image C_L is warped to the right view $C_R^{'}$.

$$\mathcal{L}_{color\_warp} = \|C_L^{'} - C_L\|^2 + \|C_R^{'} - C_R\|^2 \tag{3.8}$$

Cyclic Reconstruction Loss ($\mathcal{L}_{cyc}$)

Inspired by recent advances in cyclic networks [67], we employ a cyclic reconstruction loss. This helps to ensure that the learned color correction parameters do not collapse to trivial results. Inverse color correction is performed for each corrected image, producing $\widehat{I_L}$ and $\widehat{I_R}$. That is, each corrected image is divided by the network output to estimate the original raw input images. Reconstruction loss is used to ensure that the inverted images align with the original raw inputs:

$$\mathcal{L}_{cyc} = \|I_L - \widehat{I_L}\|^2 + \|I_R - \widehat{I_R}\|^2 \tag{3.9}$$

Image Quality Losses ($\mathcal{L}_{IQ}$, $\mathcal{L}_{gray}$)

We also leverage knowledge from image processing to force our output to have high image quality using the gray world assumption, acutance, and contrast metrics. The gray world loss $\mathcal{L}_{gray}$ is based on prior knowledge of natural image statistics and assumes that the average color in natural images is gray. This constraint minimizes the distance between the average color of each color channel and gray. The contrast constraint q_c maximizes gain in contrast of the corrected image compared to the raw underwater image, which typically suffers from reduced contrast [68]. Lastly, the acutance constraint q_a indicates sharpness in an image and can be computed by the gradient strength of the image [68]. We use an image quality loss given by:

$$\mathcal{L}_{IQ} = 1.0 - (w_a q_a + w_c q_c), \tag{3.10}$$

where w_a and w_c are scalars.

Disparity Loss on Corrected Color ($\mathcal{L}_{disp\_ref}$)

Lastly, we leverage the interdependency between structure and color to provide further constraints on our output color imagery. Our corrected color images are input into the pre-trained coarse disparity network. The coarse disparity output is then refined to give a smooth disparity map. The weights of the residual refinement network are trained. We compute the loss $\mathcal{L}_{disp\_ref}$ as described in the above section. This loss on the corrected color image ensures that resulting image quality and features are sufficient for accurate disparity estimation.

Note that this approach is completely self-supervised, where the target output for each stage is input or generated by a previous stage. The network is designed to be modular, so that each learned component can be used on its own or substituted for another network or approach.

3.4 Experiments

(a) BlueROV (b) Artificial platform for ground truth

Figure 3.4: Stereo camera configured on the bottom of the BlueROV and artificial rock platform with attached color board for ground truth structure and color.

3.4.1 Data Collection

To validate our method, we deploy a BlueRobotics BlueROV2 remotely operated vehicle (ROV) equipped with a custom downward facing stereo camera system to gather underwater stereo imagery (Fig. 3.4a). A color board and artificial rock platform are submerged for ground truth (Fig. 3.4b). We collected two datasets near the Hawaii Institute of Marine Biology (HIMB). Each site contained different features and had different water column properties. The first site, HIMB #1, was in an open bay containing coral, with relatively murky water. The dataset contains 1371 images and 10 test images with ground truth. The second site, HIMB #2, features rocks and manmade objects such as cement blocks. This site was in a sheltered canal. The dataset contains 2676 images and 5 test images with ground truth. Note that all images containing the ground truth color board and artificial

rock platform were removed for training. See Appendices A and C for more information about the data and the ROV platform used for data collection, respectively.

3.4.2 Training Details

We pre-train the disparity module using an in-air dataset [69] for 100k iterations, which takes approximately 30 hours on a Titan V graphics processing unit (GPU). We then finetune the disparity network on both HIMB datasets for 10k iterations. Table 3.1 shows training parameters used for training the disparity estimation module of UWStereoNet.

Table 3.1: Training parameters for the disparity estimation module of UWStereoNet.

Parameter	**Value**
w_1	0.3
w_2	0.7
β_1	0.8
β_2	0.01
β_3	0.001
γ_1	0.8
γ_2	0.05
γ_3	0.005
Crop height	256
Crop width	512
Batch size	1
Max. num. disparity	192
Initial learning rate	$1e^{-5}$
Learning rate decay	0.1
Learning rate decay frequency	50000

For training the color correction network, we train only on an individual HIMB dataset for 4k iterations, and show testing on the corresponding test set. Training of the color correction module takes approximately 1 hour, which we believe is reasonable for adapting to different environments. Table 3.2 shows training parameters used for training the color correction module. See Appendix B for further implementation details.

3.5 Results & Discussion

3.5.1 Qualitative Results

Figure 3.5 shows qualitative results of our color correction network. We compare our method to more traditional image processing approaches, including histogram equalization

Table 3.2: Training parameters for the color correction module of UWStereoNet.

Parameter	Value
Batch size	1
ψ_1	0.001
ψ_2	1.0
ψ_3	100.0
ψ_4	1.0
w_a	$1e^{-8}$
w_c	$1e^{-8}$
w_{init}	0.5
w_{ref}	1.0
Learning rate	$1e^{-5}$

and gray world image correction [30], as these are popular methods for preprocessing of underwater data for high level vision tasks. We also compare to UGAN, a state-of-the-art deep learning approach for underwater image restoration, using the pretrained model [52]. Qualitatively, histogram equalization gives the sharpest resulting image with enhanced contrast. UGAN results in a haloing effect in areas that were not fully corrected. Compared to other methods, our method results in a consistent color across different viewpoints.

Figure 3.6 shows qualitative results of disparity estimation. The traditional Semi-Global Block Matching (SGBM) approach [70] results in the sparsest disparity map, especially around edges and occluded regions. The proposed disparity network provides much denser results with notable improvement between pretraining on in-air data and finetuning on underwater data.

3.5.2 Quantitative Evaluation of Disparity Estimation

To measure the accuracy of each method's disparity output quantitatively, we create ground truth by scanning the rock platform (Fig. 3.4b) with an ASUS Xtion Pro, and inputting the scan into ElasticFusion [50]. The resulting cloud was next filtered and transformed into a mesh, from which one million points were sampled to obtain a dense, evenly distributed set of reference points. To enable comparison, output disparity from each method is cropped to mask only the rock platform region. This section is then converted to a point cloud, which is filtered for invalid disparity ranges. The point cloud is hand-registered to the reference ground truth point cloud in Cloud Compare [71]. This coarse registration is improved through the Iterative Closest Point algorithm [72]. Finally, a modified Hausdorff distance between the two clouds is calculated. Table 3.3 shows that SGBM outperforms the proposed network when it is only pre-trained in air, but once finetuned on underwater

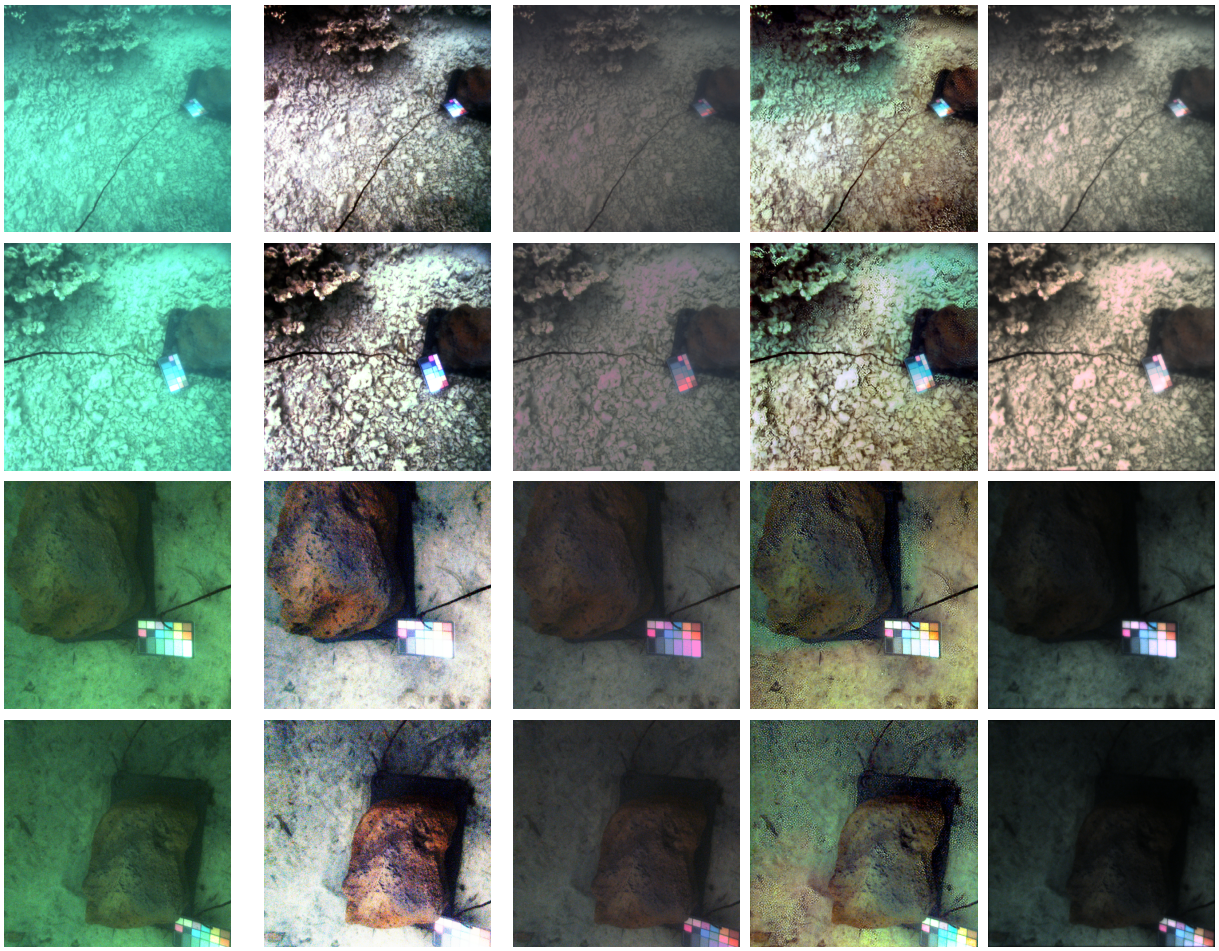


Figure 3.5: Color correction: We compare the results of each method on four test image sets. The first column displays a raw stereo image, followed by histogram equalization, which shows the sharpest image but becomes oversaturated. Gray World results in an over-amplified red channel on the color board. The fourth column contains the output of UGAN, which has unnatural coloring on the ocean floor. Finally, UWStereoNet's output is provided in the last column, with photorealistic coloring for the natural terrain and color board.

Raw Stereo Image SGBM Pretrained - in-air Finetuned - underwater

Figure 3.6: Disparity estimation: The leftmost image is a raw stereo image, followed by the results of Semi-Global Block Matching (SGBM) when implemented on the respective stereo pair. We then provide the results of UWStereoNet, first from pretraining on in-air images, then of finetuning with underwater imagery. This practice yields a denser map of the scene, particularly on the side of the rock, as well as better texturing of the coral.

data, our approach achieves the highest performance of any technique compared. Through experimentation, it was also noted that the disparity estimation network trained directly on underwater imagery is robust to varying water conditions. However, to achieve this it was is important to have representative features in the data. We noted that models trained only on the coral site performed poorly where models trained across both sites were more successful on the quantitative rock evaluations. We also show the number of valid disparity points estimated by each method. Our proposed method generates a denser point cloud with a higher number of valid points in the region of interest. In other words, with our proposed method, we can recover an accurate disparity estimate in challenging regions that lead to failure when using the traditional feature-based approach SGBM.

Table 3.3: The mean and standard deviation of modified Hausdorff distance across point clouds generated from test images for each trained deep neural network (DNN) and traditional Semi-Global Block Matching (SGBM) when compared against the ground truth point cloud for the artificial rock structure. Number of valid points in resulting point clouds is also shown, where a higher value indicates improved point cloud density.

Dataset	**HIMB #1 - Coral**		
Method	**Mean**	**STD**	**# Pts.**
SGBM (full DP)	0.0862	0.0250	8936
Pretrained	0.1444	0.0936	17611
Finetuned	**0.0709**	0.0341	14658

Dataset	**HIMB #2 - Rocky**		
Method	**Mean**	**STD**	**# Pts.**
SGBM (full DP)	0.0738	0.0129	15837
Pretrained	0.2136	0.1264	41491
Finetuned	**0.0632**	0.0346	44636

3.5.3 Quantitative Evaluation of Color Correction

Table 3.4 shows quantitative evaluation of color correction. For each test image, we took the average value for each color on the color board. We then computed root mean square error (RMSE) in RGB-space compared to the ground truth color board imaged in air. Here we show the mean and standard deviation of RMSE across the test set for each site. Our method outperforms the other traditional and state-of-the-art methods.

Table 3.4: The mean and standard deviation of root mean square error (RMSE) (m) from ground truth color board across color corrected test images.

Dataset	**HIMB #1 - Coral**		**HIMB #2 - Rocky**	
Method	**Mean RMSE**	**STD**	**Mean RMSE**	**STD**
Raw	0.2203	0.0439	0.2219	0.0554
Hist. Eq.	0.1301	0.0232	0.1132	0.0108
Gray World	0.1579	0.0365	0.1204	0.0310
UGAN [52]	0.1461	0.0245	0.1555	0.0414
Our Method	**0.1065**	0.0187	**0.1037**	0.0159

3.5.4 Ablation Experiments

Table 3.5: Ablation experiments show results of training when individual components of the loss function are dropped out.

Dataset	**HIMB #1 - Coral**		**HIMB #2 - Rocky**	
Loss Components	**Mean RMSE**	**STD**	**Mean RMSE**	**STD**
No IQ	0.4741	0.0115	0.1055	0.0205
No photometric	0.1136	0.0216	0.1029	0.0157
No disparity loss	0.1647	0.0386	0.1045	0.0216
No gray world loss	–	–	–	–
No cyclic loss	0.1354	0.01843	0.0973	0.0254
Full loss	0.1065	0.0187	0.1037	0.0159

Table 3.5 shows ablation experiments for the developed loss function for the color correction module in UWStereoNet. For each experiment, an individual component of the loss was dropped out to determine the change in performance without each loss component. Note that without the gray world loss, the network failed to converge. These results indicate that the gray world and image quality losses are the predominant losses driving the color correction network. However, the other loss components are also important constraints that must hold to reach a realistic solution for corrected color of underwater images. The results of the ablation experiments also indicate that the ideal loss function may depend on characteristics

of the dataset. For experiments shown here, the hyperparameters were set experimentally to give sufficient results across both datasets. Further tuning of hyperparameters for a specific site or dataset may lead to improved results.

3.6 Conclusion

Our proposed method is a novel, modular learning pipeline for dense depth estimation and color correction of raw underwater stereo imagery. By leveraging knowledge from image processing, computer vision, and underwater light propagation, we are able to perform these tasks without supervision. Our experiments validate our method quantitatively and qualitatively to show that we outperform existing methods on both tasks. Future work will focus on integrating this learned vision system onto an underwater robotic platform in the field.

CHAPTER 4

Towards Real-time Underwater 3D Reconstruction

4.1 Introduction

Real-time perception is critical for achieving full autonomy of robotic systems. In particular, dense real-time three-dimensional (3D) reconstruction would be an enabling technology for many tasks carried out by autonomous underwater vehicles (AUVs) and remotely operated vehicles (ROVs), including, but not limited to, navigation, obstacle avoidance, mapping, and grasping and manipulation. There is already widespread use of robots and automated systems for subsea intervention missions such as deep-sea scientific sampling, resource extraction, and maintenance of offshore equipment. However, currently many of these tasks require a human operator in-the-loop. Real-time dense 3D reconstruction would transform current practice in subsea autonomous intervention for these precision applications.

As previously mentioned, these problems have been intensely studied for terrestrial applications, and real-time active color and depth (RGB-D) sensors have led to significant advances in perceptual capabilities of robotic systems on land. However, due to attenuation of electromagnetic signals underwater, success of active RGB-D sensing is limited in marine applications. Alternatively, passive optical sensing can be achieved even at depth using artificial light at close range.

In this chapter, we explore the potential for passive optical sensors to achieve real-time dense 3D reconstruction in underwater environments – a critical task for achieving autonomy across many applications. We present novel work that proposes advancements in two promising directions. First, we propose a novel end-to-end system that leverages deep learning to enable online 3D reconstruction of a scene from underwater stereo cameras. Second, we present a novel end-to-end system that uses a computational imaging approach leveraging plenoptic cameras to generate real-time 3D reconstruction of a scene. For each method, we demonstrate novel integration of the physics-based model of underwater light propagation to account for attenuation effects of the water column in online 3D model building.

We present qualitative and quantitative results that compare these models to terrestrially scanned ground truth to validate the accuracy of the proposed approaches.

This chapter is organized as follows: §4.2 will present background and prior work; §4.3 will give an overview of the proposed water column compensated 3D reconstruction approach; §4.4 will present detailed methodology for incorporating stereo cameras into this 3D reconstruction framework with experimental results for validation; §4.5 will present detailed methodology for incorporating plenoptic cameras into this 3D reconstruction framework with experimental results for validation; and finally, §4.6 will present conclusions and future work.

4.2 Background

4.2.1 Real-time Dense 3D Reconstruction

Real-time dense 3D reconstruction is a critical perception task for fully autonomous robotic systems. Recently developed RGB-D simultaneous localization and mapping (SLAM) systems have demonstrated the ability to perform this task with a high degree of success for land-based applications [73] [74]. Fusion-based methods perform online alignment of overlapping color images and depth maps to maintain and update a single 3D model, while tracking an estimated pose to enable global loop closures [75] [76]. Each of these methods exploits RGB-D sensors capable of returning high-resolution color images with corresponding depth or range measurements to sense the surrounding environment. We propose to extend a fusion-based framework, ElasticFusion [50], to work towards real-time underwater 3D reconstruction.

4.2.2 Underwater 3D Reconstruction with Stereo Cameras

Unfortunately, the aqueous medium presents unique challenges to transferring these terrestrial approaches to underwater environments. Complex effects on light propagation limit the effectiveness of active RGB-D sensors subsea and lead to a violation of the brightness constancy constraint (BCC) [25], each of which are commonly used to achieve real-time mapping in terrestrial applications. There have been recent advances in methods that use optical sensors for underwater applications. Several offline dense 3D reconstruction techniques have been developed and tested underwater with stereo camera data [77] [21]. Jordt-Sedlazeck et al. developed an underwater light propagation model to account for the effects of refraction through a flat port camera housing in both camera calibration and structure-from-motion [78] [79]. Additionally, Bryson et al. developed methods to perform color correction of im-

ages captured in underwater environments for color consistent reconstructions [29]. While these methods provide a basis for overcoming several fundamental limitations of underwater vision, they are all offline techniques. The goal of this chapter is to achieve real-time underwater 3D reconstruction.

4.2.3 Plenoptic Cameras

Plenoptic, or light field, cameras have recently gained interest as increased computing power and improved image resolution have enabled the development and release of consumer-grade instruments from Raytrix [80] and Lytro [81]. Instead of a traditional single lens, light field cameras have an array of micro lenses behind the main lens that captures the position and direction of each light ray to the imaged scene, making it possible to refocus the scene, or to obtain different viewpoints from a single image after the image is captured. From this information, it is possible to extract depth information as well as a high-resolution image from a single passive optical sensor [82]. There have been several methods recently developed to achieve real-time tracking and visual odometry with plenoptic cameras for use in real-time robotics applications [83] [84] [85]. Prior work on 3D reconstruction using light field cameras has focused on static single scene reconstruction [86]. Tao et al. [87] exploited both the defocus and correspondence cues to create high quality depth maps of single frame images, which could then be interpolated to form a 3D surface reconstruction. Our focus for this work is on performing the reconstruction of an online video feed rather than a single static scene in order to obtain a complete 3D model of a large object spanning multiple frames. Furthermore, our proposed methods are developed specifically for the underwater domain.

4.3 Technical Approach Overview

4.3.1 Fusion-based Reconstruction Framework

Our proposed architectures are built around the fusion-based reconstruction framework developed by Whelan, et al. [50], with modifications for the underwater domain. In fusion-based frameworks, the global 3D model follows the surfel representation proposed by Keller, et al. [74], where a surfel $\mathcal{M}^{\mathcal{S}}$ contains position, color, radius, normal, weight, timestamp of initialization, and timestamp of last update. The set of all surfels $\mathcal{M}$ is segmented into active and inactive subsets. The active subset consists of points that have been seen within a set time threshold δ_t, where the inactive subset consists of points that have not been seen for greater than δ_t. This bounds the computational complexity and memory footprint of the matching process and enables real-time performance.

In real-time, each incoming RGB-D frame is added to the active map and compared to the inactive model that, based on pose estimates, lies in the same spatial region. Tracking of the estimated pose is done by solving for the motion parameters (position and orientation) that minimize the following cost function:

$$E_{track} = E_{icp} + w_{rgb} E_{rgb} \tag{4.1}$$

where w_{rgb} is a tuning parameter, E_{icp} is the point-to-plane error function for geometric pose estimation, and E_{rgb} is the photometric error function based on intensity change from pixel to pixel. If the active model can be matched to an inactive portion, a local loop closure is made so the inactive map segment is made active again, aligned with the matching active portion through Iterated Closest Point (ICP) [88] and fused to the global 3D model. The red, green, and blue color (RGB) values are fused using a moving average of colors.

4.3.2 Application to Underwater Environments

In the 3D reconstruction framework described above, the term E_{rgb} represents the photometric consistency of an observation to the model. It is an energy term that expresses the BCC assumption underlying many terrestrial vision algorithms. As discussed previously, underwater optical imaging is a challenging problem because of light attenuation, light scattering, moving light sources, and other violations of that assumption. Thus, we propose to directly account for water column effects in each of our proposed frameworks. More specifically, we correct the RGB values of input images based on input depth estimation, and enforce a photometric constraint that is more consistent with reality simultaneously to structure computation. As such, this proposed method seeks to improve both the geometric accuracy as well as photometric quality of the resulting 3D model.

4.3.3 Validation Procedure

To demonstrate that the proposed methods have validity we gathered a 3D model underwater and compared the accuracy and fidelity of that model to ground truth gathered in air. To do this we generated a ground truth dataset by scanning the object with an RGB-D structured light sensor. Here we present a metric to demonstrate the accuracy of the model structurally with respect to the terrestrial ground truth.

To produce a quantitative comparison we use the well established point-to-surface distance metric and calculate it over the experimentally gathered mesh compared with the ground truth mesh. The point-to-surface distance $d(p, S)$ for a point p and a surface S is

given by [89]:

$$d(p, S) = \min_{p \in S} d(p, p') \tag{4.2}$$

As a point-to-point distance measure $d(p, p')$, we use Euclidean distance in $\mathbb{R}^3$. Then to determine the distance between two surfaces S_1, S_2 we define a one-sided distance $E(S_1, S_2)$ as follows:

$$E(S_1, S_2) = \max_{p \in S_1} d(p, S_2) \tag{4.3}$$

Since this distance is not necessarily symmetric, we then take the Hausdorff distance, so the final result is the maximum of $E(S_1, S_2)$ and $E(S_2, S_1)$.

4.4 Towards Real-time Underwater 3D Reconstruction with Stereo Cameras

4.4.1 Methods

In this section, we propose a novel pipeline for achieving online underwater 3D reconstruction using stereo cameras. The proposed pipeline relies on our prior work developed in Chapter 3, which leverages unsupervised learning to achieve dense depth estimation and color correction of raw underwater stereo images. Figure 4.1 shows a flowchart of the proposed pipeline. Raw underwater stereo images are rectified and input to UWStereoNet for inference. The output is dense corrected RGB-D images. Currently, inference is achievable at a rate of approximately 2 Hz. The output RGB-D data is then input into a framework developed for terrestrial applications that is capable of achieving real-time 3D reconstruction. For this work, we use ElasticFusion for the 3D reconstruction step. This process outputs a metrically accurate 3D model of an underwater scene.

4.4.2 Experiments

4.4.2.1 Experimental Setup

To validate the proposed pipeline for achieving online 3D reconstruction with stereo cameras, we collected a dataset that contains ground truth of 3D structure. To collect the ground truth, we scanned an artificial rock platform in air with an Asus Xtion Pro active RGB-D sensor, and performed 3D reconstruction with ElasticFusion [50]. We then submerged the artificial rock platform near Sydney Institute of Marine Science (SIMS) and

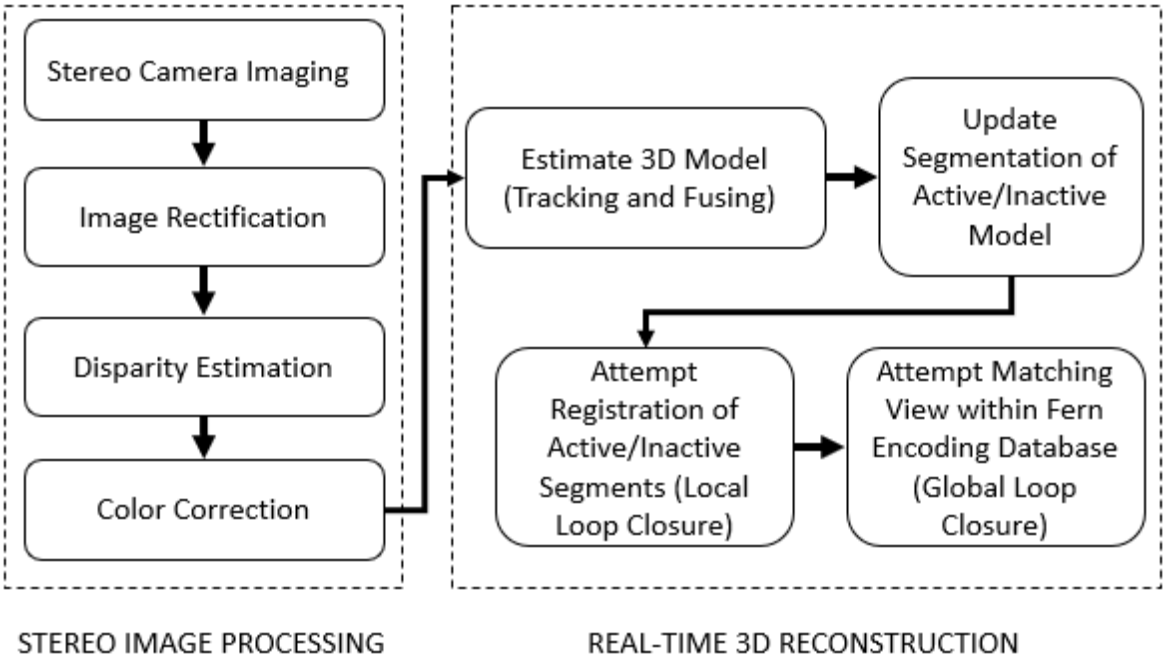


Figure 4.1: Flowchart of overall pipeline for the methods presented in this section. The key points are organized into the following sections: stereo image processing with compensation for underwater lighting effects, and real-time 3D reconstruction.

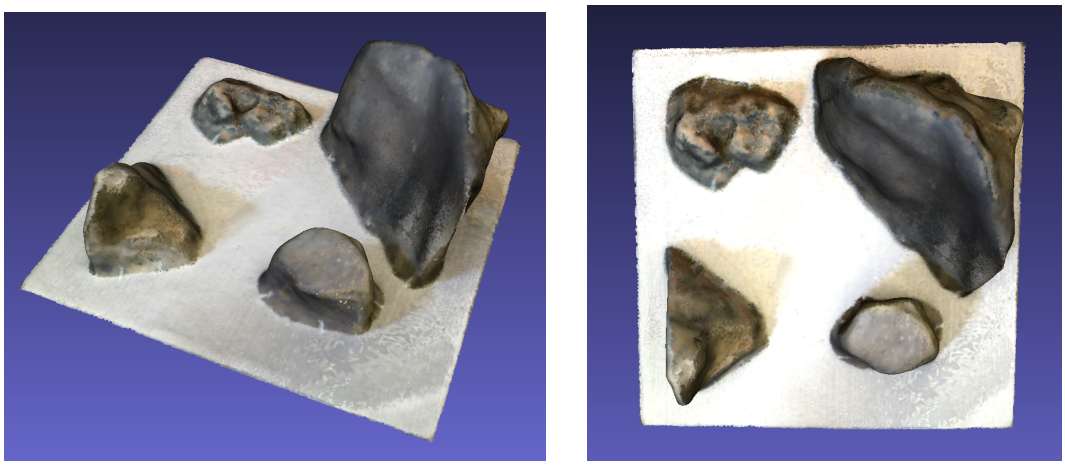
conducted a smooth imaging survey with the Deep Robot Optical Perception Laboratory (DROP) Lab BlueROV2 equipped with a custom stereo camera payload containing two color machine vision cameras. Note that the water quality was relatively clear in this local area. Appendices A and C detail more information about the data and the ROV platform used for data collection, respectively.

For training UWStereoNet, we first attempted to use data collected near the SIMS test site for training. However, training of the disparity network did not converge using training data from this site. We believe this is due to lack of structure and texture in the surrounding area, as most of the site consisted of sand and moving seaweed. For the purposes of this experiment, we pre-trained the disparity network using the Cityscapes dataset [69] and finetuned using the Hawaii Institute of Marine Biology (HIMB) datasets used throughout Chapter 3. To train the color correction module, we used data collected at the SIMS site including the artificial rock platform. We tested the trained model on a held out survey of the rock platform. This output dense depth estimation and corrected color images, which we used directly as input to ElasticFusion [50].

For all experiments, we used the open-source software, MeshLab [90] [91], to compute the Hausdorff distance of the output 3D point clouds compared to the ground truth mesh.

4.4.3 Results

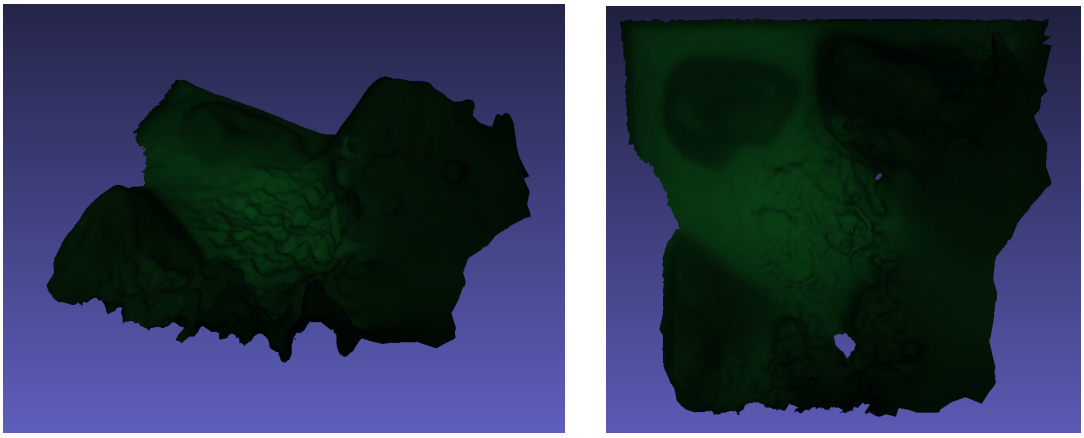
Figures 4.2a and 4.2b show the side and top views of the ground truth 3D model scanned in air with an Asus Xtion Pro active RGB-D sensor and reconstructed with ElasticFusion [50].



(a) Side View (b) Top View

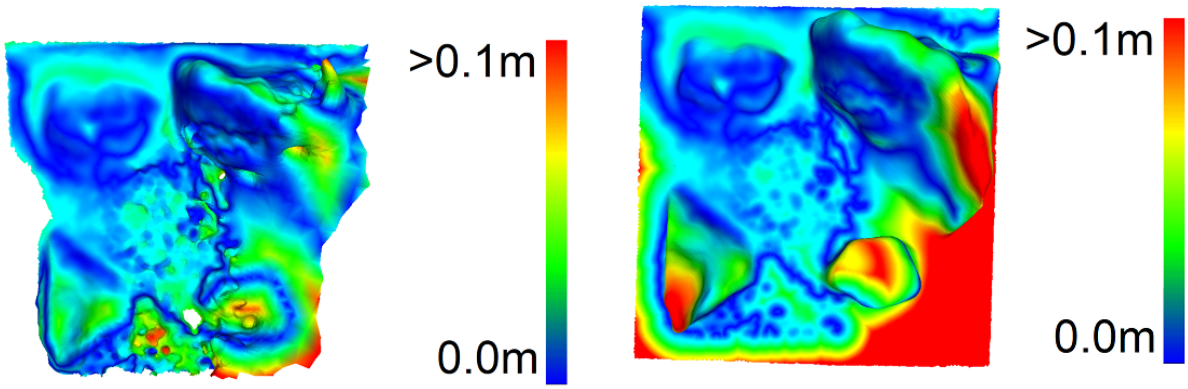
Figure 4.2: Ground truth mesh gathered with ASUS Xtion Pro RGB-D sensor and reconstructed with ElasticFusion.

We first compare our proposed pipeline to the performance of a widely-used commercially available software, Agisoft Photoscan [92], which performs offline dense 3D reconstruction from images. We show results of Photoscan 3D reconstruction using data collected with stereo cameras in an underwater survey of the artificial rock platform. Figures 4.3a and 4.3b show qualitative views of the 3D model computed using data gathered from the robotic platform underwater, and reconstructed using Photoscan. Figures 4.4a and 4.4b show the distance between the 3D model computed using Photoscan compared to the ground truth 3D model. Note that this is a one-sided metric. With the ground truth mesh as the reference (Fig. 4.4a), the visualization shows the distance of each vertex in the resulting Photoscan mesh to the closest point in the ground truth mesh. This direction does not penalize holes or missing points in the resulting Photoscan mesh. However, it shows the error of estimated points using Photoscan. From the other direction, with the resulting Photoscan mesh as reference (Fig. 4.4b), the visualization shows the distance of each vertex in the ground truth mesh to the closest vertex in the resulting Photoscan mesh. This leads to large distances, or high error, where Photoscan fails to estimate valid points. For visualization purposes, the maximum distance is clipped to $0.1m$.



(a) Side View (b) Top View

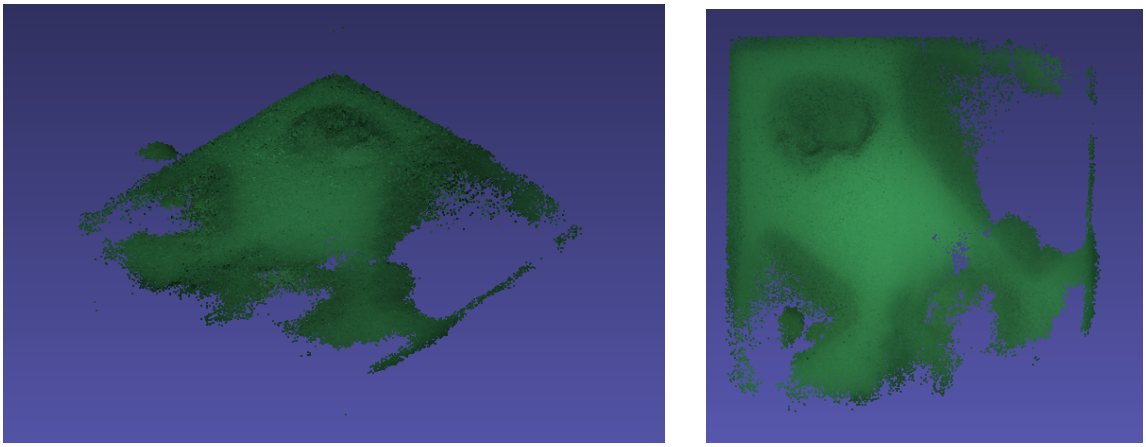
Figure 4.3: Output 3D model computed using data collected with machine vision stereo cameras underwater and reconstructed with Agisoft Photoscan 3D reconstruction software.



(a) Reference: Ground truth (b) Reference: Photoscan

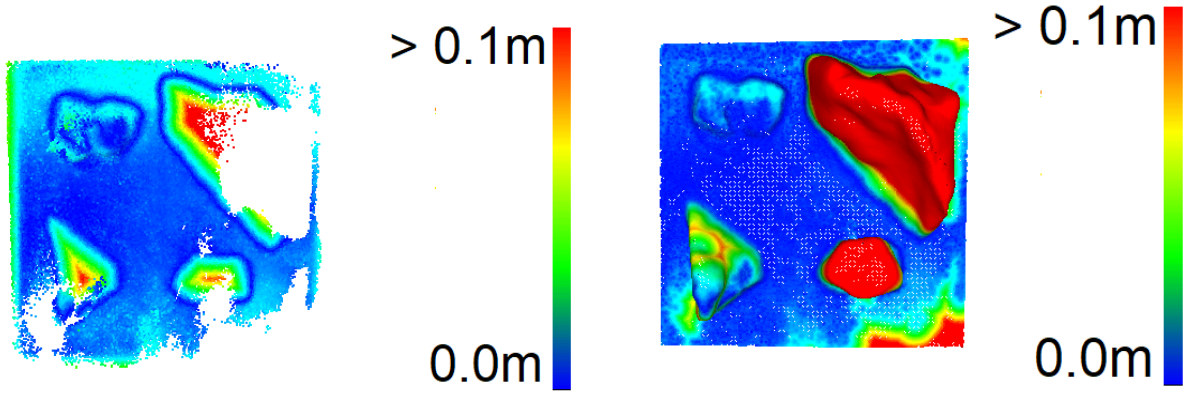
Figure 4.4: Visualization of the distance between the 3D model computed using Agisoft Photoscan compared to ground truth. Comparison is shown using ground truth as the reference cloud (left), and the Photoscan result as the reference cloud (right). For visualization purposes, the maximum distance is clipped to $0.1m$.

We also compare our results to a pipeline that uses traditional feature-based disparity estimation, Semi-Global Block Matching (SGBM), and raw underwater images input directly into ElasticFusion. Figures 4.5a and 4.5b show qualitative views of the resulting 3D model. Figures 4.6a and 4.6b show the distance between the 3D model computed using SGBM and ElasticFusion, compared to the ground truth 3D model gathered in air. Again, we show results using both the ground truth as reference (Fig. 4.6a) and the estimated result as reference (Fig. 4.6b). For visualization purposes, the maximum distance is clipped to $0.1m$.



(a) Side View (b) Top View

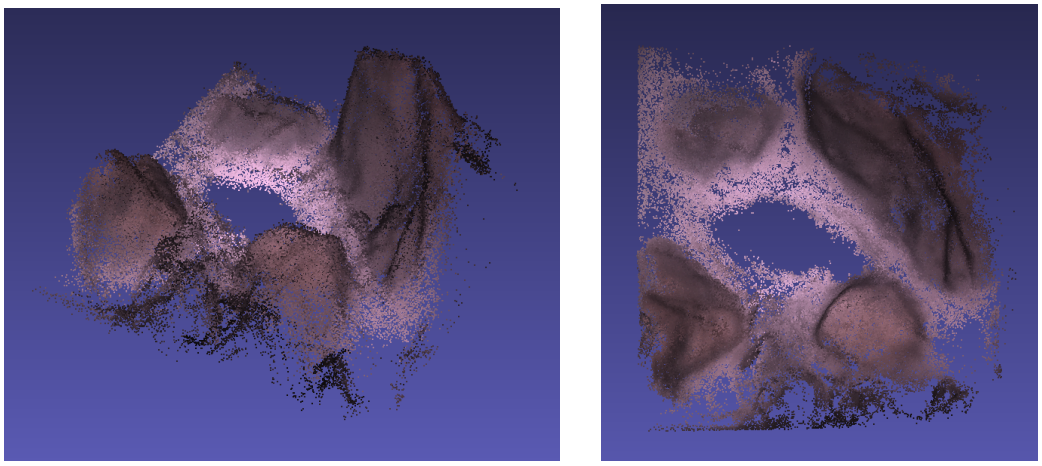
Figure 4.5: Output 3D model computed using data collected with machine vision stereo cameras underwater using Semi-Global Block Matching (SGBM) for disparity estimation and ElasticFusion for 3D reconstruction.



(a) Reference: Ground truth (b) Reference: SGBM

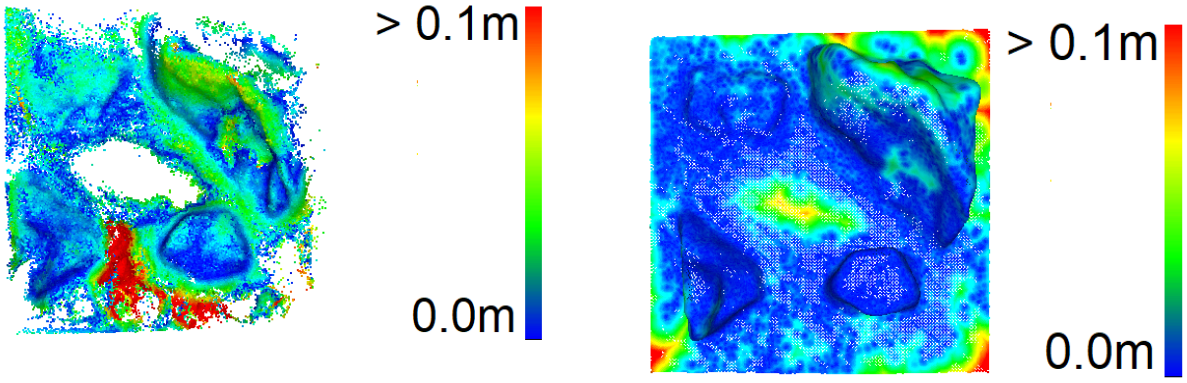
Figure 4.6: Visualization of the distance between the 3D model computed using Semi-Global Block Matching (SGBM) for disparity estimation and ElasticFusion for 3D reconstruction, compared to ground truth. Comparison is shown using ground truth as the reference cloud (left), and the output result as the reference cloud (right). For visualization purposes, the maximum distance is clipped to $0.1m$.

Lastly, Figures 4.7a and 4.7b show the side and top views of the 3D model computed with data gathered from the robotic platform underwater and reconstructed with our proposed pipeline for online underwater 3D reconstruction. Figures 4.8a and 4.8b show the distance comparing the 3D model output using our method to the ground truth mesh. Once again, for visualization purposes, the maximum distance is clipped to $0.1m$.



(a) Side View (b) Top View

Figure 4.7: Output 3D model computed with data gathered from the robotic platform underwater and reconstructed with our proposed pipeline.



(a) Reference: Ground truth (b) Reference: Our Method

Figure 4.8: Visualization of the distance between the 3D model computed using our proposed method compared to ground truth. Comparison is shown using ground truth as the reference cloud (left), and our result as the reference cloud (right). For visualization purposes, the maximum distance is clipped to $0.1m$.

Table 4.1 shows quantitative error using the Hausdorff distance for each 3D reconstruction method shown above compared to the ground truth 3D model. Our proposed method results in the lowest Hausdorff distance compared to ground truth gathered in air.

4.4.3.1 Timing Studies

Table 4.2 shows approximate time required to perform 3D reconstruction using Photoscan, SGBM, and our proposed method. Photoscan is an offline method. Once all images

Table 4.1: Hausdorff distance between the dense point cloud generated by each method compared to ground truth.

Method	**Hausdorff Distance (m)**
Photoscan	0.3046
SGBM	0.3982
Our Method	**0.1935**

are collected, it takes approximately 562 seconds for dense 3D reconstruction using a survey of 244 image pairs. Note that additional time is required for mesh texturing. Alternatively, ElasticFusion is a real-time 3D reconstruction framework that is capable of operating at a rate of over 30 Hz. Note that our current implementation is not end-to-end. For proof-of-concept, we convert the underwater survey, with images collected at a rate of 3.5 Hz, to a log file that can be input to ElasticFusion to simulate online 3D reconstruction. To evaluate timing of SGBM and our method, we only compare the bottleneck operations for each method, noting that this neglects overhead that may be present in an end-to-end implementation. For SGBM, the main bottleneck operation is the disparity estimation step, which takes approximately 0.43 seconds per stereo pair. For the full test survey containing 244 image pairs, this would take approximately 105 seconds, neglecting any overhead required for the end-to-end implementation. Similarly, for our proposed method, the bottleneck operation is the inference step for UWStereoNet, which performs dense depth estimation and color correction. This inference takes approximately 0.53 seconds per stereo pair, which would take approximately 129 seconds for the full survey. Note that inference for UWStereoNet has not been optimized for computation time, so there is potential to achieve faster computation time, which is left for future work.

Table 4.2: Approximate time required to generate dense point cloud across each method given input number of stereo pairs.

Method	**Approximate Time (s / min)**
Photoscan	562/9.37
SGBM	105/1.75
Our Method	129/2.15

4.4.4 Discussion

This section presented a pipeline for leveraging a deep neural network (DNN), UWStereoNet, for online underwater 3D reconstruction with stereo cameras. UWStereoNet enables computation of dense depth maps and corrected color imagery from raw underwater stereo

imagery. Experimental results demonstrated proof-of-concept that this output could be used within a real-time 3D reconstruction framework developed for terrestrial applications. We compared results to two different methods. First we compared to commercially-available 3D reconstruction software, Photoscan. Photoscan is an offline method that optimizes the resulting 3D model over the entire dataset. This method takes over 9 minutes to perform 3D reconstruction for these experiments. However, it is important to note that as the number of images increases for large surveys, the time to perform 3D reconstruction will increase drastically, which can hinder efforts to perform 3D reconstruction over large sites. Next we compared to a pipeline that leverages traditional disparity estimation methods SGBM for depth estimation as input to ElasticFusion for 3D reconstruction. Although this method has potential to perform online 3D reconstruction, it showed poor performance in reconstructing the rocks on the platform. We believe this is due to inconsistencies in valid disparity estimates from frame-to-frame, as the rocks feature many edges with dropoffs that are challenging for SGBM to reconstruct consistently under changing viewpoints. Comparatively, our method performs accurate dense depth estimation directly. Our proposed method resulted in the lowest Hausdorff distance for the test survey. This demonstrates the potential of DNNs to enable real-time underwater 3D reconstruction. Future work will focus on optimizing the inference time of UWStereoNet to achieve real-time underwater 3D reconstruction with stereo cameras. Additionally, we will focus on an end-to-end implementation to demonstrate online 3D reconstruction with a robot in the field.

4.5 Towards Real-time Underwater 3D Reconstruction with Plenoptic Cameras

4.5.1 Methods

While traditional stereo cameras have met with great success underwater [77], there are several challenges that are inherent to operating stereo cameras in underwater environments. The operational range limitation due to water column effects is complicated by the fact that typical submersible platforms lack the instant control authority to follow an undulating benthos precisely, leading to rapidly varying ranges between camera and target. This creates a challenge for traditional optics as the depth of field (DOF) is insufficient to maintain focus across the typical range of platform altitudes. Additionally, the physical stereo camera setup is often quite cumbersome as the underwater housing for the cameras becomes heavier and more expensive as the inner diameter increases to accommodate two cameras with even a modest baseline of separation.

Plenoptic, or light field, cameras have been proposed as an alternative to traditional stereo cameras for underwater applications [93]. These cameras increase the DOF enabling focus across a much greater range of alitudes from the target. Additionally, they allow for depth recovery from a single imaging sensor, and single optics, resulting in a compact form factor. This reduces the challenge, complication, and weight for design of a larger depth-tolerant pressure housing. Thus, non-traditional computational imaging approaches have great potential to lead to advanced perceptual capabilities of robotic systems. Still, computational imaging sensors such as plenoptic cameras have not yet seen widespread application to address challenges across robotics.

This section proposes a novel method to integrate plenoptic cameras into a framework for real-time underwater 3D reconstruction. Figure 4.9 shows a flowchart of our proposed processing pipeline, broken into three main parts: processing of data from the light field camera, incorporation of an underwater light propagation model for this data, and the real-time 3D reconstruction procedure.

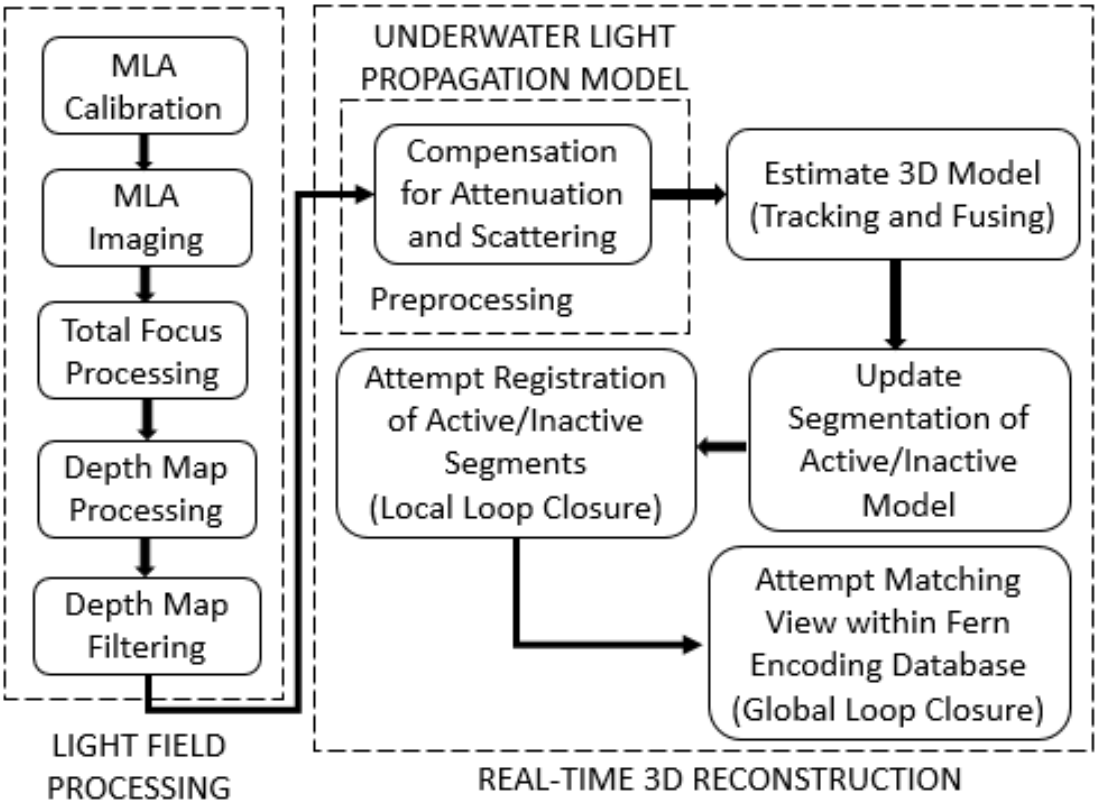


Figure 4.9: Flowchart of overall pipeline for the methods presented in this section. The key points are organized into three main sections: light field processing, compensation for underwater lighting effects, and real-time 3D reconstruction.

4.5.1.1 Light Field Processing

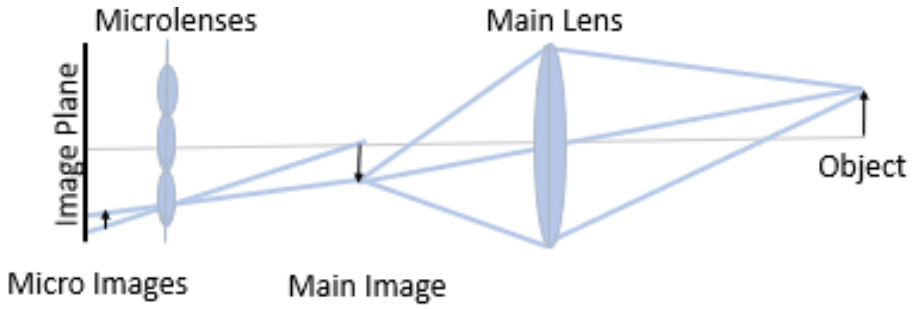


Figure 4.10: (Modified from [80]) Camera geometry of a Raytrix light field camera showing the placement of the micro lens array between the main lens and the image plane. This allows the ray intensity and direction to be stored to provide a color image and depth map from a single camera.

Figure 4.10 shows the basic geometry of a light field camera, where the main difference from traditional cameras is the microlens array (MLA) placed between the main lens and the image plane. The following outlines the basic pipeline for light field processing shown in Fig. 4.9. For further details of development and implementation of these steps, the reader is referred to [80], as these details are out of the scope of this work.

4.5.1.2 Calibration

First an MLA calibration is required to compute intrinsic camera parameters, including parameters of the micro lens structure for the light field camera. The raw uncalibrated image returned by the light field camera displays the raw intensity values read by the imaging sensor during image capture. This may be either monochrome or, for our use, color. As shown in Fig. 4.11a, the individual micro lenses can still be distinguished in the raw image.

To remove the segmentation of micro lenses from the raw image, a gray image is taken during the MLA calibration with a white filter for continuous illumination to highlight only the edges and vignetting of each micro lens. This provides a template for eliminating the micro lens vignetting from the raw image to produce a processed image. Finally, a metric calibration must also be performed to provide conversion from pixel space units to metric units.

The result of the calibration provides intrinsic parameters, which include micro lens diameter and offset, x- and y-direction field-of-view, depth-of-field, working distance (effective focus distance within which an object could move and remain in focus), distortion parameters, as well as lateral and depth resolution.

4.5.1.3 MLA imaging

Each of these micro lenses captures a small fraction of the total view of the imaging plane. These micro images are synthesized to form the processed image at a plane of focus (Fig. 4.11b). To do this, the depth (or virtual depth – the distance from the object to the *micro* lens) must be assumed for each pixel. For each point (x, y) in object space, a set of micro lenses is compiled that captures that point in their image plane. Then the point's projection through each of these micro lenses onto the full image plane is determined. The final color value for the pixel in the full image plane is taken as an average of pixel intensities from the corresponding micro images.

4.5.1.4 Total focus processing

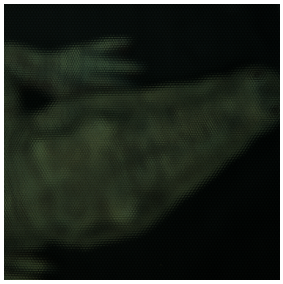
With a plenoptic camera, DOF increases and working distance – or the range of depths a scene can remain in focus – is larger than a traditional camera. Additionally, plenoptic cameras can be designed as multi-focus cameras, where there are different types of micro lenses with each type set to a different focus throughout the MLA. Each group of similar micro lenses is considered as a subarray. An image with"total focus," or one in which a large DOF is maintained can be synthesized using multiple subarrays of different focal lengths to compute an image in focus over a large working distance. Figure 4.11c shows a total focus image. Note the clarity versus the single plane of focus in Fig. 4.11b.

4.5.1.5 Depth map processing

Plenoptic cameras are designed such that each micro image shares correspondences with its neighbor, so that when the MLA is calibrated to determine geometric parameters between micro lenses, these correspondence cues can be used for determining depth via triangulation. Note that the best triangulation performance is achieved in regions of high contrast. Figure 4.11d shows a sample depth map from our light field camera. Note the noise that appears in the low contrast region. The following section describes an approach to ameliorate this issue.

4.5.1.6 Depth map filtering

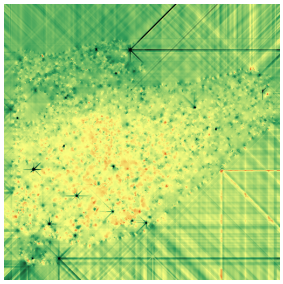
To decrease noise in the final depth map, we filtered the depth with a combination of mean, median, and bilateral filtering. Additionally, we implemented a maximum depth filter which eliminates noise or erroneous depth measurements beyond the plane of interest by eliminating these points from consideration in the reconstruction process. Robustness



(a) Raw MLA image initially returned by the light field camera prior to calibration. The edges of each micro lens are visible due to vignetting.

(b) Refocused image after MLA calibration and refocusing for single focal length.

(c) Image with total focus showing the large DOF achievable. Note that the scene is in focus through the full depth of the object.



(d) Depth map after MLA and metric calibration with colors ranging from yellow to green as depth from the camera increases.

Figure 4.11: Set of processing steps used to go from raw microlens array (MLA) image to focused images and depth maps. This data serves as the input for our 3D reconstruction pipeline. The use of the plenoptic camera affords us color and depth (RGB-D) images at high framerate in a domain where other techniques like pattern projection and time-of-flight are difficult or impossible to employ. Note the high noise in the depth image (d). Such images necessitate the use of a three-dimensional (3D) reconstruction pipeline where noise can be suppressed through multiple observations.

to remaining noise is afforded by the subsequent processing pipeline to generate a full 3D model from many noisy observations

4.5.1.7 Compensation for Water Column Effects

After the light field processing steps, both color and dense range information are available in the graphics processing unit (GPU) and so the entire water column compensation can be implemented as a pre-processing fragment shader in the frame integration step of the 3D reconstruction pipeline (see Fig. 4.9). Here we employ a modified Jaffe-McGlamery model [28] [40], which when implemented on the GPU can achieve real-time performance in our online 3D reconstruction system:

$$L_r(\lambda, z) = L_0 e^{-\eta(\lambda) z} \tag{4.4}$$

where L_0 is the initial radiance and $L_r(\lambda, z)$ is received radiance after traveling distance z. Values for the attenuation coefficient $\eta(\lambda)$ can be estimated using previous approaches [29] and the proposed technique assumes that $\eta(\lambda)$ has been estimated a priori. In implementa-

tion we discretize $\eta(\lambda)$ to the three major wavelengths captured by our camera – red, green, and blue light – and use the known pure water attenuation coefficients shown in Table 4.3.

Table 4.3: Pure water attenuation coefficients accounting for both absorption and scattering.

λ in $[nm]$	Attenuation η in $[m^{-1}]$
440 (blue)	0.015
510 (green)	0.036
650 (red)	0.350

Note that the water column effects within a single frame in the MLA processing steps are currently being ignored for implementation simplicity and in practice the intra-frame variations in depth are much less significant than the inter-frame camera movement in effecting lighting path lengths. However, such a correction could be easily added to the single frame depth estimation step and is planned for future work.

4.5.2 Experiments

4.5.2.1 Experimental Setup

We tested the above approach in a controlled lab setting (see Fig. 4.12) with a Raytrix R5 light field camera (see Fig. 4.13a and Table 4.4 for technical specifications [94]) in a custom flat port underwater housing (Fig. 4.13b). The underwater housing is rated for a depth of up to 500 m. For our optical setup we mounted a 25 mm lens, focused at infinity, on the Raytrix R5. This focus was chosen so the total target would be in focus from a maximum survey altitude of 1 m above the object. Note that, in general, the ideal survey altitude for plenoptic cameras is limited to closer ranges compared to stereo camera surveys. The achievable lateral resolution of this setup is 0.88 mm, and the achievable depth resolution is 7.38 mm with an approximate x- and y-plane field-of-view of 1.33 m, and depth-of-field of 2.64 m.

The test procedure was as follows: the object was placed at the bottom of a large water tank, the camera was calibrated in its housing in water, and then the camera was moved over the object at a distance of approximately 1 meter from the object in multiple controlled passes while the data was processed on two networked and GPU equipped Intel i7 3.0Ghz Desktop PCs with 16GB of RAM. With our system consisting of an NVIDIA GTX 980, we achieved processing rates around 25-30Hz for images at an output resolution of 1024×1024. This frame rate is sufficient for typical 3D reconstruction tasks we envision on an AUV or ROV. Note that the calibration provides intrinsic and extrinsic camera parameters including

Table 4.4: Technical specifications for the Raytrix R5 light field camera [94].

Specification	**Value**
Resolution	2048x2048
Pixel Size	5.5 micrometer square pixels
Max Frames Per Second	56
Dimensions	52x52x37mm
Focal Length	25mm

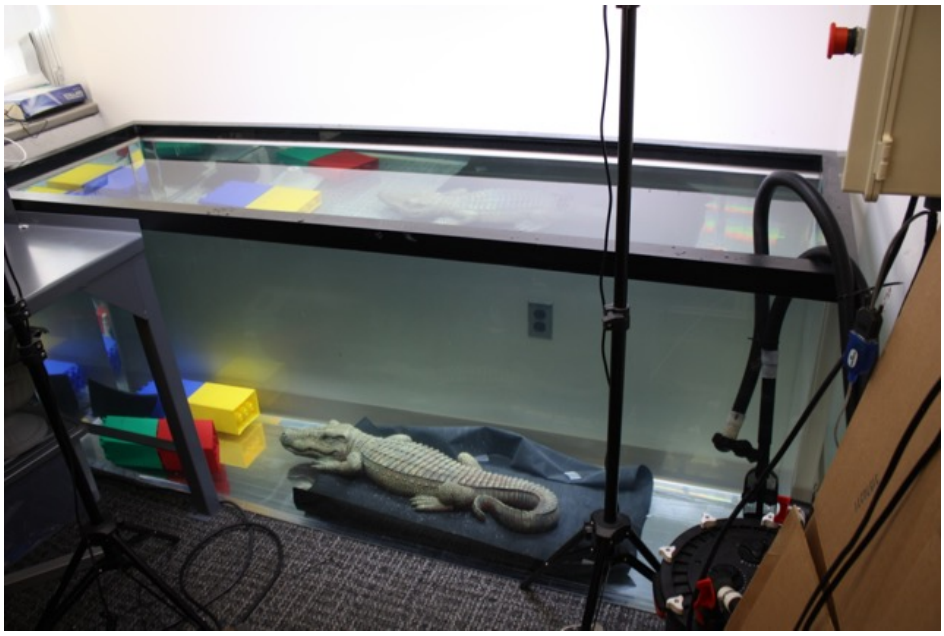


Figure 4.12: Laboratory water tank setup and target object for reconstruction. Lighting and water clarity can be tightly controlled enabling us to test the limits of the proposed approach.

distortion effects, which are accounted for during light field processing to obtain online depth maps and total focus images for input to our proposed 3D reconstruction methods. As previously described, we used the open-source implementation of ElasticFusion as a base for our reconstruction methods [50], and modified this existing implementation to integrate our underwater light propagation model and account for water column effects. Since we have RGB and dense depth information directly output from the plenoptic camera, this model consists of a single constant-time operation, and thus this integration does not significantly impact the run-time of the proposed pipeline.

(a) Raytrix R5 light field camera [94]

(b) Flat port housing

Figure 4.13: Underwater imaging setup for light field experimentation, with (a) the Raytrix R5 camera contained in (b) a custom underwater housing. Calibration through the flat viewport proved reliable in laboratory experimentation.

4.5.3 Results

Figure 4.14 shows the full reconstructed 3D model with and without inclusion of the light propagation compensation for qualitative comparison. Note the restoration of red light in the visual texture of Fig. 4.14b. The power of the proposed approach is that not only does it improve tracking and modeling, but the resulting models are more visually appealing and more similar to those gathered in air.

The resulting model was stored and then aligned to the ground truth model through ICP. The distance metric described above was then used to measure the geometric accuracy with and without the proposed water column attenuation compensation. Note that this distance metric is typically used to express the structural or geometric differences between two aligned models. However, this same approach can be expanded to measure attribute distance (e.g. color, normal, etc.) by modifying the function $d(p, p')$ to express things like photometric distance [89]. Here we will also evaluate photometric consistency with the ground truth using this formulation. Results are shown in Tables 4.5 and 4.6 for geometric and photometric comparisons, respectively.

Table 4.5: Hausdorff geometric distance between reconstructed model and ground truth.

	Uncorrected (m)	**Proposed method (m)**
Max	0.117685	0.100227
Mean	0.026717	0.023818
RMS	0.037340	0.033677

(a) Results without incorporation of light propagation model

(b) Results with incorporation of light propagation model

Figure 4.14: Textured 3D models gathered in water in real-time using the proposed approach. The resolution of the models was high enough to detect the ridges on the back of the target bolstering support for this approach as a pathway to real-time grasping and manipulation underwater. Note in (a) the absence of and in (b) the compensation for the red spectrum light typically lost in the water.

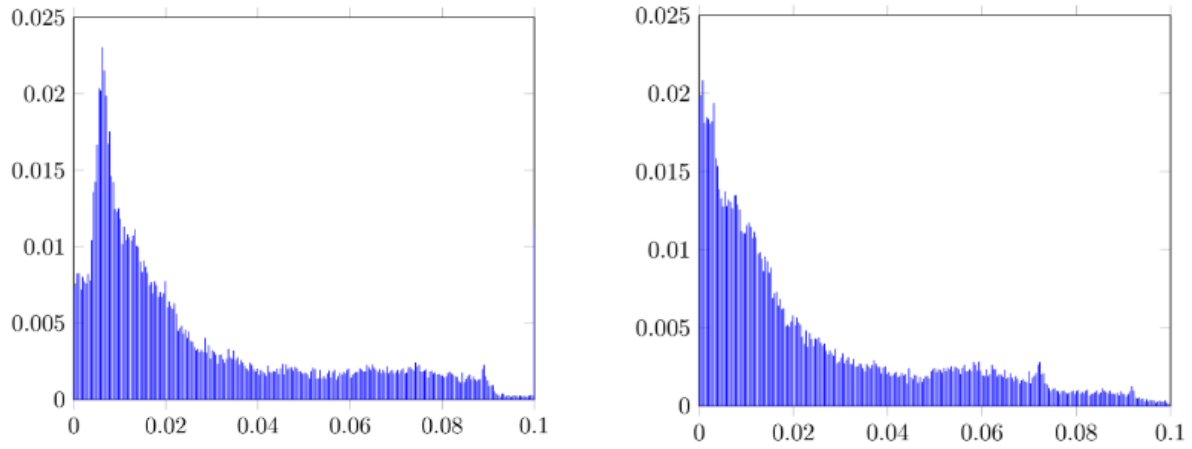
The uncorrected model is obtained by performing real-time dense 3D reconstruction with the light field camera in water, with no compensation for water column effects, whereas the proposed method performs the reconstruction while simultaneously accounting for these effects using the light propagation model explained above. The maximum, mean, and root mean square geometric Hausdorff distances are presented in meters from the ground truth model obtained in air with a structured light sensor. The photometric Hausdorff distances are presented as pixel intensity units across all three wavelengths (red, green, and blue). The resulting 3D model obtained with the proposed method shows improvements in both geometric and photometric accuracy compared to the model obtained without water column

Table 4.6: Hausdorff photometric distance between reconstructed model and ground truth.

	Uncorrected (pixel intensity)	**Proposed method (pixel intensity)**
Max	1.56031	1.55126
Mean	0.977898	0.952538
RMS	1.02169	0.997918

compensation. For the purposes of this comparison, we used the same light field data as input to the uncorrected method and our proposed method to produce the corrected and uncorrected results shown here. Note the decrease in geometric error facilitated by improved photometric matching in the fusion process.

In addition, the histograms of these distances appear in Fig. 4.15, demonstrating the distribution of normalized frequency, or count, of individual point-to-plane distances of vertices across the entire model. Note that the error between the resulting 3D model and ground truth is consistently lower with our proposed method for compensation of water column effects compared to the model produced with no correction.



(a) No correction

(b) With correction

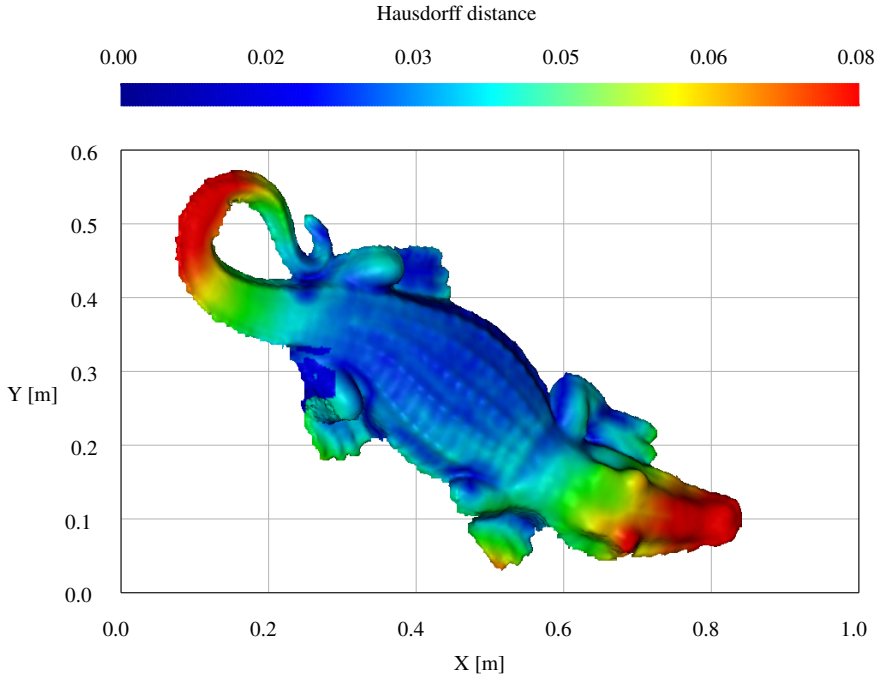
Figure 4.15: Plot of Hausdorff geometric distance error (m) between the resulting 3D models and ground truth, with and without the light propagation model. Note the decrease in error in (b) where there is an increase in occurrence of low distance faces indicating the model more closely matches reality.

In addition to the overall distance metrics, we present the spatial layout of error in Fig. 4.16a and Fig. 4.16b for the uncorrected and attenuation corrected modeling procedure, respectively. Note the head of the model shows a decrease in error, or distance from the ground truth, particularly around the snout, in the water column corrected approach. This is attributable to the improved tracking and the decrease in falsely induced curvature as the camera moved along the flat model.

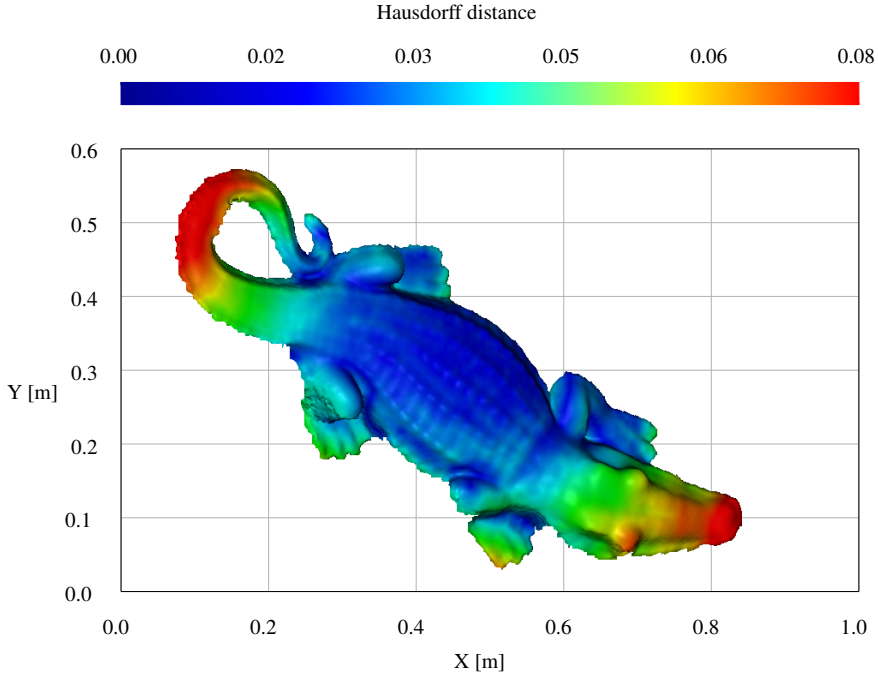
4.5.4 Discussion

Plenoptic cameras are still a nascent field but these results suggest that they offer a promising alternative to other sensors for underwater perception. As they improve, so too will their applicability and our understanding of how they can be used in the robotics do-

main. Still, there are limitations to our proposed approach, which lead to motivations for future work. The light propagation model presented here was chosen for its relative ease of implementation and integration into the reconstruction framework. Additionally, it runs in real-time as the correction is computed through a single operation to account for both attenuation and scattering effects. However, it is important to note that this model does not account for several other water column effects or for refraction through the flat viewport housing. These unmodeled effects must be realized when considering the limitations of the resulting 3D model and future work will focus on mitigating these effects. Next, as we rely on external software for light field processing to provide online images and depth maps from the plenoptic camera, we neglect the influence of the water column on depth map estimation and only correct for these effects during tracking and fusing. Future work will integrate water column compensation in the light field processing step to improve results. Lastly, our experiments presented here are controlled in-lab experiments with pure water only. It is important to extend this testing to a variety of water conditions and environments in order to ensure development of a robust approach that can be transferred to implementation on ROVs and AUVs in the field.



(a) Results without incorporation of light propagation model



(b) Results with incorporation of light propagation model

Figure 4.16: Spatial layout of Hausdorff distance between uncorrected and water column corrected model and ground truth. The color bar indicates the degree to which the gathered model is consistent with the ground truth. The values are projected onto the ground truth model for visualization. Note the decrease in error around the snout of the model. This highlights the higher robustness to drift in the lighting corrected fusion as this was the last area to be scanned in this run.

4.6 Conclusion

Overall this chapter presented novel methods to advance towards real-time underwater 3D reconstruction. We have explored two promising directions. The first approach leveraged deep learning for stereo camera systems to achieve online dense depth estimation and color correction as input into a tracking and fusion 3D reconstruction framework. The second approach integrated a plenoptic camera into a real-time 3D reconstruction framework, with compensation for water column effects to improve accuracy in color and structure of the output 3D model. Each of these contributions provides a basis for employing the current state-of-the-art in terrestrial vision and 3D reconstruction to the underwater domain. Results presented with data gathered show promise for achieving accurate 3D models of submerged objects with our proposed pipelines. Future work will aim to achieve online dense 3D reconstruction on a robot in situ, which is key for enabling full autonomy for subsea robotic systems.

CHAPTER 5

Conclusions & Future Directions

5.1 Conclusions

This dissertation presented novel hybrid model-based, data-driven frameworks to address key challenges to perception of robotic systems in underwater environments. In particular, the developed methods integrated physics-based models and cross-disciplinary knowledge into learning-based frameworks to enable development of unsupervised and self-supervised methods for color correction and dense depth estimation of raw underwater images. Furthermore, a novel pipeline was implemented to demonstrate an application of deep neural networks (DNNs) to enable online underwater three-dimensional (3D) reconstruction with underwater stereo cameras. Lastly, we also explored the potential for computational imaging to enable real-time underwater 3D reconstruction. Methods developed throughout this thesis were validated on real data gathered in underwater environments. Software and datasets developed and collected in support of this dissertation have been made publicly available for the underwater vision community. While methods developed throughout this thesis were applied to the underwater domain, its aim is to work towards robust perception systems for deploying robotic platforms in natural and unstructured environments often encountered in field robotics.

5.2 Future Directions

There are several immediate goals for future work in order to enable direct application of the proposed methods to address practical challenges across marine science and engineering. In particular, future work should focus on optimizing the required computation time and memory for the developed methods in order to allow practical deployment of these methods on a robot in situ. Additionally, further evaluation will be required to determine the ideal operational ranges for these methods across sites of varying water quality and environmental

conditions. Overall, we believe the work presented here shows promise for achieving real-time underwater 3D reconstruction surveys across small areas in the near future, while more effort will be needed to expand application to large scale surveys featuring varying water conditions across the site.

Ultimately, this work demonstrated the potential for hybrid model-based, data-driven systems to advance state-of-the-art for applications across field robotics. Future research will focus on *generalizability*, *transparency*, and *efficiency* to enable deployment of robust real-time autonomous systems in complex, dynamic environments.

Eliminating the Need for Hand-labeled Training Data

Currently, state-of-the-art methods rely on hand-annotated labels for network training. A major goal of future research should be to develop learning-based frameworks that do not rely on hand-labeled data. Throughout this thesis, novel methods were developed to incorporate physics-based models and cross-disciplinary knowledge to present unsupervised or self-supervised approaches that do not require labels at all. Another promising direction is sim-to-real: training on simulated data and testing on real data. Solving sim-to-real has the potential to enable many new applications of learning-based systems since labels can be generated cheaply and efficiently. Ultimately, developing methods that do not rely on hand-labeled data could enable the translation of learning-based frameworks to solve problems in new domains in order to facilitate advancement of robotic capabilities within these domains that are not possible with existing methods.

Transparent Network Structure with Interpretable Output

Currently, DNNs are not interpretable to human users. State-of-the-art supervised networks are trained with large datasets with paired samples of input and labels. Supervised DNNs take these pairings and learn to model highly nonlinear mappings from input to the output label. However, it is currently not possible to interpret how the network gets from input to output. This makes it challenging to determine how robust a network is to changing input. Demonstrating interpretability, or transparency, of learned models is critical to establishing measures of trust and safety in learning-based systems. One advantage of leveraging domain-aware knowledge and structure of model-based approaches is that there is already a rich understanding of how to interpret intermediate components and final output of these models. Future research should further explore the incorporation of physics-based knowledge and model-based structure into learning frameworks to enable interpretability of output from DNNs without sacrificing accuracy of the learned model.

Learning for Real-time Computational Imaging

Computational imaging systems leverage high-performance computing and processing power to provide rich sensory information. This rich output has potential to enable novel tasks in robotics. However, current requirements for processing power and data storage from these systems are not practical for deployment of real-time robotic systems in remote environments. Recent work has demonstrated the potential for deep learning frameworks to reduce these requirements in developing real-time computational imaging systems. Future research should focus on developing learning-based frameworks that enable practical application of non-traditional sensing platforms on mobile robots in remote environments. Future research should also focus on developing methods for image understanding to perform real-time high-level processing of this rich sensory output.

APPENDICES

APPENDIX A

Datasets

The following appendix describes the datasets collected throughout this thesis.

Jamaica – A dataset of stereo images collected in Port Royal, Jamaica in April 2015. The images were collected using Australian Centre for Field Robotics's (ACFR) Diver Rig platform featuring one color machine vision camera and one monochrome machine vision camera. The dataset contains a total of 6500 images gathered in a single dive. The maximum depth from the seafloor is approximately $1.5m$.

MHL – A dataset of stereo images collected in a pure water test tank in the Marine Hydrodynamics Laboratory (MHL) at the University of Michigan in August 2016. The images were collected using the DROP Stereo Camera platform featuring one color machine vision camera and one monochrome machine vision camera. A total of over 7000 underwater images are compiled from this survey. Ground truth structure and color are available for this dataset.

HIMB #1 – A dataset of stereo images collected at the Hawaii Institute of Marine Biology (HIMB) in January 2018. The images were collected using the DROP BlueROV platform featuring two color machine vision cameras. There are a total of 1371 images in this dataset. Images are rectified and cropped to a resolution of 645 x 515. Ground truth structure and color are available for this dataset.

HIMB #2 – A dataset of stereo images collected at the Hawaii Institute of Marine Biology (HIMB) in January 2018. The images were collected using the DROP BlueROV platform featuring two color machine vision cameras. There are a total of 2676 images in this dataset. Images are rectified and cropped to a resolution of 645 x 515. Ground truth structure and

color are available for this dataset.

SIMS – A dataset of stereo images collected at the Sydney Institute of Marine Science (SIMS) in December 2018. The images were collected using the DROP BlueROV platform featuring two color machine vision cameras. Images are rectified and cropped to a resolution of 2368 x 1920. Ground truth structure and color are available for this dataset.

APPENDIX B

Software

The following appendix describes the open-source software developed throughout this thesis.

WaterGAN – The project page for WaterGAN is available at `https://github.com/kskin/WaterGAN`.

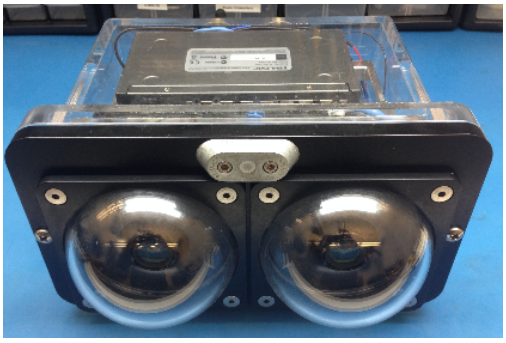
UWStereoNet – The project page for UWStereoNet is available at `https://github.com/kskin/UWStereo`.

APPENDIX C

Hardware

The following appendix describes the hardware systems developed throughout this thesis.

DROP Stereo Camera – The DROP Stereo Camera platform (Fig. C.1a) features one monochrome Prosilica GT1380 machine vision camera and one color Prosilica GT1380C machine vision camera. The cameras are housed in a custom underwater stereo camera housing. For imaging experiments, the DROP Stereo Camera can be mounted on an imaging rig shown in Fig. C.1b.



(a) DROP Stereo Camera system in custom underwater housing

(b) DROP stereo camera system mounted on imaging rig

Figure C.1: The DROP stereo camera system was used in pure tank imaging surveys.

DROP Diver Rig – The DROP Diver Rig (Fig. C.2) features two color Prosilica GT1380C

machine vision cameras and one Point Grey Blackfly BFLY-PGE-50S5C fisheye camera. The diver rig has a Microstrain 3dm-gx5-45 IMU and a depth sensor.

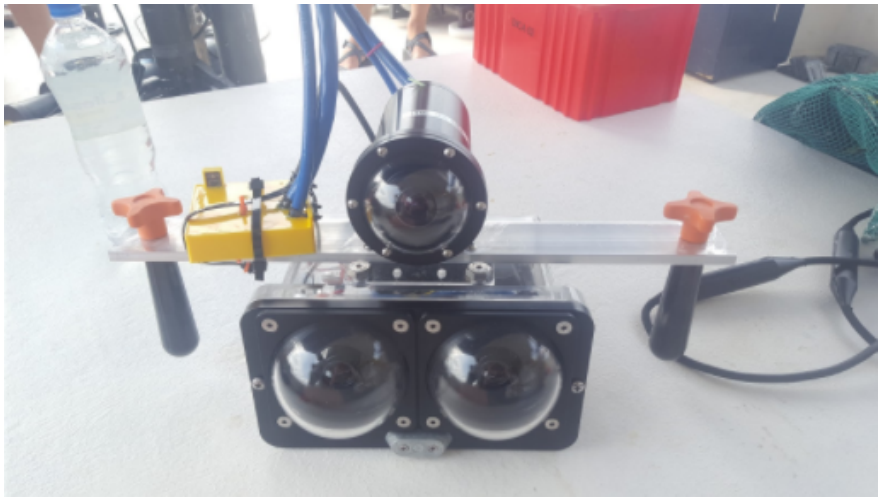
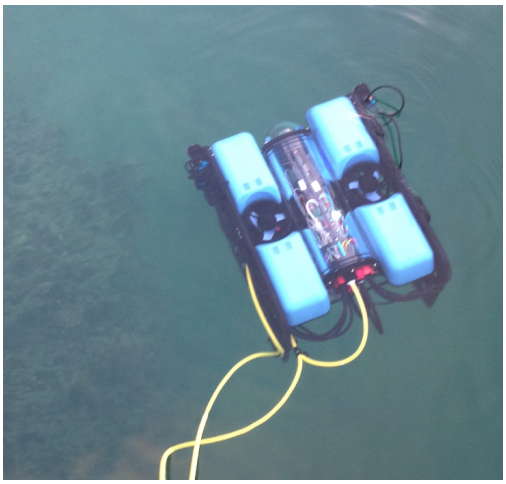


Figure C.2: DROP Diver Rig

DROP BlueROV2 – The DROP BlueROV2 (Fig. C.3a) consists of a Blue Robotics BlueROV2 platform with a custom stereo vision payload that features two downward-facing machine vision cameras (Fig. C.3b).



(a) BlueROV2

(b) Underside of the BlueROV2 showing downward facing stereo cameras.

Figure C.3: The DROP BlueROV2 was developed for stereo imaging surveys in shallow water environments.

REFERENCES

REFERENCES

[1] T. Bishop, P. Tuddenham, and M. Ryan, "Then and now: The hms challenger expedition and the "mountains in the sea" expedition," *National Oceanic and Atmospheric Administration*, 2003. [Online]. Available: `https://oceanexplorer.noaa.gov/explorations/03mountains/background/challenger/challenger.html`.

[2] National Centers for Environmental Information, "Trackline geophysical data," *National Oceanic and Atmospheric Administration*, 2018. [Online]. Available: `https://maps.ngdc.noaa.gov/viewers/geophysics/`.

[3] C. Paull, "Imaging the k-t boundary: Increasing resolution," *Schmidt Ocean Institute*, 2013. [Online]. Available: `https://schmidtocean.org/cruise-log-post/increasing-resolution/`.

[4] R. E. Hansen, "Introduction to sonar," *INF-GEO4310 Course Material, University of Oslo*, 2012.

[5] Kongsberg Maritime, "Multibeam echosounder, maximum depth 11000 m," 2018. [Online]. Available: `https://www.km.kongsberg.com`.

[6] J. Dillon, "Seeing with sound: Why sonar resolution matters for seabed mapping," 2018. [Online]. Available: `http://krakenrobotics.com/seeing-with-sound-why-sonar-resolution-matters-for-seabed-mapping/`.

[7] D. McGowen and R. Morris, "Choosing side scan sonar frequencies," *Sea Technology*, 2013.

[8] Kongsberg Maritime, "Synthetic aperture sonar," 2018. [Online]. Available: `https://www.km.kongsberg.com`.

[9] T. P. Hughes, J. T. Kerry, M. Álvarez-Noriega, J. G. Álvarez-Romero, K. D. Anderson, A. H. Baird, R. C. Babcock, M. Beger, D. R. Bellwood, R. Berkelmans, *et al.*, "Global warming and recurrent mass bleaching of corals," *Nature*, vol. 543, no. 7645, pp. 373–377, 2017.

[10] T. O.A.X.C. S. Survey, "Before & after," [Online]. Available: `https://www.coralreefimagebank.org/before-after`.

[11] W. W. C. Gieskes, C. Veth, A. Woehrmann, and M. Graefe, "Secchi disc visibility world record shattered," *Eos, Transactions, American Geophysical Union*, vol. 68, no. 9, pp. 123–123, 2017.

[12] C. D. Mobley, *Light and Water: Radiative Transfer in Natural Waters*. New York: Academic Press, 1994.

[13] B. Christenson, K. Németh, D. Rouwet, F. Tassi, J. Vandemeulebrouck, and J. C. Varekamp, "Volcanic lakes," *Volcanic Lakes*, pp. 1–20, 2015.

[14] Y. Schechner and N. Karpel, "Clear underwater vision," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004, pp. 536–543.

[15] M. Sheinin and Y. Y. Schechner, "The next best underwater view," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 3764–3773.

[16] S. Chandrasekhar, *Radiative Transfer*. 2013.

[17] D. Akkaynak and T. Treibitz, "A revised underwater image formation model," in *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 6723–6732.

[18] DARPA, "Darpa urban challenge," 2007. [Online]. Available: `http://archive.darpa.mil/grandchallenge/`.

[19] Z. Zhang, "Microsoft kinect sensor and its effect," *MultiMedia*, vol. 19, no. 2, pp. 4–10, 2012.

[20] J. S. Jaffe, "Development of a laser line scan LIDAR imaging system for AUV use," University of California, San Diego, Tech. Rep., 2011.

[21] M. Johnson-Roberson, M. Bryson, B. Douillard, O. Pizarro, and S. B. Williams, "Out-of-core efficient blending for underwater georeferenced textured 3d maps," in *Proceedings of the 2013 Fourth International Conference on Computing for Geospatial Research and Application (COM.Geo)*, 2013, pp. 8–15.

[22] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[23] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, Jun. 2008.

[24] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *Proceedings of the 2011 International Conference on Computer Vision (ICCV)*, 2011, pp. 2564–2571.

[25] P. H. K. Berthold and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17.1-3, pp. 185–203, 1980.

[26] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems 27 (NIPS 2014)*, 2014, pp. 2672–2680.

[27] A. Jordt, "Underwater 3d reconstruction based on physical models for refraction and underwater light propagation," PhD Dissertation, Kiel University, 2013.

[28] J. S. Jaffe, "Computer modeling and the design of optimal underwater imaging systems," *IEEE Journal of Oceanic Engineering*, vol. 15, no. 2, pp. 101–111, 1990.

[29] M. Bryson, M. Johnson-Roberson, O. Pizarro, and S. B. Williams, "True color correction of autonomous underwater vehicle imagery," *Journal of Field Robotics*, pp. 853–874, 2015.

[30] M. Johnson-Roberson, M. Bryson, A. Friedman, O. Pizarro, G. Troni, P. Ozog, and J. C. Henderson, "High-resolution underwater robotic vision-based mapping and 3d reconstruction for archaeology," *Journal of Field Robotics*, pp. 625–643, 2016.

[31] M. Bryson, M. Johnson-Roberson, O. Pizarro, and S. B. Williams, "Colour-consistent structure-from-motion models using underwater imagery," *Robotics: Science and Systems*, 2013.

[32] K. A. Skinner, E. Iscar, and M. Johnson-Roberson, "Automatic color correction for 3D reconstruction of underwater scenes," in *Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 5140–5147.

[33] N. Carlevaris-Bianco, A. Mohan, and R. M. Eustice, "Initial results in underwater single image dehazing," in *Proceedings of OCEANS 2010 MTS/IEEE SEATTLE*, Seattle, WA, USA, 2010, pp. 1–8.

[34] P. Drews Jr., E. do Nascimento, F. Moraes, S. Botelho, and M. Campos, "Transmission estimation in underwater single images," in *Proceedings of the 2013 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2013, pp. 825–830.

[35] P. D. Jr., E. R. Nascimento, S. S. C. Botelho, and M. F. M. Campos, "Underwater depth estimation and image restoration based on single images," *IEEE Computer Graphics and Applications*, vol. 36, no. 2, pp. 24–35, 2016.

[36] Y.-S. Shin, Y. Cho, G. Pandey, and A. Kim, "Estimation of ambient light and transmission map with common convolutional architecture," in *Proceedings of OCEANS 2016 MTS/IEEE Monterey*, Monterey, CA, 2016, pp. 1–7.

[37] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[38] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, "Learning from simulated and unsupervised images through adversarial training," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2242–2251.

[39] L. Sixt, B. Wild, and T. Landgraf, "Rendergan: Generating realistic labeled data," *Frontiers in Robotics and AI*, vol. 5, p. 66, 2018.

[40] B. L. McGlamery, "Computer analysis and simulation of underwater camera system performance," UC San Diego, Tech. Rep., 1975.

[41] L. Lopez-Fuentes, G. Oliver, and S. Massanet, "Revisiting image vignetting correction by constrained minimization of log-intensity entropy," in *Advances in Computational Intelligence*, Springer International Publishing, June 2015, pp. 450–463.

[42] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for scene segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 2481–2495, 2017.

[43] X. Mao, C. Shen, and Y.-B. Yang, "Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections," in *Advances in Neural Information Processing Systems 29 (NIPS 2016)*, 2016, pp. 2802–2810.

[44] V. Jain, J. F. Murray, F. Roth, S. Turaga, V. Zhigulin, K. L. Briggman, M. N. Helmstaedter, W. Denk, and H. S. Seung, "Supervised learning of image restoration with convolutional networks," in *Proceedings of the 2007 IEEE International Conference on Computer Vision (ICCV)*, 2007, pp. 1–8.

[45] A. Janoch, S. Karayev, Y. Jia, J. T. Barron, M. Fritz, K. Saenko, and T. Darrell, "A category-level 3-d object dataset: Putting the kinect to work," in *Proceedings of the 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2011, pp. 1168–1174.

[46] K. Lai, L. Bo, and D. Fox, "Unsupervised feature learning for 3d scene labeling," in *Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 3050–3057.

[47] N. Silberman and R. Fergus, "Indoor scene segmentation using a structured light sensor," in *Proceedings of the 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2011, pp. 601–608.

[48] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon, "Scene coordinate regression forests for camera relocalization in rgb-d images," in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 2930–2937.

[49] O. Pizarro, A. Friedman, M. Bryson, S. B. Williams, and J. Madin, "A simple, fast, and repeatable survey method for underwater visual 3d benthic mapping and monitoring," *Ecology and Evolution*, vol. 7, no. 6, pp. 1770–1782, 2017.

[50] T. Whelan, S. Leutenegger, R. F. Salas-Moreno, B. Glocker, and A. J. Davison, "Elasticfusion: Dense slam without a pose graph," in *Robotics: Science and Systems (RSS)*, 2015.

[51] J. Li, K. A. Skinner, R. M. Eustice, and M. Johnson-Roberson, "Watergan: Unsupervised generative network to enable real-time color correction of monocular underwater images," *IEEE Robotics and Automation Letters (RA-L)*, vol. 3, no. 1, pp. 387–394, 2018.

[52] C. Fabbri, M. J. Islam, and J. Sattar, "Enhancing underwater imagery using generative adversarial networks," in *Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, Queensland, Australia, 2018, pp. 7159–7165.

[53] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4040–4048.

[54] W. Luo, A. G. Schwing, and R. Urtasun, "Efficient deep learning for stereo matching," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 5695–5703.

[55] Y. Feng, Z. Liang, and H. Liu, "Efficient deep learning for stereo matching with larger image patches," in *Proceedings of the 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics*, 2017, pp. 1–5.

[56] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A convolutional network for real-time 6-dof camera relocalization," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, Washington, DC, USA, 2015, pp. 2938–2946.

[57] P.-H. Huang, K. Matzen, J. Kopf, N. Ahuja, and J.-B. Huang, "Deepmvs: Learning multi-view stereopsis," in *Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 2821–2830.

[58] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 270–279.

[59] Y. Zhong, Y. Dai, and H. Li, "Self-supervised learning for stereo matching with self-improving ability," *arXiv preprint arXiv:1709.00930*, 2017.

[60] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6612–6619.

[61] J. Zhang, K. A. Skinner, R. Vasudevan, and M. Johnson-Roberson, "Dispsegnet: Leveraging semantics for end-to-end learning of disparity estimation from stereo imagery," *IEEE Robotics and Automation Letters (RA-L)*, vol. 4, no. 2, pp. 1162–1169, 2019.

[62] D. Akkaynak, T. Treibitz, T. Shlesinger, Y. Loya, R. Tamir, and D. Iluz, "What is the space of attenuation coefficients in underwater computer vision?" In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 568–577.

[63] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2881–2890.

[64] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[65] J. Zbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *Journal of Machine Learning Research*, vol. 17, no. 1-32, pp. 2287–2318, 2016.

[66] B. Li, X. Peng, Z. Wang, J. Xu, and D. Feng, "Aod-net: All-in-one dehazing network," in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 4780–4788.

[67] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2242–2251.

[68] W. V. Barbosa, H. G. Amaral, T. L. Rocha, and E. R. Nascimento, "Visual-quality-driven learning for underwater vision enhancement," in *Proceedings of the 2018 IEEE International Conference on Image Processing (ICIP)*, 2018, pp. 3933–3937.

[69] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 3213–3223.

[70] H. Hirschmuller, "Accurate and efficient stereo processing by semi-global matching and mutual information," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2005, pp. 807–814.

[71] D. Girardeau-Montaut, "Cloud compare—3d point cloud and mesh processing software," *Open Source Project*, 2015.

[72] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, Feb. 1992.

[73] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "Dtam: Dense tracking and mapping in real-time," in *Proceedings of the 2011 IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 2320–2327.

[74] M. Keller, D. Lefloch, M. Lambers, S. Izadi, T. Weyrich, and A. Kolb, "Real-time 3d reconstruction in dynamic scenes using point-based fusion," in *Proceedings of the 2013 International Conference on 3D Vision (3D-V)*, 2013, pp. 1–8.

[75] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *Proceedings of the 2011 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2011, pp. 127–136.

[76] T. Whelan, M. Kaess, M. Fallon, H. Johannsson, J. Leonard, and J. McDonald, "Kintinuous: Spatially extended kinectfusion," in *Robotics: Science and Systems Workshops (RSS Workshops)*, 2012.

[77] M. Johnson-Roberson, O. Pizarro, S. B. Williams, and I. Mahon, "Generation and visualization of large-scale three-dimensional reconstructions from underwater robotic surveys," *Journal of Field Robotics*, vol. 27, no. 1, pp. 21–51, 2010.

[78] A. Jordt-Sedlazeck and R. Koch, "Refractive calibration of underwater cameras," in *Proceedings of the 2012 European Conference on Computer Vision (ECCV)*, vol. 7576, 2012, pp. 846–859.

[79] A. Jordt-Sedlazeck and R. Koch, "Refractive structure-from-motion on underwater images," in *Proceedings of the 2013 IEEE International Conference on Computer Vision (ICCV)*, 2013, pp. 57–64.

[80] C. Perwass and L. Wietzke, "Single lens 3d-camera with extended depth-of-field," in *IS&T/SPIE Electronic Imaging*, 2012.

[81] Lytro, "Lytro," [Online]. Available: `https://www.lytro.com/`.

[82] E. Adelson and J. Wang, "Single lens stereo with a plenoptic camera," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 99 –106, 1992.

[83] F. Dong, S.-H. Ieng, X. Savatier, R. Etienne-Cummings, and R. Benosman, "Plenoptic cameras in real-time robotics.," *International Journal of Robotics Research*, vol. 32, no. 2, pp. 206–217, 2013.

[84] D. G. Dansereau, I. Mahon, O. Pizarro, and S. B. Williams, "Plenoptic flow: Closed-form visual odometry for light field cameras," in *Proceedings of the 2011 IEEE/RSJ*

International Conference on Intelligent Robots and Systems (IROS), 2011, pp. 4455–4462.

[85] N. Zeller, F. Quint, and U. Stilla, "Narrow field-of-view visual odometry based on a focused plenoptic camera," *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. II-3/W4, pp. 285–292, 2015.

[86] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, and M. H. Gross, "Scene reconstruction from high spatio-angular resolution light fields," *ACM Transactions on Graphics*, vol. 32, no. 4, 73:1–73:12, 2013.

[87] M. Tao, S. Hadap, J. Malik, and R. Ramamoorthi, "Depth from combining defocus and correspondence using light-field cameras," in *Proceedings of the 2013 IEEE International Conference on Computer Vision (ICCV)*, 2013, pp. 673–680.

[88] Z. Zhang, "Iterative point matching for registration of free-form curves and surfaces," *International Journal of Computer Vision*, vol. 13, no. 2, pp. 119–152, 1994.

[89] M. Roy, S. Foufou, and F. Truchetet, "Mesh comparison using attribute deviation metric," *International Journal of Image and Graphics*, vol. 4, no. 1, pp. 127–140, 2004.

[90] P. Cignoni, M. Callieri, M. Corsini, M. Dellepiane, F. Ganovelli, and G. Ranzuglia, "MeshLab: an Open-Source Mesh Processing Tool," in *Eurographics Italian Chapter Conference*, V. Scarano, R. D. Chiara, and U. Erra, Eds., The Eurographics Association, 2008.

[91] P. Cignoni, C. Rocchini, and R. Scopigno, "Metro: Measuring error on simplified surfaces," in *Computer Graphics Forum*, Blackwell Publishers, vol. 17, 1998, pp. 167–174.

[92] Agisoft, "Photoscan," [Online]. Available: `https://www.agisoft.com/`.

[93] D. Dansereau, "Plenoptic signal processing for robust vision in field robotics," PhD Dissertation, Australian Centre for Field Robotics, School of Aerospace, Mechanical and Mechatronic Engineering, The University of Sydney, 2014.

[94] Raytrix, "Raytrix," [Online]. Available: `https://raytrix.de/`.