

UNIVERSIDADE FEDERAL DE MINAS GERAIS
Instituto de Ciências Exatas
Programa de Pós-Graduação em Ciência da Computação

Walysson Vital Barbosa

**Sistema para Restauração de Imagens Subaquáticas Ponta a Ponta
Orientado pela Qualidade Usando Uma Rede Neural Convolucional
Auto-Supervisionada**

Belo Horizonte
2020

Walysson Vital Barbosa

**Sistema para Restauração de Imagens Subaquáticas Ponta a Ponta
Orientado pela Qualidade Usando Uma Rede Neural Convolutiva
Auto-Supervisionada**

Versão Final

Dissertação apresentada ao Programa de Pós-Graduação em
Ciência da Computação da Universidade Federal de Minas
Gerais, como requisito parcial à obtenção do título de Mestre
em Ciência da Computação.

Orientador: Erickson Rangel do Nascimento
Coorientador: Mário Fernando Montenegro Campos

Belo Horizonte
2020

Walysson Vital Barbosa

**Quality-driven End-to-end Restoration System for Underwater Images using
a Self-supervised Convolutional Neural Network**

Final Version

Thesis presented to the Graduate Program in Computer Science of the Federal University of Minas Gerais in partial fulfillment of the requirements for the degree of Master in Computer Science.

Advisor: Erickson Rangel do Nascimento
Co-Advisor: Mário Fernando Montenegro Campos

Belo Horizonte
2020

Barbosa, Walysson Vital.

B238s Sistema para restauração de imagens subaquáticas ponta a ponta orientado pela qualidade usando uma rede neural convolucional auto-supervisionada [manuscrito] / Walysson Vital Barbosa. — 2020.
79 f. il.; 29 cm.

Orientador: Erickson Rangel do Nascimento.
Coorientador: Mário Fernando Montenegro Campos.
Dissertação (mestrado) - Universidade Federal de Minas Gerais – Departamento de Ciência da Computação
Referências: f.74-79.

1. Computação – Teses. 2. Processamento de imagens -- Técnicas digitais – Restauração e conservação -Teses. 3. Visão subaquática – Teses. 4. Redes neurais convolucionais – Teses. I. Nascimento, Erickson Rangel do II. Campos, Mário Fernando Montenegro. III. Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Computação. IV.Título.

CDU 519.6* 82.10(043)



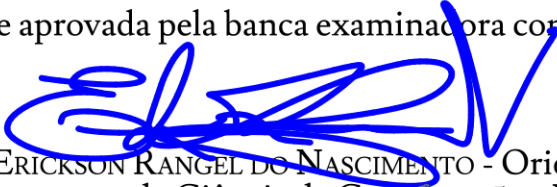
UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Sistema para Restauração de Imagens Subaquáticas Ponta a Ponta, Orientado pela Qualidade, Usando Uma Rede Neural Convolutiva Auto-Supervisionada.

WALYSSON VITAL BARBOSA

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:


PROF. ERICKSON RANGEL DO NASCIMENTO - Orientador
Departamento de Ciência da Computação - UFMG


PROF. MARIO FERNANDO MONTENEGRO CAMPOS - Coorientador
Departamento de Ciência da Computação - UFMG


PROF. PAULO LILLES JORGE DREWS JUNIOR
Centro de Ciências Computacionais - FURG


PROF. FLÁVIO LUIS CARDEAL PÁDUA
Departamento de Computação - CEFET-MG

Belo Horizonte, 4 de Fevereiro de 2020.

Acknowledgments

First of all, I thank the greater entity that rules the universe, for having allowed me to exist. Not only do we exist, at the same time there are gifts that must be polished as we go through experiences. Learning from mistakes and perfecting the moves that lead us to achieve our goals.

I would like to especially thank the employees of the Exact Sciences Institute at UFMG, who helped in the process of building the second underwater image dataset, described in this thesis. From the organization of the environment to the removal of equipment after the end of experiments.

To the research funding institutions CAPES, CNPq and FAPEMIG, for the financing that enabled the acquisition of resources used throughout this research.

To my colleagues and friends at Verlab and from the Computer Science Department, especially Alan Deivite, Bruna Frade, Camila Laranjeira, Daniel Balbino, Hugo Oliveira, Jéssica Sena, Michel Melo Virgínia Mota and Washington Ramos, for always being available to answer questions and help in tasks related to my Master's activities. To Maurício Ferrari, the laboratory technician, who actively assisted us during the acquisition of the datasets, one of the main contributions of this work.

To Henrique Amaral and Thiago Lages, for their contribution to much of this project development. Both helped in two considerably important stages of this research. Assisting me in acquiring the images that make up the underwater image datasets.

To the examining board, for its dedication and attention while reading and evaluating the work developed during my Master's. And to Professor Erickson Nascimento, for his patience in guiding me throughout this experience, providing me great teachings that I will surely take with me on my future endeavors.

Finally, my family and friends. These who have always been by my side at all times, supporting my decisions and encouraging me to always seek the best on this journey, which still holds countless achievements. In particular, to my parents and siblings, whom I love unconditionally.

*“Without a sense of purpose,
we’re setting up to fail.”*
(Imagine Dragons, 2012)

Resumo

O avanço tecnológico tem nos permitido extrair informações e analisar os mais variados tipos de ambientes. O meio subaquático está incluído nesse conjunto de lugares e tem sido amplamente estudado nos últimos anos devido a áreas emergentes de pesquisas subaquáticas. No entanto, existem algumas razões pelas quais estudar neste ambiente se torna um desafio. Estruturas presentes debaixo d'água, como as de sítios arqueológicos, muitas vezes não podem ser movidas para fora desse meio, pois podem perder suas propriedades e, conseqüentemente, serem danificadas. Além disso, imagens tiradas nesses ambientes possuem qualidade muito baixa em comparação com imagens de fora d'água. O ambiente subaquático causa diversos efeitos durante o processo de aquisição da imagem. Raios de luz são espalhados e absorvidos enquanto viajam até o sensor da câmera. A presente dissertação propõe um método de restauração de imagens de cenas subaquáticas baseado na extração de parâmetros utilizando redes neurais convolucionais (CNNs) combinada com métricas de qualidade de imagem. Os parâmetros extraídos da imagem subaquática original são aplicados ao modelo de formação da imagem para recuperar a radiância original da imagem. Não são necessários dados rotulados, já que a rede é treinada com base apenas nas métricas de qualidade calculadas usando as imagens subaquáticas original e restaurada. A metodologia proposta se sobressaiu em 60% dos casos em comparação às demais abordagens apresentadas quando aplicadas na restauração de imagens subaquáticas, levando em consideração a métrica UCIQE. Além disso, dois conjuntos de imagens subaquáticas são apresentados, adquiridos num processo planejado e direcionado ao problema de restauração de imagens subaquáticas.

Palavras-chave: Restauração de Imagens, Visão Subaquática, Redes Neurais Convolucionais, Métricas de Qualidade de Imagem

Abstract

Advances in technology have allowed humans to delve into the depths of Earth and to study the outer space, even if our resources are not sufficient to help us answer all questions about each one of these environments. The underwater environment is one of those places, which has been vastly studied in past years due to the increasing use of underwater research locations. However, there are a few reasons why studying this environment is challenging. In most cases, structures located underwater cannot be moved out of this medium as they can lose their properties and be damaged. Moreover, images taken in these environments have very poor quality in comparison to images from out of water places. The water medium causes various effects during the image acquisition process. Rays of light are scattered and absorbed as they travel to the camera. This thesis proposes an underwater image restoration method based on convolutional neural networks and image quality metrics, the former being considered universal function approximators. Features extracted from the original underwater image are applied to the inverse image formation model in order to recover the original image radiance. No labeled data is needed as the network is trained based only in the quality metrics computed using the original and restored underwater images. In 60% of the cases, our proposed methodology performs better than the techniques applied to the improvement of underwater images, taking into consideration the UCIQE metric. Additionally, two underwater image datasets are presented, which were acquired on a planned process, focusing on underwater image restoration purposes.

Keywords: Image Restoration, Underwater Vision, Convolutional Neural Networks, Image Quality Metrics

List of Figures

1.1	Disparity between objects distance from observer.	17
1.2	SUN underwater dataset sample.	19
2.1	Contrast-based enhancement.	23
2.2	Enhancement results using filter banks and a multi-objective function.	24
2.3	Enhancements using CLAHE and USM techniques	24
2.4	Enhancement by using fusion of feature maps.	25
2.5	Dark channel prior based image restoration.	26
2.6	Non-local image dehazing.	27
2.7	Underwater image restoration using blurriness and background light.	28
2.8	Red sea degraded and recovered underwater scene images.	28
2.9	Multi-scale descattering.	29
2.10	MSCNNDehazing result.	30
2.11	WaterGAN restoration result.	31
2.12	DehazeNet architecture.	31
2.13	DehazeNet: successful outdoor image restoration.	32
2.14	DehazeNet: underwater image restoration attempt.	32
3.1	A simple ANN model.	40
3.2	Example of a RGB input image of a CNN	40
3.3	CNN overview.	41
3.4	Example of a filter of size 3×3 being convoluted through a patch of an image.	41
3.5	Example of a pooling filter of size 3×3 being convoluted through a patch of an image.	42
4.1	Resulting radiance that arrives to the camera is a combination of the percentage of light reflected from the scene and background light.	45
4.2	Diagram of our two-stage learning.	45
4.3	UNderwater image and its transmission map.	47
5.1	UVision18 RGB image samples.	51
5.2	UVision18 3D scenes.	52
5.3	<i>UVision18</i> synthetic sample rendered with Physically-Based Rendering Engine (PBRT).	53
5.4	Setting up of the UVision19 dataset.	55

5.5	Scenes available in the UVision19 dataset.	56
5.6	RGB-D image samples and their depth maps.	57
5.7	Underwater images with non-sifted green tea powder.	58
5.8	RGB image samples.	59
5.9	Pinhole camera model.	59
5.10	Kinect One calibration images.	60
5.11	Kinect One synthetic images.	62
6.1	Underwater images samples.	64
6.2	Underwater restorations using different approaches.	66
6.3	Experiments applied to UVision19 dataset images.	71

List of Tables

5.1	Spectral absorption and molecular scattering coefficients per light wavelength in a pure water medium, provided by Mobley [38].	53
6.1	Visual quality using Underwater Color Image Quality Evaluation (UCIQE) metric (best in bold).	65
6.2	Visual quality using UCIQE metric (best in bold) on UVision19 experiments. .	69

List of Acronyms

AI	Artificial Intelligence	38
ANN	Artificial Neural Networks	39
BP	Backpropagation Algorithm	42
CLAHE	Contrast-Limited Adaptive Histogram Equalization	23
CONV	Convolution Layer	41
CNN	Convolutional Neural Networks	18
DCP	Dark Channel Prior	25
FR	Full-Reference	36
FC	Fully-Connected Layer	42
GAN	Generative Adversarial Network	30
HVS	Human Visual System	36
ICIP	International Conference on Image Processing	20
IEEE	Institute of Electrical and Electronics Engineers	20
IQA	Image Quality Assessment	36
IQM	Image Quality Metrics	47
MSE	Mean Squared Error	46
NR	No-Reference	37
PBRT	Physically-Based Rendering Engine	9
POOL	Pooling Layer	42
PSF	Point-Spread Function	35
PSNR	Peak Signal-to-Noise Ratio	37
RELU	Rectified Linear Unit	42
ROS	Robot Operating System	55
RR	Reduced-Reference	37
SFM	Structure from Motion	29
SSIM	Structural Similarity Index	37
UDCP	Underwater Dark Channel Prior	26

USM	Unsharp Mask	23
UCIQE	Underwater Color Image Quality Evaluation	11
UUV	Unmanned Underwater Vehicles	18
VGS	Visual Gradient Similarity	37
VIF	Visual Information Fidelity	37
VSNR	Visual Signal-to-Noise Ratio	37

Contents

1	Introduction	16
1.1	Applications	17
1.2	Problem Definition	18
1.3	Thesis Statement	19
1.4	Contributions	19
1.5	Thesis Structure	20
2	Related Work	22
2.1	Image Processing	22
2.2	Image Formation Model	25
2.3	Machine Learning Based Approaches	30
3	Theoretical Foundations	33
3.1	Image Formation Process	33
3.2	Image Quality Metrics	36
3.3	Machine Learning Overview	38
4	Methodology	44
4.1	Image Formation Model	44
4.2	Transmission Map Estimation	46
4.3	Visual-Quality-Driven Learning	47
5	Datasets	50
5.1	UVision18 Dataset	51
5.2	UVision19 Dataset	54
6	Experiments	63
6.1	Evaluation Metrics	63
6.2	Experiments on UVision18	64
6.3	Experiments on UVision19	66
6.4	Parametrization	67
7	Conclusion	72
7.1	Future Work	73

Chapter 1

Introduction

With image processing and learning approaches rapidly evolving, the ability to make sense of what is going on in a single picture is improving in both scalability and accuracy. Every year a massive amount of images is being labeled and organized in datasets, which are applied to a broad range of applications going through image compression [12, 28], object detection [23, 32], scene understanding [16, 48], image restoration [7], and many other activities.

In the field of image restoration, researchers have been trying to use distinct sets of techniques mainly following image enhancement or restoration. Enhancement methods generally use digital image processing techniques. Many of these fail to recover information contained in a picture as they do not adopt physically-based approaches, discarding information about the three-dimensional structure of the scene. Whereas restoration techniques are generally based on some image formation model. They rely on simple digital image processing, combining them with advanced techniques and useful information about the environment. These information are commonly called priors and may cover some aspects as the distance from objects to the camera or how light propagates through the medium.

Regardless of an image visually seeming a two-dimensional world, objects composing an image scene are not always in the same visual plan. As we can see in Figure 1.1(a), object A and B seem to be in the exact same location in the scene. However, as seen in Figure 1.1(b), the two objects are far away when we change perspective. It is usually difficult to have access to this kind of information, which could explain the use of enhancement methods. Yet, it is possible to estimate missing data using computer vision techniques. Also, a single pixel in an image may not hold the same exact information of the same point in the real world. This is due to transformations that occur to the light during an image acquisition. For example, when we add fog to the scene environment, the resulting pixel from an object in the real world may appear dimmer in the imaged scene.

Within the environments images are taken, the most challenging ones for restoration purpose are those acquired in a participating medium. Light rays traveling in this type of medium tend to deviate from their original path proportionally to the amount

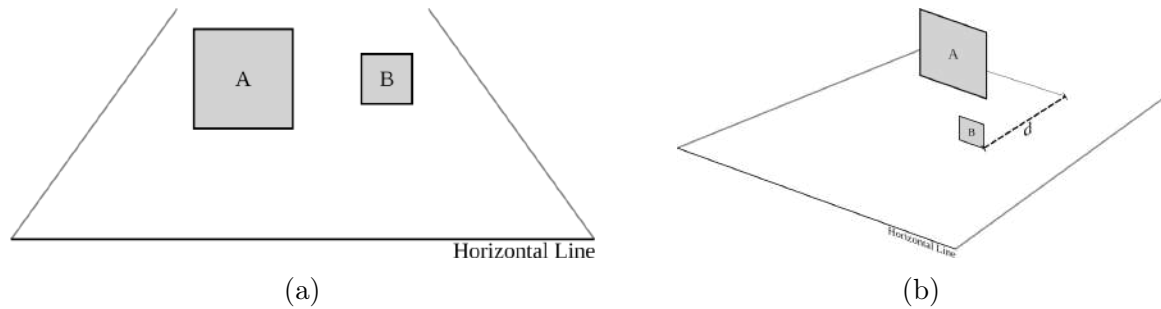


Figure 1.1: Two objects may appear to be at the same distance from the observer. However, if we have two different views from the objects in a scene, we are able to perceive the disparity between their distances.

of particles present in the environment. For instance, in places where the humidity is too high, the level of fogginess may be too heavy at certain times of the day. This fog makes the view blurry and it is difficult to visually segment objects and colors. Another example can be elucidated in underwater ambiances. Depending on the depth and level of particles present in the medium, it may be impossible to perceive the structures in some locations. These light rays carry the radiance of scene objects to the camera sensor. Radiance is the radiant flux emitted, reflected, transmitted or received by a surface. Despite the complexity of this operation, some works have been able to achieve impressive results in environments with participating medium [4, 13].

1.1 Applications

Restoring the visual quality of images acquired from underwater environments remains a great challenge for image processing and computer vision communities. Underwater images are crucial in many important applications, such as biological research, maintenance of marine vessels, and studies of submerged archaeological sites. Most of the time, structures and objects underwater cannot be removed from their location. They demand adequate handling due to their material properties. Thus, having the ability to analyze such objects without directly manipulating them is substantial to the workflow of mentioned activities.

Marine research, for example, has been helping understand how underwater environment works. Some studies focus on the forms of life present in this medium, while others analyze the impact of disasters, human or natural, on the functioning of marine ecosystems. In their work, Lu et al. [35] propose a method to classify marine organisms, including sand. In the pipeline of their approach, which they call FDCNet, they

first try to remove the haloing effects caused by water applying a descattering technique. Another interesting work follows the disaster that occurred in Japan during the 2011 earthquake. Yamakita et al. [61] gather deep-sea images to compile a dataset that helps evaluate the status of fishery in the region.

Ødegård et al. [42] use data acquired from Unmanned Underwater Vehicles (UUV) to detect archaeological artifacts of interest in wreckage sites. Structures in some of those locations need special care and, in most situations, may not have their artifacts removed from water in order to preserve their properties. UUVs could be used to address such constraint, as some have vision-based sensors used by researchers to study the underwater medium. However, Drews et al. [18] state that work is needed in this area to improve UUVs obstacle avoidance problem.

Thus, recovering underwater scenes information through image restoration would benefit a great variety of end users, including the aforementioned issue present in underwater vehicle navigation.

1.2 Problem Definition

As pointed out earlier, underwater images suffer degradation. As light travels through the environment, it is scattered and absorbed before arriving the camera sensors. Consequently, only a percentage of the scene materials properties will be acquired in the image formation process. The final result can be far from what would be expected if the scene did not undergo such effects. A sample of underwater images taken from the SUN dataset [60] can be seen in Figure 1.2. Observing these images, it is possible to see the predominance of blue and green colors, a characteristic present in most of the underwater environments. Red color contributes little, unbalancing color distribution in the imaged scene. Its wavelength weakens as light rays go deeper underwater.

Despite remarkable advances in restoring underwater images with learning methods like Convolutional Neural Networks (CNN), these methods are limited by the number of images and the quality of ground truth data used in the training. In underwater environments, the light is scattered and absorbed when traveling its way to the camera. As a consequence, objects distant from the camera appear dimmer, with low contrast and color distortion. The ground truth of an underwater image is then another image of the same scene but immersed in a non-participating media without scattering and absorption. Building datasets with high quality and a large number of images is hard or infeasible, since in most cases it is difficult to acquire images of an underwater scene in a non-participating media, *e.g.*, images taken from under the sea. Hence, the ability to

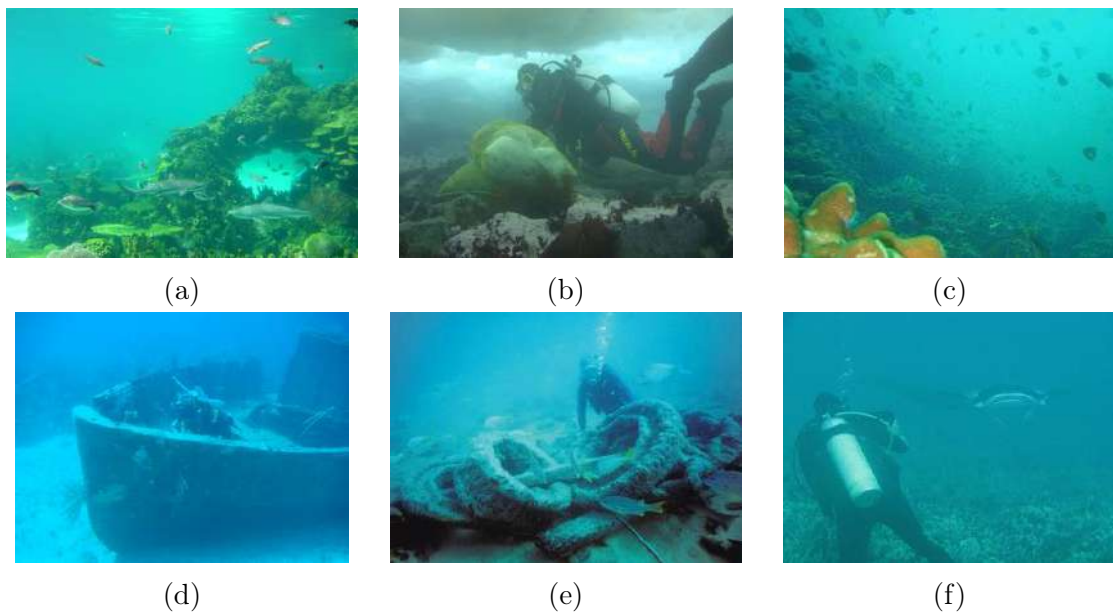


Figure 1.2: A small sample extracted from the underwater subcategory of the SUN dataset.

work with a small number of images or with simulated underwater images plays a key role in restoring the visual quality of underwater images.

1.3 Thesis Statement

The problem to be overcome in this thesis is to build a method to restore underwater images. The approach is based on [CNNs](#), in order to extract features with no labeled data. The learning process will try to improve the quality of an input image, by following well-defined underwater image quality metrics.

1.4 Contributions

In this thesis, we propose a new learning approach for restoring the visual quality of underwater images. Our method aims at obtaining the restoration model by working with simulated data and not demanding a large amount of real data. It is grounded on a set of image quality metrics that guide the optimization process toward the restored image. We also contribute with two new datasets containing images taken in controlled

underwater ambiences. The importance of these datasets is validated in our experiments, which show that our approach outperforms other methods qualitatively and quantitatively when considering the [UCIQE](#) metric, proposed by Yang and Sowmya [62].

We presented our preliminary results during the 2018 Institute of Electrical and Electronics Engineers ([IEEE](#)) International Conference on Image Processing ([ICIP](#)), which took place in Athens, Greece, from October 7 to October 10, 2018 [6]. These results include the first underwater images dataset we built during our research along with the initial restorations achieved by our proposed approach, combining qualitative and quantitative results.

1.5 Thesis Structure

This thesis is organized as follows:

Chapter 2 - Related Work presents state-of-the-art works related to the problem being addressed in this thesis. They range from works trying to enhance the visual quality of images, using some simple techniques, to works that perform image restoration, relying on more robust and advanced methods. We discuss the advantages and disadvantages of each approach, emphasizing the points we aim on solving by applying our method.

Chapter 3 - Theoretical Foundations introduces concepts that helps us understand all of the process involving our approach. We start by explaining the formation of images, how light rays interact with the medium and objects' material properties. We also introduce measurements on how to evaluate the quality of images, analyzing scenes structure and color space distribution. We conclude the chapter by contextualizing neural network foundations and the [CNN](#) aspects we include in our approach.

Chapter 4 - Methodology defines the methodology we developed to tackle the problem in consideration. We present the propagation model used to restore underwater images, along its properties and restrictions. Also, we describe a set of image quality metrics and their correlation to the human visual perspective and some quality priors. Details of the deep learning model and adaptations needed to use it in the underwater domain are depicted in the end of this chapter.

Chapter 5 - Datasets details the steps taken to build two underwater images datasets. We describe the initial plannings, which objects we used to set up the scenes and how

we changed the turbidity level of the water. Configuring each set of images resulted in different metadata. For chronological reasons, the second dataset is more robust than the first dataset. This robustness was achieved by the gathering of information about the 3D structure of the scene, which we did not worry during the settings of the first dataset.

Chapter 6 - Experiments discusses the results achieved in the experiments performed to validate the proposed approach on the two datasets we built. We specify the parameterization performed to train our CNN model, reporting the outcomes of training and testing the restoration system pipeline. Concluding the chapter, we compare our approach to enhancement and restoration techniques proposed in other works. This comparison is done by applying evaluation metrics such as computing image quality metrics not used in our system.

Chapter 7 - Conclusion presents final words about the research we have described in this thesis. We talk about the end-to-end system we propose, the drawbacks we encountered during its planning and the results we have achieved compared to other approaches. We also emphasize on the datasets we constructed, their importance to the field of image restoration and possible usages in computer vision activities. Our work is finalized by proposing future improvements to our system and potential usability of our approach.

Chapter 2

Related Work

There has been a great concern in the areas of image processing and computer vision to develop techniques that can address the issue of recovering the visual quality and information of scenes immersed on a participating medium. We can separate these techniques in two main approaches.

First, in Section 2.1, we have image enhancement, which focuses on using digital image processing algorithms to improve the visual aspects of images not taking into consideration the real structure of the environments that pictures are taken. Then, in Sections 2.2 and 2.3, we talk about image restoration. Combining digital image processing techniques along with computer vision and knowledge from other fields of study, image restoration is based on physical properties of the medium and follows well defined priors and models to achieve a result that is not only visually pleasing but also results that are plausible according to mathematical and physical models of image formation.

This chapter briefly reviews some of the work available in the literature addressing this issue, covering approaches of image enhancement using digital image processing techniques and going through restoration algorithms that use deep learning methods.

2.1 Image Processing

Early works on image enhancement relied on image processing techniques, which focused mostly on enhancing the contrast level of the scene. Some recent works developed techniques following this concept of image enhancement [8, 3, 21, 63, 7].

Contrast enhancement is a technique commonly applied to improve the visual quality of an image. Contrast is commonly used to measure the level of an image patch texture. Figure 2.1 shows images of two underwater scenes (2.1a and 2.1d) that had their contrast enhanced (2.1b and 2.1e). Although adjusting the contrast level of an image may improve its visual quality, the result is not physically plausible. Additionally, the final image could have its contents lost by the degradation of the original image. As we can

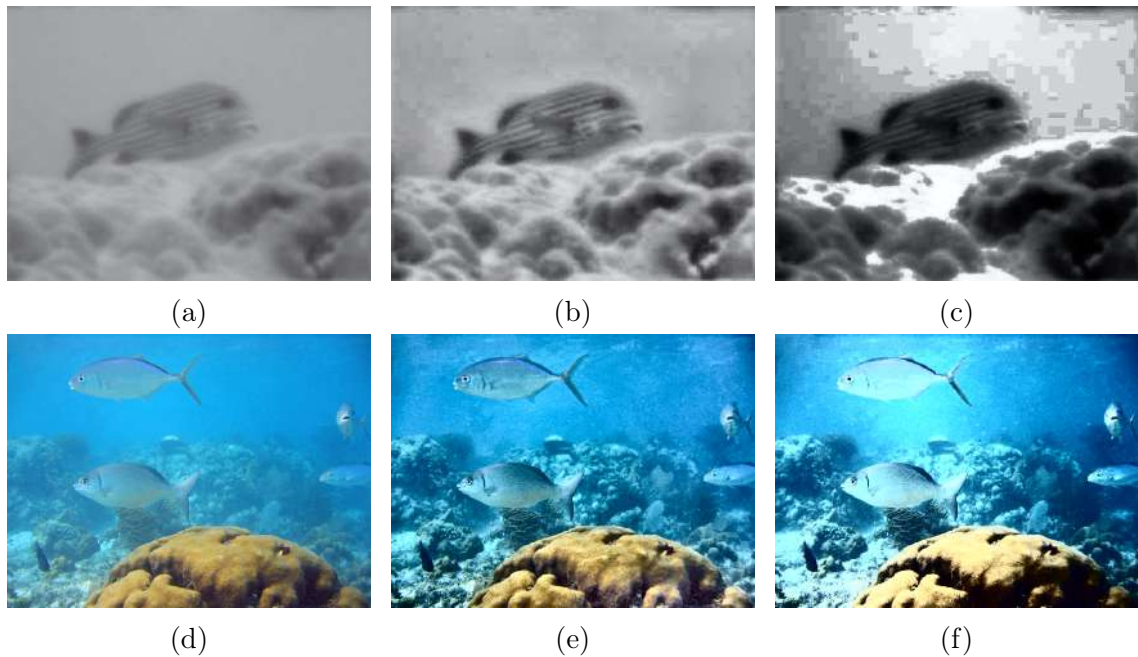


Figure 2.1: From observing the images in the middle and on the right, it is possible to understand that contrast enhancement alone cannot improve the visual quality of underwater images without losing part of the structure of objects from the original scene on the left [7].

see in Figures 2.1c and 2.1f, some regions on the images became more saturated after the increase in contrast level.

A filter bank enhancement based approach is used by Bazeille et al. [8]. This bank is composed of homomorphic filtering, wavelet denoising, anisotropic filtering, contrast adjustment, and color compensation. Orderly, these algorithms are used to reduce illumination issues, diminish the presence of noise and balance contrast, enhance edges structures and overcome prevailing colors. Figure 2.2 shows some images enhanced by Bazeille et al. [8] and Barros et al. [7] techniques.

Zheng et al. [63] use a linear combination of Contrast-Limited Adaptive Histogram Equalization (**CLAHE**) and an Unsharp Mask (**USM**). **CLAHE** reduces noise amplification resulted from traditional histogram equalization techniques, which do not take into consideration the information of local image patches. However, some portion of that noise still remains after applying **CLAHE**, making borders to lose their structure. This issue is minimized by complementing the **CLAHE** with the use of **USM**. After these two processes, they get a slightly blurred version of the original image. Regions that need contrast enhancement are selected by computing the difference between the blurred image and its original version. Then, both images are linearly concatenated to produce an enhanced image. Figure 2.3 shows an underwater image and its enhancement using this approach.

Ancuti et al. [3] derive white balanced and **CLAHE** input images from an underwater image. Using these two derived images, they estimate four weight maps, combining

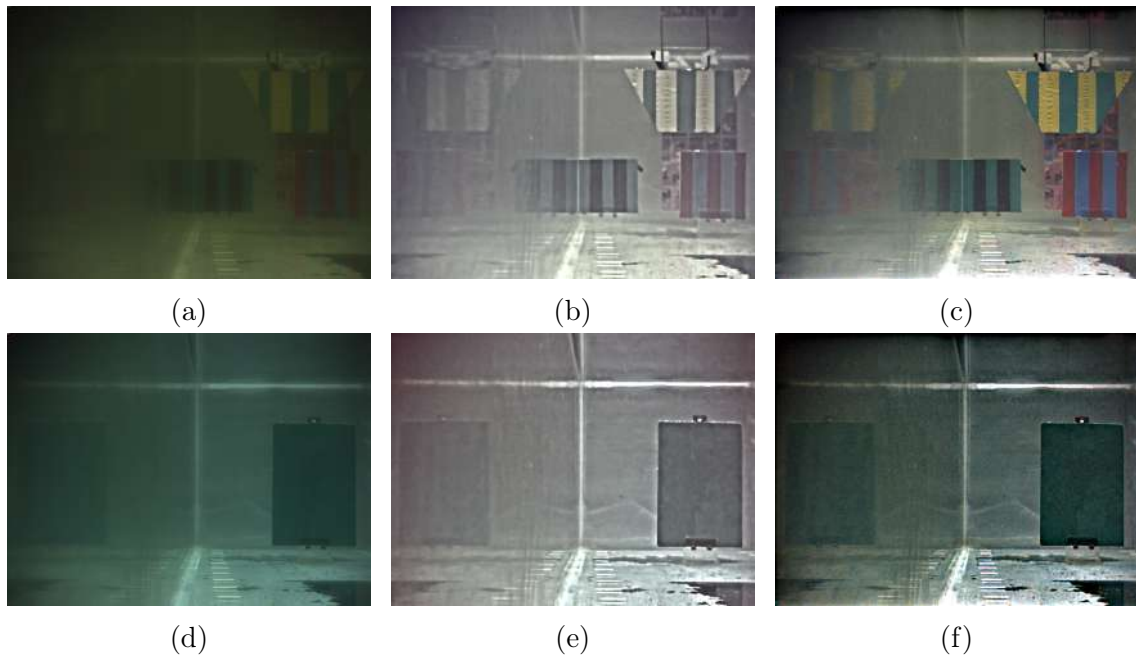


Figure 2.2: Enhancement results extracted from [7]: (a),(d) original underwater images; (b),(e) [8] enhancements; (c),(f) [7] results.

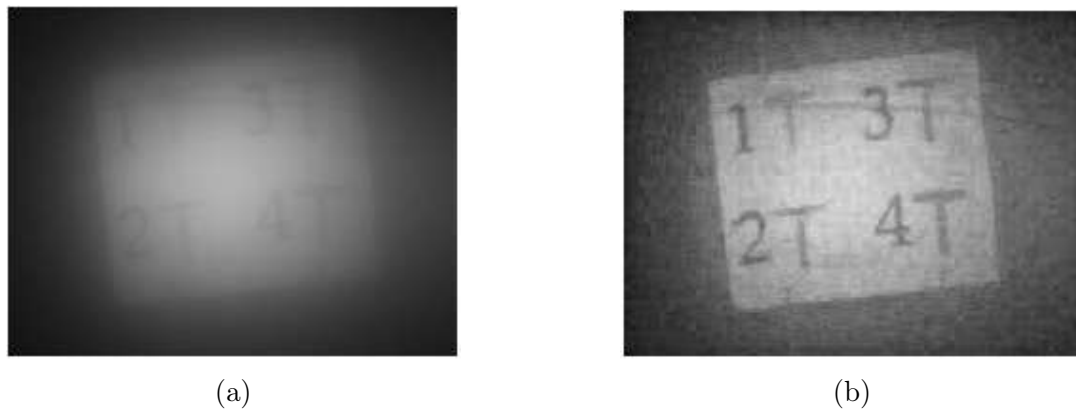


Figure 2.3: (a) Underwater image and (b) its enhancement using CLAHE and USM [63].

them by a pixelwise product and applying them in a multi scale fusion process in order to achieve the image restoration. They only apply their approach to low backlight underwater images. Figure 2.8 shows an image taken from the red sea restored using their approach. Another example can be seen in Figure 2.4a, where in the upper-left we have the original image and in the bottom its restoration.

In Figure 2.4a left, we can see the original image that was used to derive each input and the weight maps described in this method. These maps are based on (1) the application of a Laplacian filter to enhance global contrast; (2) local contrast estimation to distinguish between the different textures, which is not addressed by (1); (3) the use of a center-surround contrast concept algorithm to discriminate underwater objects saliency proposed by Achanta et al. [1]; and (4) an exposedness weight map to preserve the final image appearance.

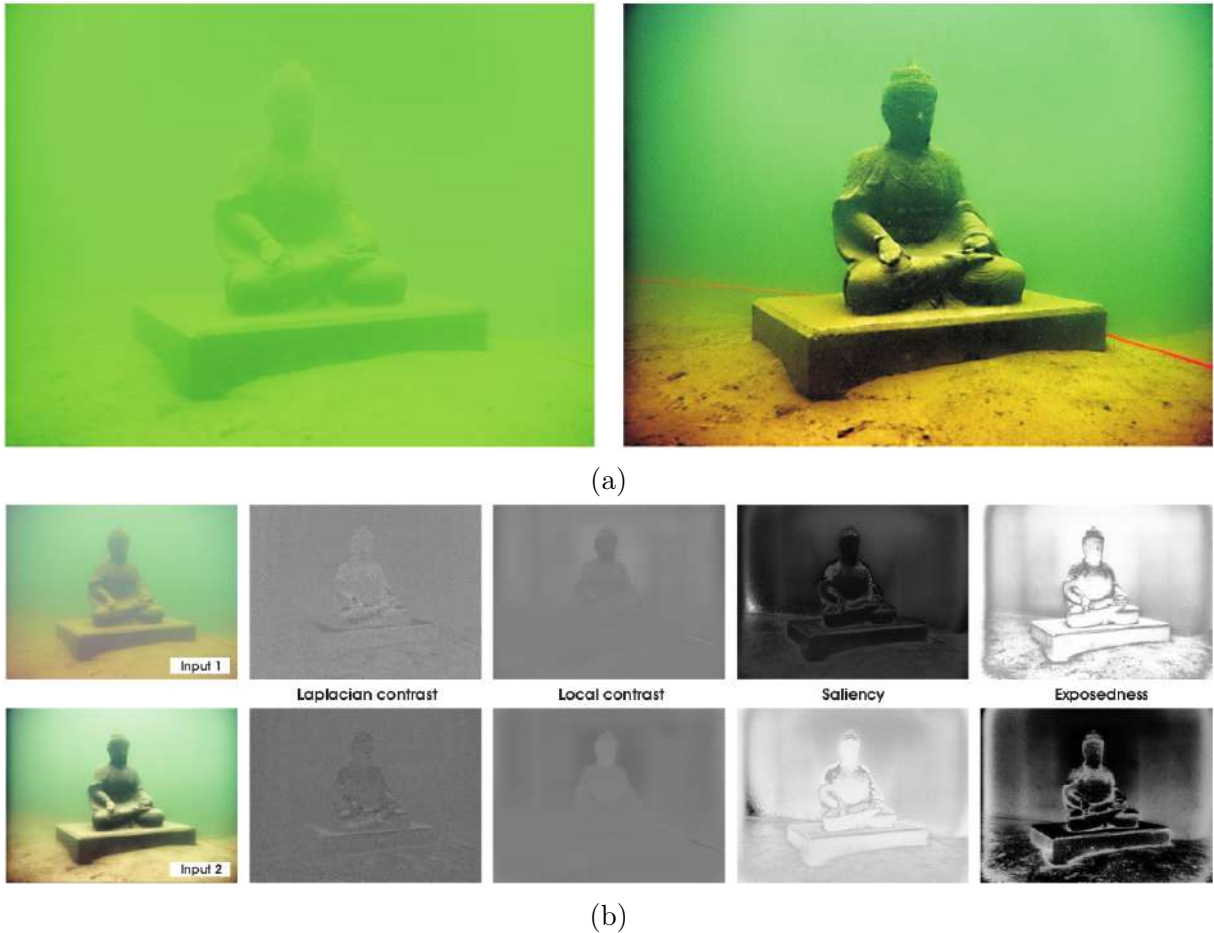


Figure 2.4: First row shows an underwater image (left) along with its enhanced version (right). This result was obtained by extracting and fusing 8 weight maps from the images in the first column (b), derived from the original underwater image [3].

While these approaches focus mainly on the visual aspects of an image, our goal is to obtain visually aesthetically good restorations that can be explained by physical models. Approaches mentioned up to this point do not take into consideration properties of the 3D scenes structure from these environments.

2.2 Image Formation Model

In the past decades, methods based on physical models have emerged as effective approaches to predict the original scene radiance [50, 57, 27, 56, 40, 25, 4, 9, 19, 44]. A representative approach is the Dark Channel Prior (DCP), proposed by He et al. [25]. The DCP is computed by taking the minimum value per channel at each pixel of an image as

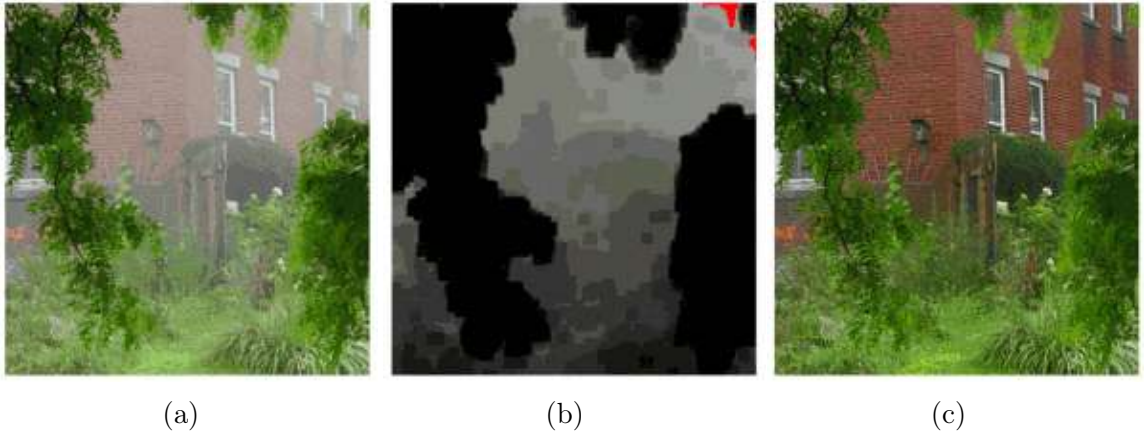


Figure 2.5: Dark channel prior based image restoration: (a) Hazy image, (b) its dark channel and (c) haze-free image [25].

$$I^{dark}(x) = \min_{y \in \Omega(x)} (\min_{c \in r, g, b} I^c(y)), \quad (2.1)$$

where $I^c(y)$ is a color channel of an image I at location y in a neighborhood $\Omega(x)$ of I centered at pixel x . Having the dark channel image, the transmission map can be estimated at each image patch and then recover the scene radiance. Figure 2.5 displays a hazy image and its dark channel. We can see in Figure 2.5c that regions in the original image corresponding to whiter areas in the dark channel received a heavier attention during the restoration process. This can be translated to restored images tending to have darker colors if the transmission map is not successfully estimated.

Mostly applied to outdoor haze-free images, the idea is that at least one of the intensity values from all color channels tends to zero. Because the assumption of the dark channel might not hold in underwater scenes, Drews et al. [19] presented the Underwater Dark Channel Prior (**UDCP**). The authors used only the blue and green channels since the red channel is drastically absorbed in underwater. The **UDCP** achieved better results than those obtained by using **DCP**.

Besides contrast, Barros et al. [7] highlight at least one more property that can be used to improve the visual quality of an image, its border integrity. As stated earlier in this document, the final result of an underwater scene acquisition is a blurred image, as the objects composing the scene have its borders diminished. That is due to the amount of light that arrives to the camera sensor. Reducing the blurriness level may increase the details of each object. Measuring how much this effect should be minimized in each image region may tell us which borders were less affected in the acquisition process. They apply an algorithm to the most degraded borders of the image in order to regain their integrity, also enhancing the image contrast. This algorithm is a multi-objective function, minimized to address the relevant features that need to be restored in underwater scenes. Restoration results can be seen in Figure 2.2.



Figure 2.6: Non-local image dehazing by clustering pixels into haze-lines: (a) hazy image and (b) recovered image [9].

Berman et al. [9] present an algorithm that assumes colors of a clean image to be approximated by a number of distinct colors, smaller than the amount of pixels in that image, forming clusters in the RGB color space. These color groups are located at different distances from the camera, which refers to distinct transmission coefficients when there is haze in the scene. Their approach clusters the pixels into haze-lines, which contain the original radiance of the image and its ambient color. Then, they estimate the transmission map, applying it to the image formation model in order to recover the original radiance. Figure 2.6 shows a hazy image recovered using Berman’s algorithm.

Tarel and Hautiere [56] use a median filtering-based approach to restore out of water images. They assume the bottom third part of an image to always have less hazing effects than the top first and second parts, which is not valid for all environments.

Peng and Cosman [44] use a physical model to estimate the image blurriness and its background light. Then, they generate three depth maps taking into consideration distinct light conditions on underwater environments: red channel, blurriness and maximum intensity prior-based depth maps. In order to construct a transmission map, they combine the three depth maps and refine the final map, finally producing a clean image, i.e., without degradation caused by the medium. Figure 2.7 shows an application of such approach. Comparing the restored image to the original hazy one, it is possible to see that, although the approach achieves a good restoration the final image still presents high level of blurriness.

Schechner and Karpel [50] emphasize that the formation of underwater images is a difficult task due to the polar visibility conditions. They state that marine animals use polarization for better vision. Thus, they elaborate a method to recover these images based on the physical model of image formation, estimating the direct transmission and the forward/backward scattering effects using two polarizing filters orientations, corresponding to extreme intensity values. Then, they obtain the depth map to restore the scene. Figure 2.8 shows an underwater scene before (upper-left) and after (upper-right) the restoration process using this technique. We can see that blurriness is still present on the restorations, as they do not focus their approach on minimizing this effect.

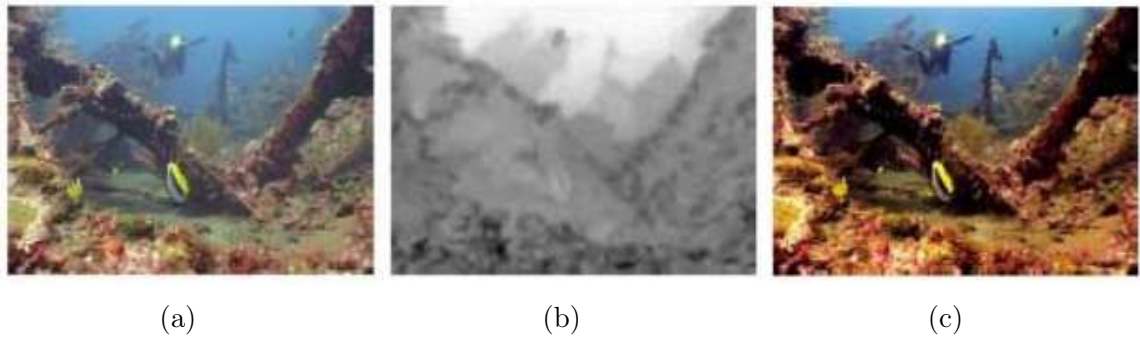


Figure 2.7: Underwater image restoration using blurriness and background light: (a) Hazy image, (b) its depth map and (c) haze-free image [44].



Figure 2.8: Red sea degraded and recovered underwater scene images extracted from [3]: original image (upper-left), Schechner and Karpel [50] restoration (upper-right) and An-cuti et al. [3] result (bottom).

Based on the radiometric underwater image formation model [36, 26], Trucco and Olmos-Antillon [57] propose a self-tuning algorithm using a simplified version of this model, initializing the parameters needed to perform the image restoration according to the global contrast of the scene. Parameters are then optimized using a quality metric in order to restore the original image. They only consider the uniform illumination in shallow waters, in which backscattering is low and does not degrade the scene at high levels.

Nascimento [41] uses a pair of cameras along with a stereo system to estimate the transmission map of an underwater scene. After estimating the attenuation coefficient

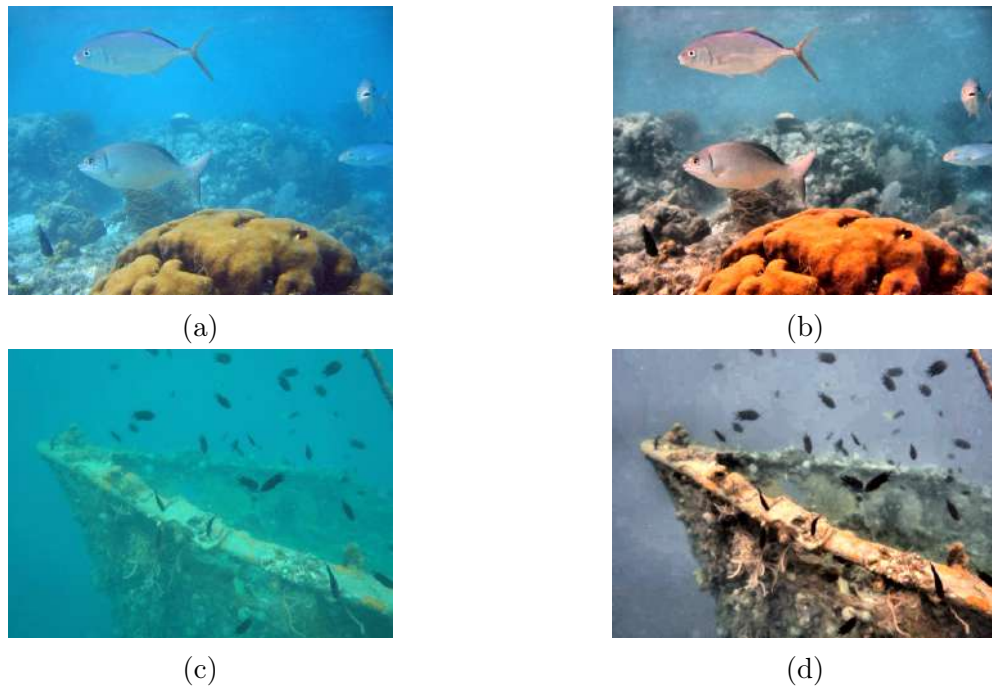


Figure 2.9: Multi-scale descattering: (a)-(c) underwater images and (b)-(d) restored images [4].

and the background light, his method restores the image by applying a physical formation model. Whereas Drews et al. [17] propose a method to automatically restore underwater images based on Structure from Motion (SFM) techniques combined with simultaneous attenuation parameters and depth map estimation.

Improving their previous work, which relied on the derivation of two input images and on the estimation of four weight maps to enhance an underwater image, Ancuti et al. [4] proposed a new image formation model-based method similar to [44]. Figure 2.9 shows some examples of underwater images restored when applying their technique.

In their newer approach, Ancuti et al. [4] estimate backscattered light aiming to improve global contrast and chrominance, computing three input images. The first image is derived using a small patch filter to better highlight regions that need contrast enhancement. Secondly, based on a larger patch filter, they compute an image to emphasize zones for regional color recovering. Then, a third image is derived using the discrete Laplacian filter to enhance fine details in the original image. Along these inputs, they also estimate three weight maps: local contrast, saturation, and saliency maps. Combining them in one by a pixelwise product. Finally, they compute a weighted sum of the inputs and the estimated maps to produce the final restored image.

These approaches are more reliable than previous image processing based techniques. This affirmation is followed by the fact that they take into consideration formation models, which can be explained by mathematical formulas and theorems.



Figure 2.10: MSCNNDehazing result. Blue and red boxes zoom into detailed from the scene that were recovered during the restoration. Images extracted from [47].

2.3 Machine Learning Based Approaches

More recently, learning techniques have shown promising results when used for recovering the visual quality of images taken from participating media [13, 47, 31, 34].

Ren et al. [47] present a multi-scale CNN to estimate the transmission map from an input image. While one network extracts more general, rough details to estimate the transmission map, the second one is used to refine the previous obtained map. Their approach is able to restore an image using the features learned. Figure 2.10 shows (a) an outdoor image and (b) the result they obtained when applying their method.

A residual deep learning approach is proposed by Liu et al. [34]. In their work, they developed an architecture that estimates a data-and-prior-aggregated transmission map. The architecture highlights the important characteristics at the same time that it tries to nullify the limitations of domain knowledge and training data information for single image dehazing. Using a modeling perspective based on an energy function, the authors refine the estimated transmission map, later applying it on underwater images to restore its visual quality.

Li et al. [31] use a Generative Adversarial Network (GAN) to generate artificial monocular underwater images from RGB-D air images. They apply three transformations. First, they estimate the attenuation coefficients. Then, backlight is approximated for each color channel. Afterwards, a vignetting effect is applied to the image. The network receives an input air image along with its depth map and a noise vector. Then, the discriminative module of the GAN classifies the produced image as real or synthetic.

After the synthetic dataset is built, they train a CNN with these images and regular RGB-D air images as ground-truth in order to restore real images. SegNet [5] is used for color restoration, using skipping layers to counterbalance the high frequency



Figure 2.11: WaterGAN restoration result. Color-shifting was also removed, as we can see in (b). Images extracted from [31].

structural decrease. As the effects included in the GAN are somewhat limited, *i.e.*, not all underwater environments have the same color tone, which they assume it is green, this approach does not seem to be a good generalizer. Thus, their major drawback is the requirement of a large dataset covering many different situations to achieve a good generalization. An example of a result from this approach can be seen in Figure 2.11.

DehazeNet, proposed by Cai et al. [13], is also a network designed for air images restoration. It restores images by extracting features as the dark channel, contrast level, color attenuation and hue disparity. These features are extracted using a CNN consisting of four modules. Their goal during the training process is to remove the haze by minimizing the error between expected and estimated transmission maps, which they state it is the key to recover a clean scene. Their ground-truth data is synthetic, where they used haze-free images to estimate their transmission map. Then, they added haze to these maps to produce the synthetic data. Impressive results are obtained when applying their approach in out of water images but do not generalize to underwater scenes. Figure 2.12 displays this network architecture.

Figure 2.13 shows an example of an air image restoration using DehazeNet. An

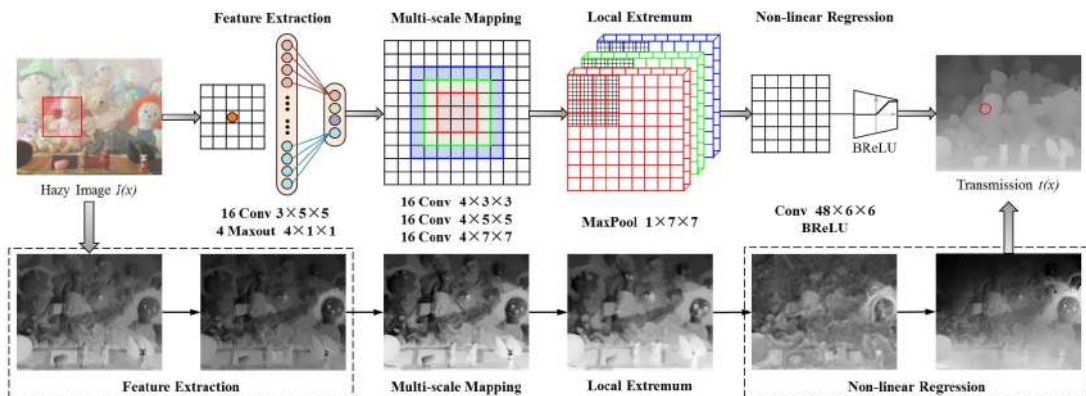


Figure 2.12: DehazeNet architecture [13].

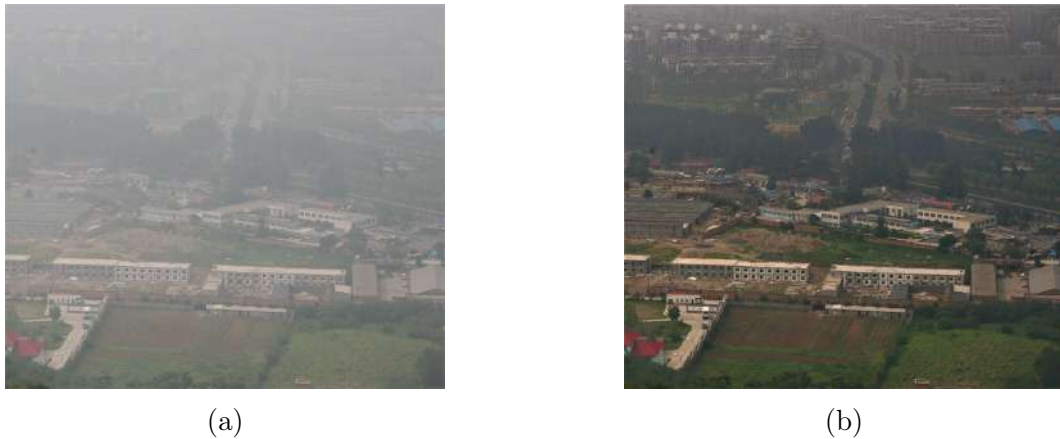


Figure 2.13: DehazeNet: successful outdoor image restoration. Images extracted from [13].



Figure 2.14: DehazeNet: underwater image restoration attempt. By comparing both images, we can see that (b) lost details present in (a) after restoration. Image from an archeological location around Turkey, taken from <https://acoustics.org/pressroom/httpdocs/155th/akal.htm>.

underwater scene restoration attempt is displayed in Figure 2.14. It can be seen that when applying the method in this domain, the result is not good as the scene blurriness is increased.

Unlike the aforementioned methods, our approach does not rely on a large dataset. Our premise is based and assessed by image quality metrics. This assumption allows us to obtain results that are directly related to the human sense of quality considering the conditions of underwater images.

There are other works addressing underwater image restoration. Sheinin and Schechner [53] propose a method to position the camera and a light source in order to minimize the scattering effect noise, generating high quality underwater images and 3D models. However, prior knowledge of the environment is needed to apply this methodology.

Chapter 3

Theoretical Foundations

This current chapter aims at contextualizing the most important concepts applied in this thesis. Initially, we discuss in Section 3.1 the process that leads to the formation of an image [55, 2]. Going through the characteristics of materials that compose a scene in the real world, and their interaction with light sources and medium properties. Later, in Section 3.2, we introduce metrics that are usually analyzed to evaluate the visual quality of an imaged scene [15]. These measures can either be based on some reference target or be statistically examined under some previous knowledge of the domain being studied. Section 3.3 concludes this chapter by briefly highlighting machine learning notations [30, 51] used in our methodology.

3.1 Image Formation Process

Three main steps are followed during the formation of an image. Understanding each one of them is important when manipulating images. Specific information will be needed to handle the most distinct environments. We detail important components of the image formation pipeline in this section.

3.1.1 The Illumination Component

Illumination can be defined as the luminous flux that bounces off from a surface. It is measured in lumane per square meter (lm/m^2) and we represent it as

$$E = \frac{dF}{dA}, \quad (3.1)$$

where E is the intensity of a point source, dF is the luminous flux and dA is the incident area at a distance r from the point source and with inclination θ relative to the normal of dA . Hence, dF can be computed as

$$dF = E_0 \frac{dA \cos \theta}{r^2}. \quad (3.2)$$

This component is very important to the process of image formation as it triggers our visual system sensors. Depending on the surface properties, the way light interacts generates different responses that affect illumination.

3.1.2 Reflection Nature and Related Models

A surface reflects light according to the material properties composing the object. The literature generally calls this the nature of reflection and distribute surfaces in three main reflectance classes, discussed in the following paragraphs.

Lambertian Reflectance: Surfaces in this class reflect light in all directions. In a diffuse manner, the whole light incident on a surface is emitted, covering a solid angle 2π radians. We can define this reflectance as

$$E_L = E_0 A \cos \theta, \quad (3.3)$$

where E_0 and θ are the incident light intensity and angle, and A is the surface area.

Specular Reflectance: Metals or mirrors, taken as few examples of this kind of surface, reflect light according to the laws of reflection, where the angle of the reflected light ray is equal to the angle of the incident light ray. The specular reflection direction can be computed as

$$\vec{E}_S = \vec{E}_0 - 2\langle \vec{E}_0, \vec{N} \rangle \vec{N}, \quad (3.4)$$

where $\langle \cdot \rangle$ is the inner product between \vec{E}_0 , the incident light ray vector, and \vec{N} , the normal to the surface.

Hybrid Reflectance: The majority of materials we find in the real world is composed of both diffuse and specular surfaces. Each class of reflectance is present at distinct amounts in the surfaces of materials. This model can be described as

$$E = \omega E_S + (1 - \omega) E_L, \quad (3.5)$$

where ω is the specular component contribution factor on the hybrid surface.

3.1.3 Point-Spread Function

Acharya and Ray [2] state that the basis of image formation can be explained by a Point-Spread Function (PSF), defined as the radiance intensity distribution in the image of an infinitely small aperture of an imaging system. It indicates how a point source of light results in a spread image in the spatial dimension, providing a measure of the image unsharpness.

If we want to image a point from a scene, the resultant image of this point will be a blurred version of it. The intensity at the center will be at its maximum and it will progressively fade away from the center, as if a Gaussian filter was passed on the image. This blurring occurs from a set of possible factors which range from not appropriate focusing of the imaging system to scatter of photons in their path to the camera sensor. Representing an image with its PSF we have

$$I(x, y) = J(x, y) \otimes P(x, y), \quad (3.6)$$

where I is the result of the input image J convoluted (\otimes) by the point spread function P at location (x, y) .

There exist multiple point-spread functions for the different environments present in the real world. From vacuum-like controlled scenes, which do not have any participating media, to gradually increasing atmosphere density and different kinds of participating media (e.g., rain, fog, sand, water). A single scene is formed by the sum of all PSFs of the points composing it.

Examples of PSFs for distinct atmosphere and underwater configurations are estimated in [37, 39].

3.2 Image Quality Metrics

During the image acquisition process, the scenes we see on pictures are a result of the light interaction with the environment and all its components. Such interactions produce some artifacts that degrade the final image at certain levels. This degradation will depend on the medium properties (e.g., atmosphere density for air images or turbidity level for underwater scenarios).

Quality assessment is commonly performed on images in order to evaluate the imaging system used in the acquisition process. A prior applied in this evaluation is the similarity of image aspects to physical attributes that the Human Visual System (HVS) finds pleasing. Examples of basic properties range from contrast sensitivity to the multichannel human vision model.

Image Quality Assessment (IQA) algorithms have been developed through various sorts of researches. The following subsections describe a few of them, which may or may not require ground-truth data, which we will describe as reference data in this thesis.

3.2.1 Full-Reference Quality Metrics

For being the first thoroughly studied, this class contains the majority of assessment algorithms. A general Full-Reference (FR) approach is to take as input a reference image and the degraded version of that image, computing an estimate of the quality of the latter relative to the reference image.

HVS-based Methods: Such approaches try to mimic the human capability of rating the quality of original and distorted images. By applying spatial filters, images are derived from input images to simulate linear responses of the primary visual cortex neurons. The final estimate of the quality is based on the difference between responses of the original image and the degraded image. This is commonly computed by taking a pointwise absolute difference operation between the two derived images, using a normalization function in the end.

Image Structure-based Methods: This class of procedures measure the quality of an image assuming that HVS draws information about the structure of natural scenes.

Thus, high quality images are those which approximate from the original image structure. One of the most popular method is the Structural Similarity Index (**SSIM**) [58], a cross-correlation-based measure that uses luminance and contrast measures to estimate quality. Another way to evaluate a degraded image structure is to compute local changes in image gradients. An example of such method is the Visual Gradient Similarity (**VGS**) index [64], in which global contrast is applied to each one of the three scales of VGS, later combining the similarity of the gradients to then perform intra and interscale pooling of the maps generated in the previous steps.

Statistical-based Methods: Following the assumptions of former methods of evaluation, approaches like the Visual Information Fidelity (**VIF**) algorithm [52] are based on the premise that **HVS** relies on statistical properties of natural environments to qualify their images. In the work of Liu and Yang [33], Peak Signal-to-Noise Ratio (**PSNR**), **SSIM**, **VIF**, and Visual Signal-to-Noise Ratio (**VSNR**), both **IQA** methods, were combined in a supervised learning technique based on decision fusion to measure the image quality.

3.2.2 Reduced-Reference Quality Metrics

In cases where there is few information about the reference image, Reduced-Reference (**RR**) algorithms are applied. An interesting approach is proposed by Gunawan and Ghanbari [22], in which they use only an edge-detected reference image. They compute a local harmonic analysis on this image in order to estimate the quality of the same scene degraded by blurring. An **RR SSIM** is also presented in the literature. Rehman and Wang [46] propose this method of extracting statistical information using a divisive normalization transform. Quality estimation is performed by a regression-by-discretization approach, which rely on the linear relationship between **FR** and **RR SSIM** algorithms.

3.2.3 No-Reference Quality Metrics

When there is no reference image for the quality evaluation, No-Reference (**NR**) **IQA** methods are applied. These techniques are generally distortion-domain specific, e.g., blurring, ringing or other types of noise. Here, we describe **NR IQA** methods that try to assess the aspects of an image which are closely related to perceived sharpness or blurriness of

scenes, some of them directly correlate to the [HVS](#) manner of qualifying images.

The majority of methods being used usually rely on the edges structure of the image but there are those which operate in the spatial domain without any priors to edges or the ones that use transform-based methods. Additionally, general-purpose non-distortion-specific methods have been developed and they are available in the literature. These approaches rely on machine learning techniques that are employed to extract natural-scene statistics in order to train an image quality classifier.

An example of a metric function that uses some of the characteristics of this class is the [UCIQE](#) metric [62]. Designed for a participating medium, this measure was elaborated by computing statistical measures on the CIE Lab color space of underwater images and correlating these features to visual subjective evaluation of humans. A set of weights was estimated using a linear regression approach in order to compose a multi-objective function which returns the quality measure of images distorted by the underwater medium.

3.3 Machine Learning Overview

Being a subfield of Artificial Intelligence ([AI](#)), which has the purpose of designing intelligent agents to fulfill activities by sensing information about the environment or making decisions relying on data of a certain domain. Machine learning gives the computer, or the device embedded with this technology, the ability to learn without explicit programming. It has three main approaches thoroughly studied in the literature:

Supervised Learning: Means the final [AI](#) model is a result of learning phases based on examples of the target domain. Typically, we use a *training set* containing n pairs $\{(x_1, y_1), \dots, (x_{n-1}, y_{n-1}), (x_n, y_n)\}$, where x_i and y_i are respectively an example and its label, and a *test set* composed by m elements $\{x_{n+1}, x_{n+2}, \dots, x_{n+m}\}$. The idea is for the model to learn from a labeled set so that it can predict unlabeled examples.

Reinforcement Learning: Involves learning what action to take in order to maximize a function. The learning agent must discover, via an exploit-and-explore approach, which decisions will yield the maximum final recompense. In reinforcement learning we can identify three main components, besides the intelligent agent. A *policy* that tells the agent what to do depending on the current state of the system. A *reward signal* defining the rewards for all the events the agent can possibly choose on its way to a goal. These

rewards can either be positive or negative, thus influencing the learner decision process. Finally, a *value function* determines if the path taken by the agent will yield a long-term maximum gain.

Unsupervised Learning: Differently from supervised and reinforcement learning, this approach does not rely on labeled data nor rewards from its environment in order to learn. It receives a *training set* $\{x_1, x_2, \dots, x_n\}$ and tries to find a pattern in the data, usually trying to reduce or increase a loss function. Examples of applications range from clustering to group the data into similar classes, to dimensionality reduction as a way of optimizing storage and post-processing by selecting the principal variables from a set of extracted features.

Each approach is applied to distinct scenarios, depending on the data and domain knowledge to be used as key to the elaboration of the [AI](#) model. There exist numerous machine learning techniques which can be used for different structures of data. Here, we are focusing on images, $2D$ arrays of n -depth size. Additionally, there are learning approaches which derive from those presented earlier, while keeping some of their context when being performed.

Self-supervised learning, for example, is a type of unsupervised learning that can be applied when there is a regression problem with limited ground-truth data. Albeit existing the possibility of this data to be used in the learning process, an objective function could replace it. This function should lead the learning process toward a desired data distribution.

Going back a little bit further, we need to contextualize Artificial Neural Networks ([ANN](#)), a bio-inspired machine learning technique, based on the biological neural system. It contains a high number of components called neurons, this quantity depends on the network complexity. [Figure 3.1](#) shows a simple example of an [ANN](#) model, though it clearly describes how this kind of system works. An input data is fed to the input layer of the network, which will pass to the hidden layer the features needed for the decision making process. Such features are a result of the application of activation functions over a weighted sum on the input data of each layer. Finally, the output layer will provide the predicted value for that data.

When we add more complexity to an [ANN](#), it normally means we add more hidden layers and neurons. This happens when we need to process images in order to do some pattern recognition or more complex digital image processing operations that would be difficult and it would take considerable time to design by hand. As the resolution of input

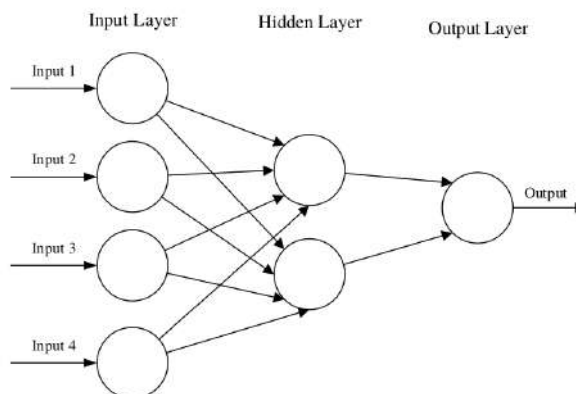


Figure 3.1: A simple ANN model containing three layers: input layer, hidden layer and output layer. Inputs from each layer are weighted summed together, being activated in every output.

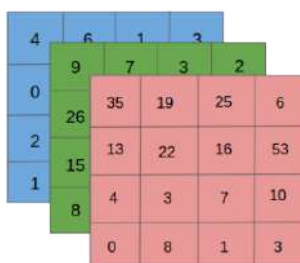


Figure 3.2: Example of a RGB input image of a CNN. This image has dimensions $4 \times 4 \times 3$, e.g., its width, height and depth are 4, 4 and 3, respectively. Image extracted from <https://medium.com/p/3bd2b1164a53>.

image increases, so does the number of neurons and layers of the network. Such models can be defined as deep neural networks, which include CNNs.

3.3.1 Convolutional Neural Networks Concepts

Similar to the application of filters in image processing, a CNN can be described as a bank of filters that are applied to the input of each layer and results in a desired result computed using the features from the output of each layer. May that result be a class for classification tasks, a bounding box for object detection or a pixel-wise regression that will result in an image encoding or decoding process, or in the transformation of pixels for restoration purposes. The architecture of a CNN is analogous to that of an ANN, being inspired in the human visual cortex. Neurons activates to stimuli in the receptive field, which is restricted to a region of the visual field.

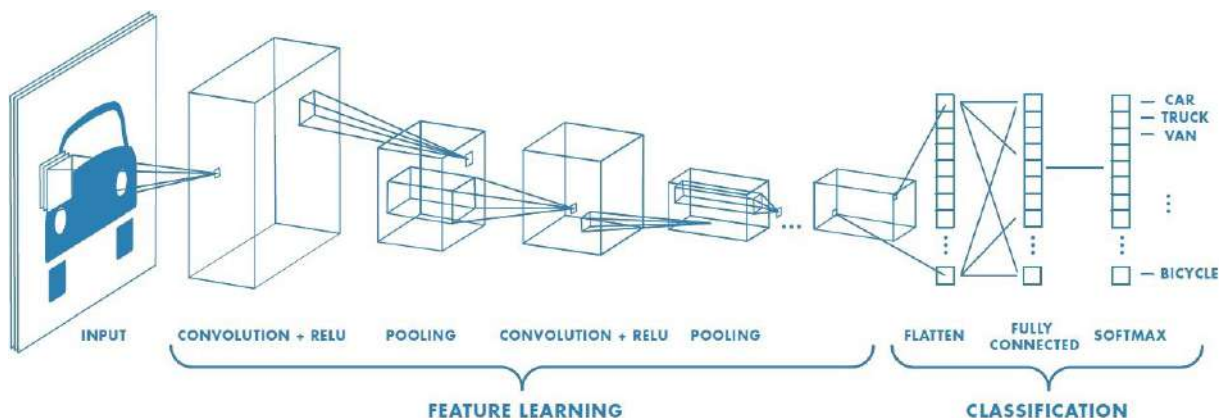


Figure 3.3: **CNN** overview. A simplified model would have convolutional layers followed by pooling layers and activation functions. Additionally, fully-connected layers may be introduced for classification purposes. Image extracted from <https://medium.com/p/3bd2b1164a53>.

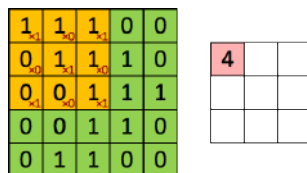


Figure 3.4: Example of a filter of size 3×3 being convoluted through a patch of an image. Image extracted from <https://medium.com/p/3bd2b1164a53>.

A **CNN** architecture comprises several modules and concepts illustrated in Figure 3.3, some of them are described and exemplified as follows.

Input Data: This is generally a 3-channel image, from any color space, depending on the application. The purpose of a **CNN** is to reduce the dimensions of this image without losing features that describe important information about the desired task. Figure 3.2 shows an example of a RGB image, with 4 pixels each dimension.

Convolution Layer (CONV): Among the different types of layers in a **CNN**, a core component is the convolutional layer, responsible for doing the heavy computational work. It consists of a set of learnable filters, generally called weights, which are convolved through the input image producing a feature map. This convolution process is similar to a sliding window, where the filter will slide through the image path horizontally and vertically until it reaches the end of the patch. Each element on the feature map is a response of that filter at each local region on the input data, these responses are highlighted by an activation function (e.g., activation responses of a filter to detect corners). Figure 3.4 shows an example of a convolution being performed. A filter of size 3×3 is used, each weight on the filter is multiplied by its corresponding location in the image patch. This operation is illustrated in the yellow block. Multiplication results are then added together and then fed to an activation function.

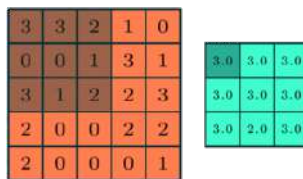


Figure 3.5: Example of a pooling filter of size 3×3 being convoluted through a patch of an image. Image extracted from <https://medium.com/p/3bd2b1164a53>.

Activation Function: These functions are generally used after each weighted sum is computed in **CONV** layers. Most of the time we get values we do not really know if they are useful for the learning process. That is the reason we apply some functions to tell if a neuron response should be fired, not fired, or to what extent the following layers should consider that response. Examples of such functions are Sigmoid, outputting probability values between 0 and 1; Softmax, making each component of a feature vector to add up to 1, mostly used in classification models; and Rectified Linear Unit (**RELU**), used in the majority of **CNN** models, this activation function puts a minimum clipping threshold of 0, where all feature values below this bound are set to 0. There exist variations of these, along other types of activation functions.

Pooling Layer (POOL): Mostly used to reduce the quantity of parameters to learn as the network goes deeper. This minimizes computational work and reduces the spatial size of features. The goal here is to reduce dimensionality by keeping dominant features, which are relevant to the learning process. *Max-Pooling* is the most used pooling operation, as it has shown to work better in actual cases. Figure 3.5 illustrates a *Max-Pooling* being performed with a 3×3 window size over an image patch. The brown block is the pooling window and its output is depicted in the dark green pixel in the 3×3 image on the right.

Fully-Connected Layer (FC): It is simply the case where each neuron on this layer is connected to all activations from the previous layer, generating an array of n outputs, which are passed through an activation function, as Softmax. Mostly used in classifier models, each output refers to a single class.

Backpropagation Algorithm (BP): This is a concept generally applied during the training process of **AI** techniques, after all operations are computed in a forward pass of the network. Depending on the error yielded by a defined loss function, the weights of each layer are updated so that future forward passes result in a lower error value.

Loss Function (\mathcal{L}): In most learning models, error is computed as the difference between the estimated output and the expected output, which could be a label for supervised learning or a domain-data-driven prior value for unsupervised approaches. This function

has great impact on the model performance as it is directly correlated to the weight updating process during [BP](#).

Chapter 4

Methodology

In this chapter, we focus on describing our solution to tackle the underwater image restoration issue. Section 4.1 describes how an image with haze or some other degrading effect can be restored by using a simplified formation model. Sections 4.2 and 4.3 explain our methodology, comprising a two-phase learning.

In the first phase of our methodology, we perform a supervised training by fine-tuning the DehazeNet [13]. This network was developed to restore images acquired from scenes presenting a high level of haze, an atmospheric phenomenon where dry and liquid particles affect the sky clarity. Afterwards, the input image is restored according to a formation model. In the second phase, we minimize a loss function composed of quality metrics to finally perform image restoration. The assortment of these metrics was realized by evaluating studies concerning underwater images and their properties. The following section describes the general idea for the formation of an underwater image. Figure 4.1 shows a simplified system of light interactions in this medium. Figure 4.2 illustrates the process we have adopted in our methodology.

4.1 Image Formation Model

In the underwater environment, the image is a combination of the light coming directly from the objects composing the scene and light that was redirected towards the camera. In this thesis, we use a commonly referenced image formation model [20, 25, 19], expressed as

$$I = Je^{-\beta d} + B(1 - e^{-\beta d}), \quad (4.1)$$

where I is the observed light intensity, J the scene radiance, B the background light, which is the light coming from other scenes not in field of view of the camera, and $t = e^{-\beta d}$ is the transmission map. This map t gives the amount of light not attenuated, due to scattering

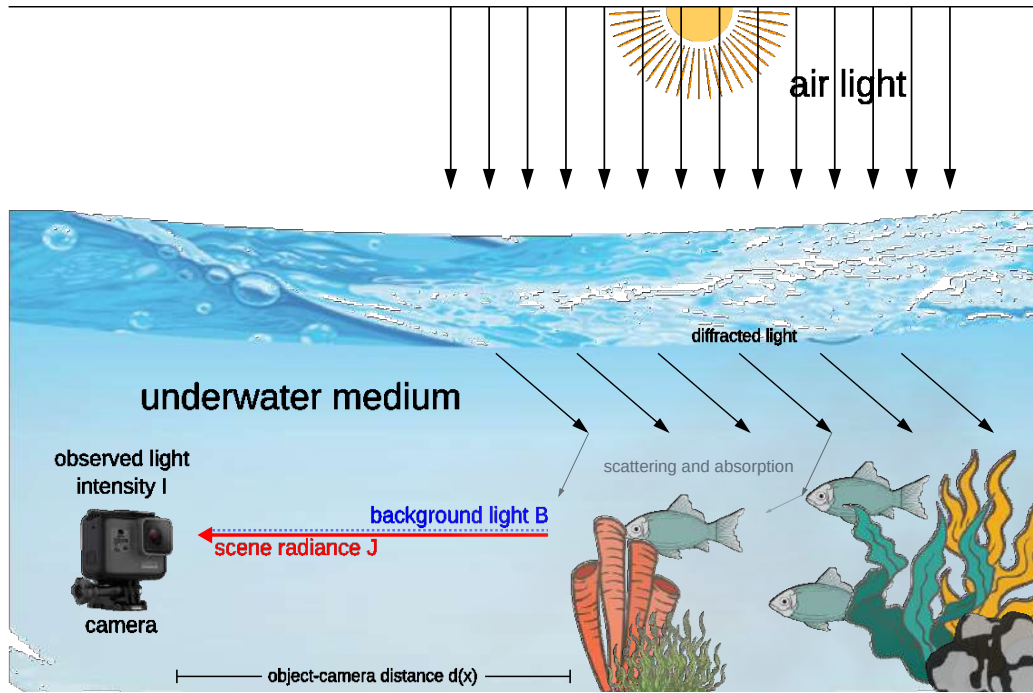


Figure 4.1: Air light, besides being diffracted, is scattered and absorbed by objects and tiny particles present in the underwater medium. The resulting radiance that arrives to the camera sensor is a combination of a percentage of light reflected from the scene and light coming from other sources, the background light.

or absorption, on a given point x at a distance $d(x)$. The parameter β represents the medium attenuation coefficient. As we need to approximate J , we reorganize Equation 4.1 as

$$Jt = I - B(1 - t). \quad (4.2)$$

We need to isolate J and we know that t is a 2D-matrix containing the amount of light not attenuated for each pixel of the scene. Then, to continue the process of approximating

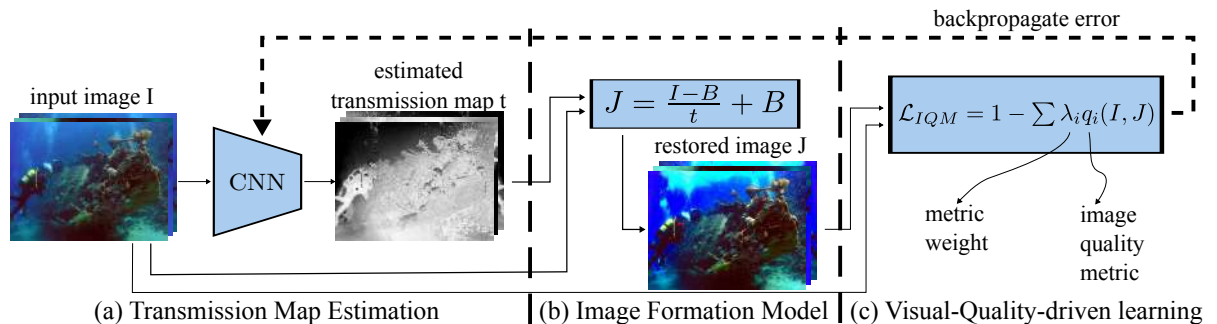


Figure 4.2: Diagram of our two-stage learning. First, we fine-tune the CNN using ground truth transmission maps, applying the mean squared error loss in the training process. Second, we take the model and adapt it by including the image restoration process (b). Finally, we perform new training in the network minimizing a loss function based on image quality metrics.

J , we multiply both terms of the equation by the inverse transmission map

$$Jtt^{-1} = (I - B(1 - t))t^{-1}. \quad (4.3)$$

After solving Equation 4.3, we have

$$J = It^{-1} - Bt^{-1} + B. \quad (4.4)$$

Finally, J can be estimated by reformulating Equation 4.4 as

$$J = (I - B)t^{-1} + B. \quad (4.5)$$

Thereby, as we already have I from the image acquired, we only need to estimate t and B to restore an image in the underwater environment. The transmission map t can be estimated using the fine-tuned DehazeNet. Following the prior of [19], we can roughly estimate the background light as

$$B = \max_{y \in \{x | t(x) \leq t_0\}} I(y), \quad (4.6)$$

where B is the pixel in a degraded image I whose transmission map value is the highest, limited by a constant t_0 . The value t_0 is chosen as the 0.1% highest pixel. If there is more than one pixel satisfying this condition, we compute their mean.

Background light is inversely proportional to the transmission map. Thus, we can extract B from the region we are unable to see objects in an underwater image, as depth is too high. This can be visualized in Figure 4.3. The darker the region in the transmission map, the most accurate the estimation of the background light in the underwater image.

4.2 Transmission Map Estimation

In order to recover the scene information, the CNN model follows an accepted physical image formation model, relying on the approximation of the transmission map. To adjust the network to our purpose, we proposed to perform a supervised approach following the training guidelines of [13]. It consists in using underwater images as input to the network and comparing its estimated transmission map to ground truth maps. The loss function used in this stage is the Mean Squared Error (MSE), defined as

$$\mathcal{L}_t(I, J) = \frac{1}{n} \sum_{x=1}^n (t_I(x) - t_J(x))^2, \quad (4.7)$$



Figure 4.3: (a) an example of an underwater image; (b) the transmission map of the image on the left.

where n is the number of pixels in the image, t_J the estimated transmission map and t_I the ground truth. Thus, the network will have its weights updated towards approximating the expected transmission map of each scene used in the training stage. Figure 4.3 shows an example of an underwater image along with its transmission map. By analyzing the map, we can see that regions closer to white are closer to the camera the picture was taken, while darker regions are far from the camera.

4.3 Visual-Quality-Driven Learning

Based on the work of Barros et al. [7], to overcome the absence of ground truth data for underwater scenes, we present an approach that assesses the result by computing a set of Image Quality Metrics (IQM). The IQM set \mathcal{X} yields a multi-objective function that measures the enhancement of four features in the restored image in comparison to the input image. The multi-objective function is given by

$$IQM(I, J) = \sum_{X \in \mathcal{X}} \lambda_X q_X(I, J), \quad (4.8)$$

where λ_X is the weight for a feature gain q_X .

We choose four metrics that are well correlated to the human visual perception to compose our IQM set: contrast level, acutance, border integrity, and gray world prior.

Contrast Level: Underwater images tend to have low contrast as the amount of water between objects and the camera increases [40]. We compute the contrast gain of a restored image J over the degraded image I as

$$q_C(I, J) = \frac{1}{n} \sum_{x=1}^n (C(J, x)^2 - C(I, x)^2), \quad (4.9)$$

where $C(image, x)$ is the contrast level of a pixel x in the grayscale version of an image, computed as

$$C(image, x)^2 = \frac{\sum_{x=1}^n \sum_{c=r,g,b} (image_c(x) - \frac{1}{n} \sum_{y=1}^n image_c(y))^2}{(\sum_{x=1}^n \sum_{c=r,g,b} image_c(x))^2}. \quad (4.10)$$

Acutance: The restoration process should also enhance the acutance metric, which measures the human perception of sharpness [29]. The restoration gain for acutance is given by

$$q_A(I, J) = \frac{1}{n} \sum_{x=1}^n G(J, x) - \frac{1}{n} \sum_{x=1}^n G(I, x), \quad (4.11)$$

where each term of the subtraction in the equation is the acutance of the degraded and the original image, respectively. $G(K, x)$ is the gradient magnitude of an image K at pixel x . We use the Sobel operator [54] to compute this magnitude.

Border Integrity: It measures the visibility of the borders after the restoration. This measurement allows us to check how much the border increased in regions that were likely to have borders, avoiding random noise to appear in the restored image. It is calculated by

$$q_{BI}(I, J) = \frac{\sum_{x=1}^n (E(J, x) \times E_d(I, x))}{\sum_{x=1}^n E_d(I, x)}, \quad (4.12)$$

where E is an edge detector, here we use the Canny edge detection [14], and E_d is a morphological operation which dilates the borders of an image by 5 pixels. \times represents an element-wise multiplication.

Gray world prior: The fourth feature is the gray world prior [11], a hypothesis that, under natural circumstances, the mean color of an image tends to gray. Therefore, we evaluate how distant from the gray world our restored image is by computing

$$q_G(J) = (I_{max} - I_{min}) - \frac{2}{n} \sum_{x=1}^n (I(x) - I_m)^2, \quad (4.13)$$

where I_{max} and I_{min} are the maximum and minimum intensity a pixel can have and I_m is the average of these two values. The first term of q_G makes the metric higher as the distance from the gray world is smaller.

After computing the gain in each quality metric, we minimize the subsequent IQM loss:

$$\mathcal{L}_{IQM} = 1 - IQM(I, J). \quad (4.14)$$

Network weights are updated by propagating the IQM error backwards. This step is done by computing the gradient of the loss function.

$$\frac{\partial \mathcal{L}}{\partial \omega} = -2 \frac{\partial J}{\partial t} \frac{\partial t}{\partial \omega} (\lambda_C \frac{\partial C}{\partial J}(C)) + \lambda_E \frac{\partial E_I}{\partial J}(E_J). \quad (4.15)$$

Note that this step does not require labeled data. Instead of using ground truth data, our method uses the quality metrics to guide the optimization process for refining the transmission map. As a result, the final image presents a physically-plausible restoration of the input with better quality than the results achieved by other approaches.

Chapter 5

Datasets

One of the disadvantages in using a convolutional neural network is that it requires a large amount of data to be trained on. This becomes an issue in the underwater domain. For instance, if we wanted to perform a supervised training, where the network model should learn how to approximate an underwater image from its out of water version. It would be difficult for this model to learn without enough data. This task is difficult as it is much complicated to have an image of the same scene taken from under and out of water.

However, we pointed out that the main idea of our methodology relies on quality metrics guiding the restoration learning process. Thus, we do not need a large set of images containing some kind of ground truth information. In order to validate that our method successfully restores underwater images, we have built four datasets, grouped into two larger datasets, including synthetic and real scenes under controlled turbidity levels to simulate a few underwater environments. These sets of images were created through two phases of experiments, with increasing level of complexity in the process. In summary:

Section 5.1 - UVision18 Dataset details the steps taken in the building process of the first dataset. We see this dataset as an initial attempt to produce an underwater image restoration dataset.

Section 5.2 - UVision19 Dataset gives the decisions made regarding the set up of this dataset, taking into consideration the process of building the first underwater dataset. It also describes the techniques applied in the assembling process.

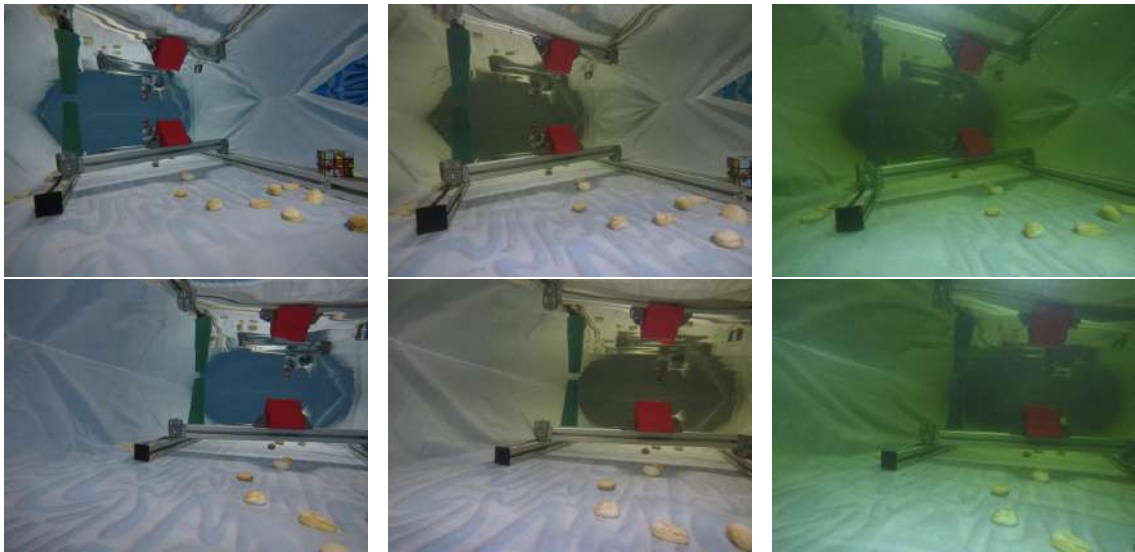


Figure 5.1: UVision18 RGB image samples. We can see water mirroring due to the position the camera was placed to take the pictures.

5.1 UVision18 Dataset

This initial dataset was built to see if the methodology was indeed suitable for solving the problem of underwater image restoration. Using a water tank of dimensions $126\text{cm} \times 189\text{cm} \times 42\text{cm}$, 665 liters of water and 3 solution configurations to simulate distinct levels of turbidity, a total of 695 images were taken with a GoPRO HERO5 Black camera (from now on we are going to refer to the camera as GoPRO). The turbidity levels were controlled using different quantities of green-tea sachets, respectively: 0g (clean water), and 80g and 160g. Images were acquired positioning the camera in different points inside the water tank. Figure 5.1 shows a sample of RGB images from *UVision18* dataset.

Listing 5.1: PBRT Li method changed code to compute the transmission map of a scene. This method can be found in `integrators/directlighting.cpp` on the tool source code repository.

```

1 Spectrum DirectLightingIntegrator::Li(const RayDifferential &ray,
2                                     const Scene &scene, Sampler &sampler,
3                                     MemoryArena &arena, int depth) const {
4     ProfilePhase p(Prof::SamplerIntegratorLi);
5     Spectrum L(0.f);
6     SurfaceInteraction isect;
7     if (!scene.Intersect(ray, &isect)) {
8         return L;
9     }
10    L += 1 - Distance(ray.o, isect.p) / 13;
11    return L;
12 }

```

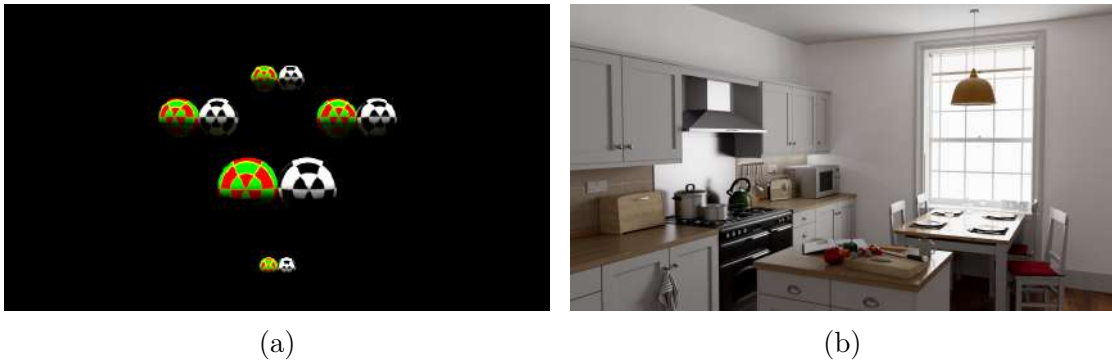


Figure 5.2: UVision18 3D scenes: (a) shows a set of spheres in different distances from the camera; (b) displays a kitchen.

To increase the amount of data, we created a set of synthetic underwater images using the [PBRT](#) [45]. In order to generate the transmission map from each scene, we changed the method that computes the amount of light that arrives in the image plane. This modification is depicted in Listing 5.1. Two base 3D scenes were rendered fixing the camera in different positions. Figure 5.2 displays the scenes used in this process.

A total of 642 images was rendered, along with each respective transmission map. We set the absorption and scattering coefficients according to Mobley [38]. These coefficients are displayed in Table 5.1, where each column respectively refers to light wavelength, spectral absorption and molecular scattering coefficients.

Listing 5.2 shows an example of a [PBRT](#) input file, describing a new water medium. Properties *sigma_a*, *sigma_s* and *scale* define, in this order, absorption, scattering and scaling of coefficients depending on the distance unit, which should be in millimeters. A sample of these synthetic images is displayed in Figure 5.3. Comparing the kitchen images to Figure 5.2b, it is possible to visualize that we removed the walls and the ceiling from the 3D scene structure.

Listing 5.2: Example of a [PBRT](#) input file describing a water medium. Variables *sigma_a*, *sigma_s* define absorption and scattering coefficients, whereas *scale* sets the scaling of these coefficients to millimeters.

```

1 MakeNamedMedium "water" "string type" [ "homogeneous" ]
2     "spectrum sigma_s" [ 600 .0014 ] "spectrum sigma_a" [ 600 .244 ]
3     "float scale" [ 0.001 ]
4
5 MediumInterface "" "water"

```

$\lambda(nm)$	$a(m^{-1})$	$b(m^{-1})$	$\lambda(nm)$	$a(m^{-1})$	$b(m^{-1})$	$\lambda(nm)$	$a(m^{-1})$	$b(m^{-1})$	$\lambda(nm)$	$a(m^{-1})$	$b(m^{-1})$
200	3.0700	0.1510	360	0.0379	0.0120	520	0.0477	0.0024	680	0.4500	0.0007
210	1.9900	0.1190	370	0.0300	0.0106	530	0.0507	0.0022	690	0.5000	0.0007
220	1.3100	0.0995	380	0.0220	0.0094	540	0.0558	0.0021	700	0.6500	0.0007
230	0.9270	0.0820	390	0.0191	0.0084	550	0.0638	0.0019	710	0.8390	0.0007
240	0.7200	0.0685	400	0.0171	0.0076	560	0.0708	0.0018	720	1.1690	0.0006
250	0.5590	0.0575	410	0.0162	0.0068	570	0.0799	0.0017	730	1.7990	0.0006
260	0.4570	0.0485	420	0.0153	0.0061	580	0.1080	0.0016	740	2.3800	0.0006
270	0.3730	0.0415	430	0.0144	0.0055	590	0.1570	0.0015	750	2.4700	0.0005
280	0.2880	0.0353	440	0.0145	0.0049	600	0.2440	0.0014	760	2.5500	0.0005
290	0.2150	0.0305	450	0.0145	0.0045	610	0.2890	0.0013	770	2.5100	0.0005
300	0.1410	0.0262	460	0.0156	0.0041	620	0.3090	0.0012	780	2.3600	0.0004
310	0.1050	0.0229	470	0.0156	0.0037	630	0.3190	0.0011	790	2.1600	0.0004
320	0.0844	0.0200	480	0.0176	0.0034	640	0.3290	0.0010	800	2.0700	0.0004
330	0.0678	0.0175	490	0.0196	0.0031	650	0.3490	0.0010			
340	0.0561	0.0153	500	0.0257	0.0029	660	0.4000	0.0008			
350	0.0463	0.0134	510	0.0357	0.0026	670	0.4300	0.0008			

Table 5.1: Spectral absorption and molecular scattering coefficients per light wavelength in a pure water medium, provided by Mobley [38].

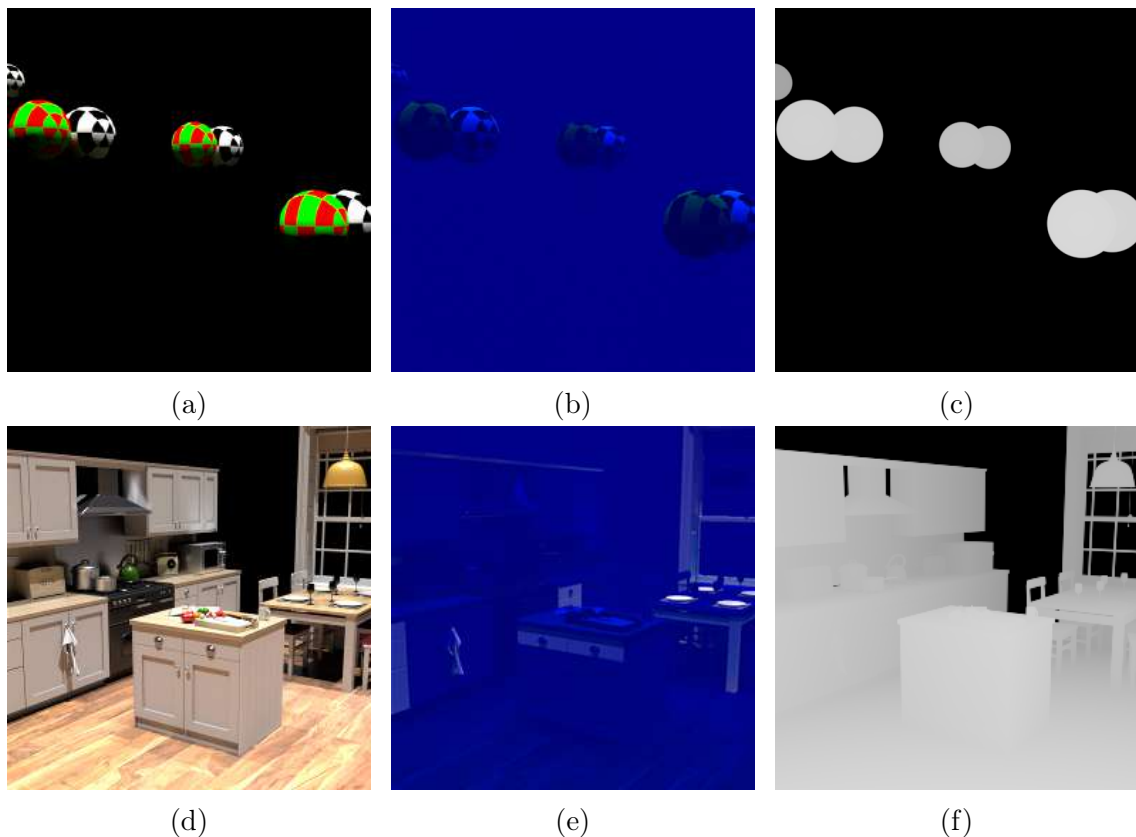


Figure 5.3: *UVision18* synthetic sample rendered with **PBRT**: Figures *a* and *d* are images from a different perspective of the 3D scenes, Figures *b* and *e* shows these images as they were underwater, Figures *c* and *f* are the transmission maps of *a* and *d*.

5.2 UVision19 Dataset

This new dataset was designed to fill the gaps the previous dataset presented, regarding restoration purposes for underwater images. We are now able to get the depth information for each planned scene. Consequently, an extensive set of turbidity levels simulating various types of water medium can be produced using the scenes available. Applying an image formation model, we could generate a considerable amount of synthetic underwater images for the scenes we elaborated in this dataset.

Setting Up. We used the same water tank described in Section 5.1. Additionally, we included an old wooden door to fix the objects that compose the scene, a 30cm-height support for the RGB camera and two 2cm-sided calibration marks.

As we filled the tank with water, the door started to float, as it has density less than water. Thus, we decided to cover the door surface with gravel, which helped keep the wooden material in the bottom of the water tank. The lateral surface was covered with jute fabric in order to minimize possible illumination and reflection issues that are not desirable for the experiments led in this thesis.

We wanted to acquire the images under controlled settings, so we customized a source light composed by two 32W fluorescent lamps fixed inside a metal box with its output blinded by a twice-folded white non-woven fabric. This allowed us to obtain a near diffuse light source. Figure 5.4 illustrates the settings used to acquire the underwater images from a set of real scenes.

Real Scenes. Figure 5.5 shows scenes we used to build the UVision19 dataset. Three red cubes are present in all the scenes. They could be used to analyze the dimming of color channels as depth and turbidity levels increase.

The idea behind leaving two calibration marks in every scene is that they may be used to calibrate these images with the images taken with the depth sensor. Therefore, if we needed to know the correspondence of a pixel on an object in the world coordinate system, we could do this by transforming it from the camera coordinate system to the world coordinate system.

A group of nine objects were placed in different positions every other scene. The arrangement aims to have an object at distinct distances from the camera sensor. This helps evaluate restoration at variate depth levels.

RGB-D Images. A Kinect One was used to acquire images from the scenes along with their depth information. The camera was positioned on a box behind the RGB camera support. Algorithm 1 describes the steps followed to take the RGB-D images. Figure 5.6 shows a sample of color images (top-row) and depth images (bottom-row) acquired using the Kinect One.

Algorithm 1: RGB-D images acquisition using a Kinect One.

```

1 Place Kinect One in the platform behind RGB camera support;
2 while not all scenes had their RGB-D images acquired do
3   Setup new scene;
4   Run save_image.py script;
5   Press "s" to save both RGB and depth images;
6   Repeat step 5 10 times;

```

The `save_image.py` script uses `iaikinect2` [59], a collection of tools and libraries for a Robot Operating System (ROS) Interface to communicate with the Kinect One. This script subscribes to `/kinect2/qhd/image_color_rect` and `/kinect2/qhd/image_depth_rect` topics available in the Kinect One, respectively providing color and depth images.

The depth image is a matrix where each cell contains the depth in millimeters of a specific point in the world. This point maps to a pixel on an RGB image, also acquired using the Kinect One. During acquisition, the sensor may not map all world points to the depth image. We try to regain this lost information by taking 10 RGB-D images of each scene. Later, we compute the average image for the RGB images. For the depth images,



Figure 5.4: Setting up of the UVision19 dataset. It is possible to see the water tank, the calibration marks and the supports for the camera, the light source and the objects.

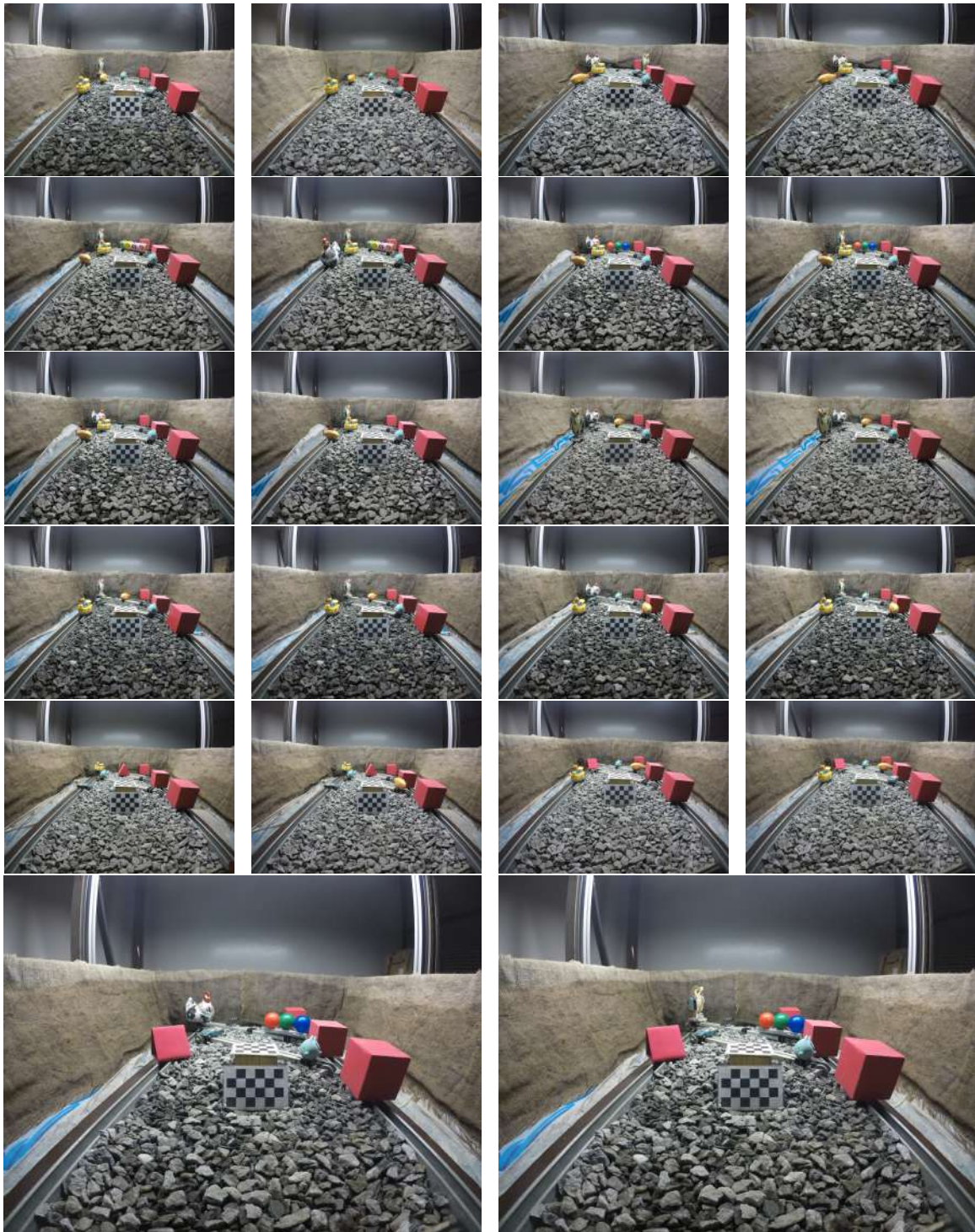


Figure 5.5: Scenes available in the UVision19 dataset. Each scene is composed by two chessboard calibration marks, three red cubes and a few sorting of objects.

we normalize them by computing

$$D_i^{final}(x, y) = \begin{cases} \max_{1 \leq j \leq 10} D_i^j(x, y), & \text{if } i \leq 4,499 \\ 4,499 & , \text{otherwise} \end{cases} \quad (5.1)$$

where D_i^{final} is the final depth image of scene i . We apply a pixel-wise max function in all

10 images of the same scene. This value is then limited by a threshold of 4499 millimeters, which indicates the maximum approximate distance the sensor can reach [43].

RGB Images. Color images were captured using a GoPRO. During this process, the water tank was filled with water at each scene imaging. Also, the room light was turned off to make the environment be illuminated only by our diffuse source light.

Green tea was also used in this phase. However, *UVision18* used green tea sachets, while *UVision19* used 80g dry leaves green tea bundles. We processed the tea in a blender and then sifted the powder to get rid of large solid parts that could create different effects in the medium. Floating particles from the blended tea could affect the restoration process of our methodology. Figure 5.7 shows an example of an image from a scene where we did not sift these particles, which clearly difficult the process of image restoration. *UVision18* green tea was not as concentrated as *UVision19* tea. Thus, for the setup of the second dataset, we needed to use lesser amounts of green tea to control the water turbidity levels.

We decided to apply three levels of turbidity in the clean water using 15g, 20g and 25g of sifted green tea, respectively. Algorithm 2 describes the pipeline of RGB images acquisition. We took pictures using the 10-shot burst functionality in the 4 modes available in the GoPRO camera: linear, medium, narrow and wide. Figure 5.8 shows a sample illustrating the distinct levels of turbidity present in the dataset, taken in the medium mode of the camera.

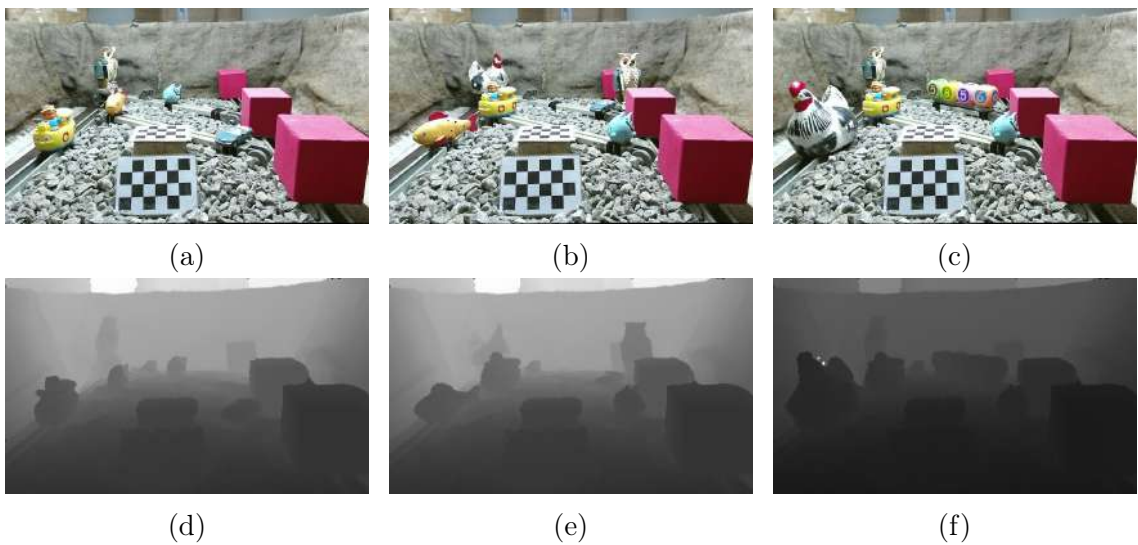


Figure 5.6: RGB-D image samples. Top-row images shows color images from three scenes, whereas bottom-row images are their depth maps after a normalization process. For visualization purpose only, the depth images had their depth values inverted, where darker regions represent closer objects.

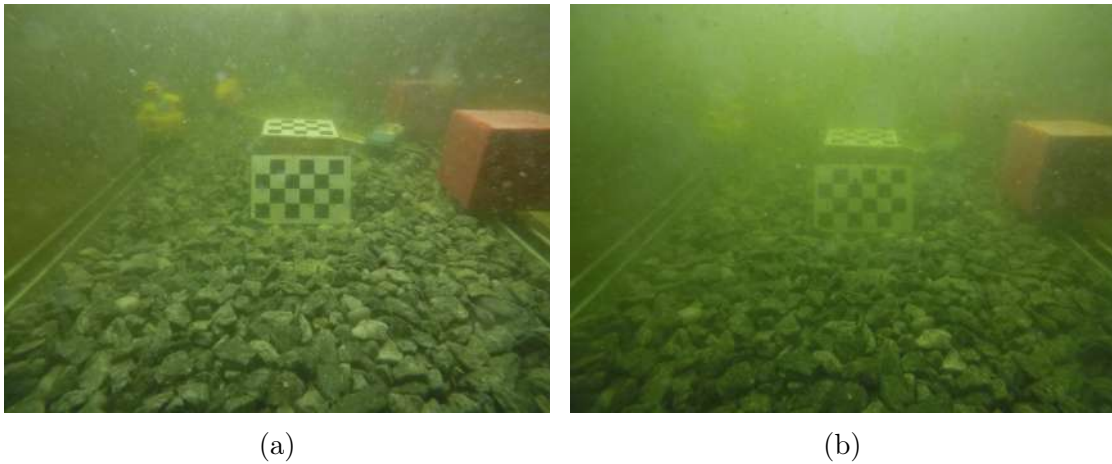


Figure 5.7: Underwater images with non-sifted green tea powder.

Algorithm 2: Color images acquisition using a GoPRO HERO5 Black.

```

1 Place the GoPRO on the camera support;
2 while not all scenes have their images acquired do
3   | Setup new scene;
4   | Fill tank with clean water;
5   | Take the images for this scene in the clean water;
6   | foreach quantity of green tea  $\in [15g, 5g, 5g]$  do
7     | | Add the specified quantity of green tea to the water;
8     | | Take the images for this scene at this turbidity level;

```

Cameras Calibration. In order to correlate RGB pixels with world coordinates, it is needed to perform a process known as camera calibration. This also allows us to remove distortion effects that commonly occur in pinhole cameras and are related to geometric properties of the lenses. It is possible to see on Figure 5.9 that a point in the world coordinate system is mapped to a point C in the camera coordinate system, which is then mapped to an image plane point q defined by the coordinates (u, v) .

When ongoing a calibration process, the aim is to estimate camera intrinsic parameters, which gives us the optical point and the focal length of the camera and its distortion parameters. It also gives us the camera extrinsic parameters, which gives us its rotation matrix and translation vector. Thus, we are able to know its position and orientation according to the world coordinate system.

a) Kinect One Sensor Calibration. To calibrate our Kinect One, we used the calibration tool `kinect2_calibration` available in the `iaikinect2` [59]. The `kinect2_calibration` tool calibrates the IR sensor of the Kinect One to the RGB sensor and the depth measures. It uses OpenCV [10] to calibrate the two cameras.

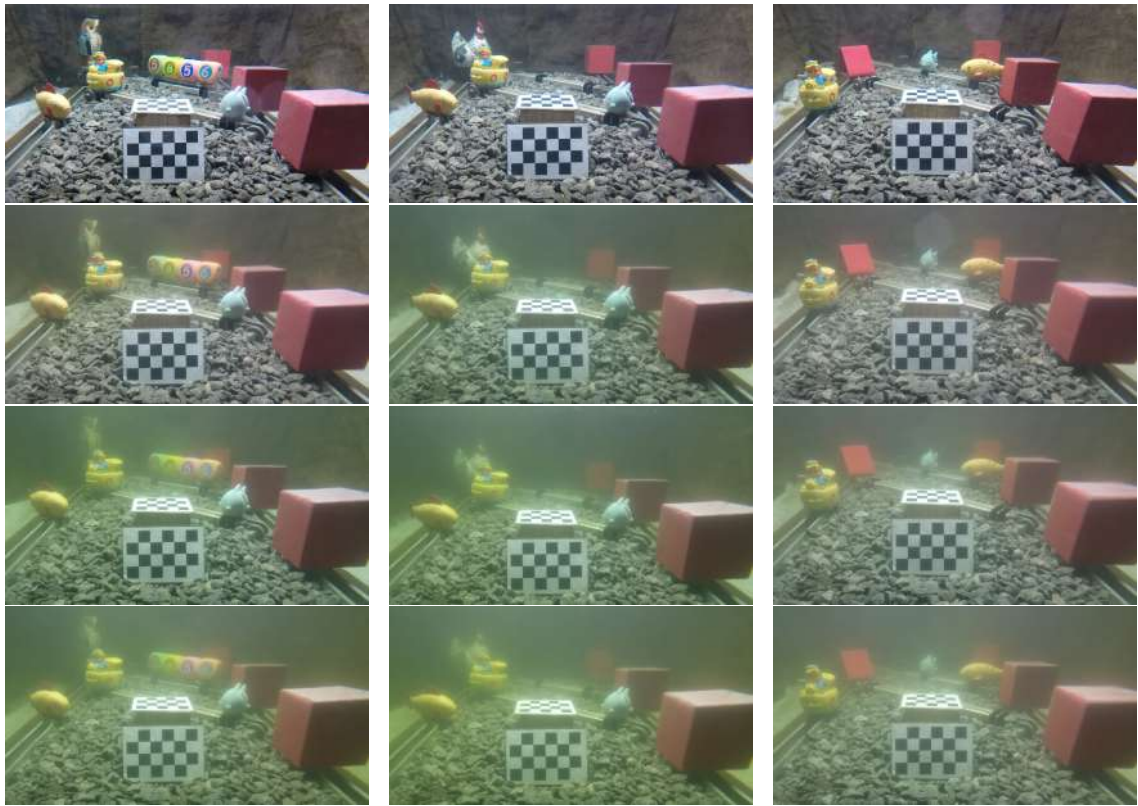


Figure 5.8: RGB image samples. Turbidity levels from top to bottom rows: clean water, 15g green tea, 20g green tea, 25g green tea.

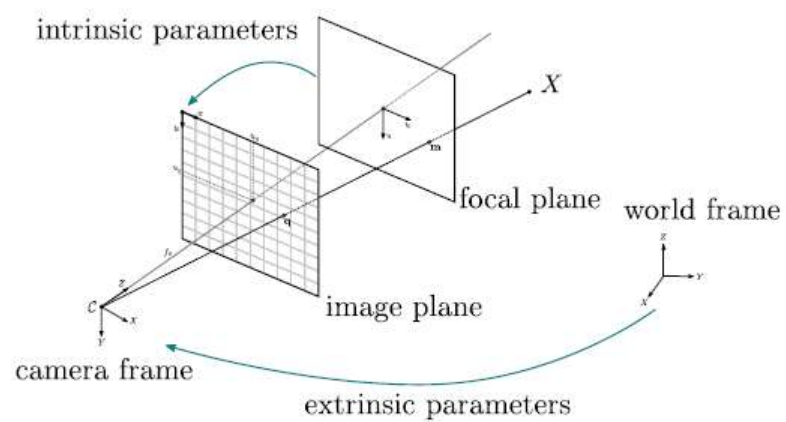


Figure 5.9: Pinhole camera model.

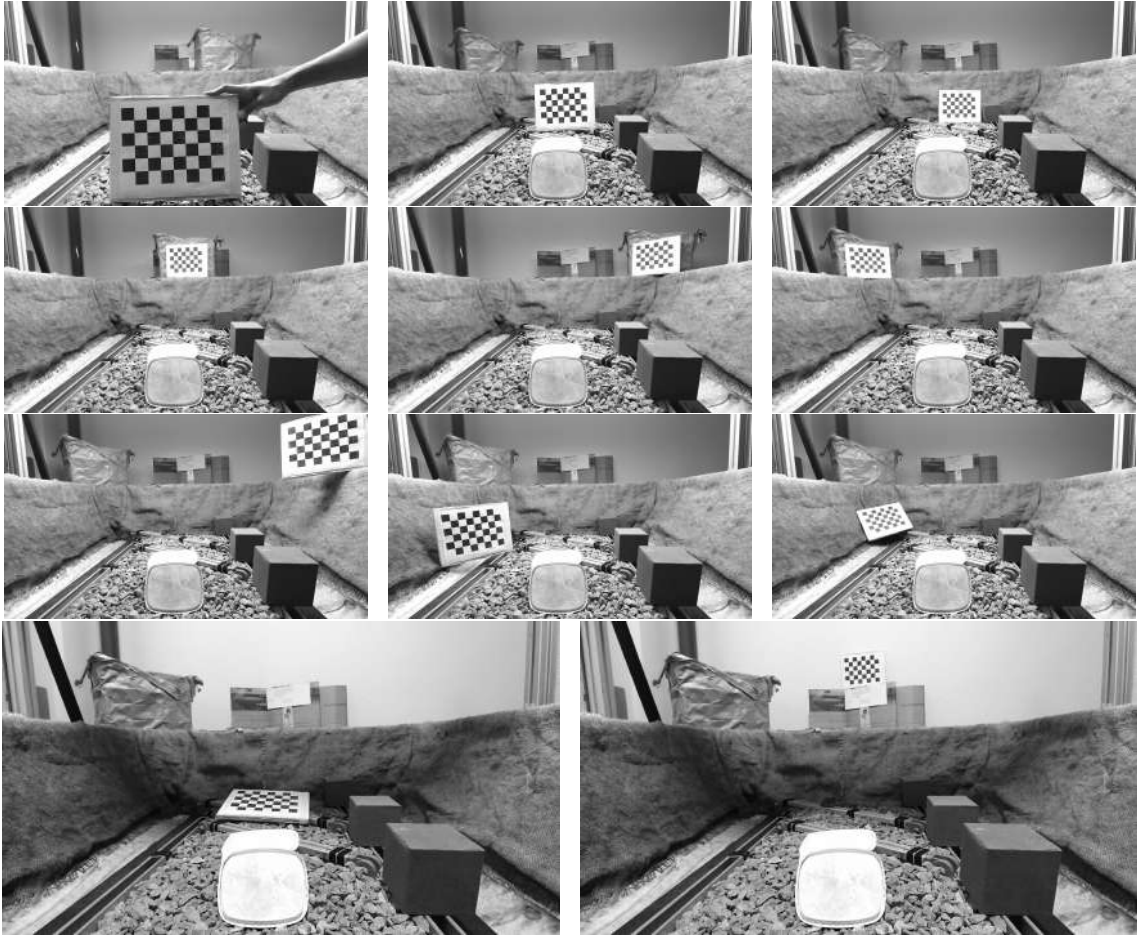


Figure 5.10: Kinect One calibration images. Fixed marks were hidden to avoid unsuccessful camera calibration.

For this calibration, we used a 2.8cm -sided 8×6 chessboard calibration mark. Figure 5.10 shows 11 pictures we took of this mark in different positions and rotations, always visible to both IR and RGB sensors. After following the calibration steps described in `iaikinect2`, we are able to generate the RGB-D images. Each pixel in the depth image is directly related to the same coordinate in the RGB image taken using the Kinect One.

b) Stereo Calibration. We also tried to perform a stereo system calibration composed by the two cameras. We used the MATLAB camera calibration toolbox [49]. Although all the steps were followed, the stereo calibration was not successful.

This may have happened due to the lack of calibration marks in the scenes, nevertheless one of the two marks we placed was much inclined in relation to the cameras. This calibration would allow us to produce synthetic underwater images with near properties to the underwater images we have acquired in our dataset. Additionally, the synthetic dataset could have been created using the clean water images acquired with the GoPRO camera, which have higher resolution and cover a larger world space.

Synthetic Data using RGB-D Images As the RGB images taken using the GoPRO camera were representing underwater mediums limited to the turbidity levels we created using green tea, we needed more underwater images that could resemble a few other aquatic environments. Thus, we selected three attenuation coefficients from [38]: 0.244, 0.65 and 2.07, from 600nm, 700nm and 800nm wavelengths, respectively. We built a look up table for the possible transmission values applied to each location in the image scene. Depth range of Kinect One goes up to 4500mm. This table is filled using the transmission map equation

$$t_{\beta} = e^{-\beta d(x,y)}, \quad (5.2)$$

where d ranges from 0m to 4.5m and β assumes the attenuation coefficients described earlier. Thus, we end up with three transmission look up tables.

Then, we manually selected a patch referring to the farthest region in the underwater RGB images in the first column from Figure 5.8. For each of the turbidity levels present in our real underwater images set, we used the selected patch to compute the background light, expressed as

$$B_c = \frac{1}{N} \sum P_c, \quad (5.3)$$

where $c \in \{reg, green, blue\}$ and N is the number of pixels in the patch P .

Finally, we are able to synthesize our underwater images. For each combination of scene, attenuation coefficient and background light, we apply Equation 4.1 to generate a new image that is added in our artificial dataset. Figure 5.11 lists some images from this synthetic subset of our second dataset.

We consider these datasets essential contributions for the literature. They impact and add value to the work that has been done to tackle the issue of image restoration. We specifically approached the underwater medium and its complexities. In fact, our real underwater images do not cover all possible configurations of subaqueous environments. However, with the depth information of each scene, we are able to apply the most variate set of parameters in order to represent a vast number of aquatic sets. At the same time, any type of participating media may be generated with our RGB-D data.

All datasets are publicly available at our project homepage.



Figure 5.11: Kinect One synthetic images. Attenuation coefficients used from top to bottom rows: 0.244, 0.65, 2.07. These coefficients are displayed in Table 5.1, respectively referring to 600nm, 700nm and 800nm wavelengths. Approximated background light from left to right columns: clean water, 15g green tea, 20g green tea, 25g green tea.

Chapter 6

Experiments

In this chapter, we show the results we obtained by applying our methodology on both datasets. We wanted to validate the applicability of these sets in image restoration tasks. Additionally, a series of metrics is used to evaluate the effectiveness of our approach. Section 6.1 defines some evaluation metrics applied to compare our method to a few relevant approaches. In Section 6.2, we discuss the processes taken to apply the proposed methodology in *UVision18*. Section 6.3 describes *UVision19* usage and experimentation. Qualitative and quantitative analyses are presented, comparing our method to some relevant works in the field of underwater image enhancement and restoration.

6.1 Evaluation Metrics

For results comparison, we used the **UCIQE** metric proposed by Yang and Sowmya [62]. They state this metric is in accordance with the human visual assessment for underwater image quality, correlating statistical measures of chroma, contrast, and saturation with this quality index.

To obtain the image chroma we first decompose it in luminance (L), a and b bands from the CIELab color space [24]. Thus, chroma can be computed as

$$C = \sqrt{a^2 + b^2}. \quad (6.1)$$

The **UCIQE** value is given by

$$UCIQE = c_1 \times \sigma_C + c_2 \times con_L + c_3 \times \mu_S, \quad (6.2)$$

where σ_c is the standard deviation of chroma, defined as

$$\sigma_c = \sqrt{\frac{\sum (C - \mu_C)^2}{N}}, \quad (6.3)$$

μ_C is the average of chroma, con_L is the contrast of luminance, computed as

$$con_L = \frac{\sum \max_{1\%} L - \sum \min_{1\%} L}{\sum \max_{1\%} L + \sum \min_{1\%} L} \quad (6.4)$$

and μ_s is the average of saturation, given by

$$\mu_s = \frac{1}{N} \sum \frac{C}{L}. \quad (6.5)$$

Furthermore, $c_1 = 0.4680$, $c_2 = 0.2745$, $c_3 = 0.2576$ are weighting factors that control the contribution of each quality metric. They were estimated through a 4-fold cross-validation training process, by the authors of this metric.

6.2 Experiments on UVision18

During the training phase of the first experiments, we selected a subset of 300 images from the 642 synthetic images to perform a fine-tuning of DehazeNet. This subset contains RGB images along with their transmission maps. The fine-tuning of the network is performed following the specifications described by the authors in their paper [13]. After 1,500 epochs of supervised training, we switched the process to a self-supervised phase of 1,180 epochs.

We used three subsets of different datasets in the self-supervised part of training. The first two subsets are composed of 40 images from the underwater-related scene categories of the SUN dataset [60] and 60 images from a dataset created by Nascimento et al.

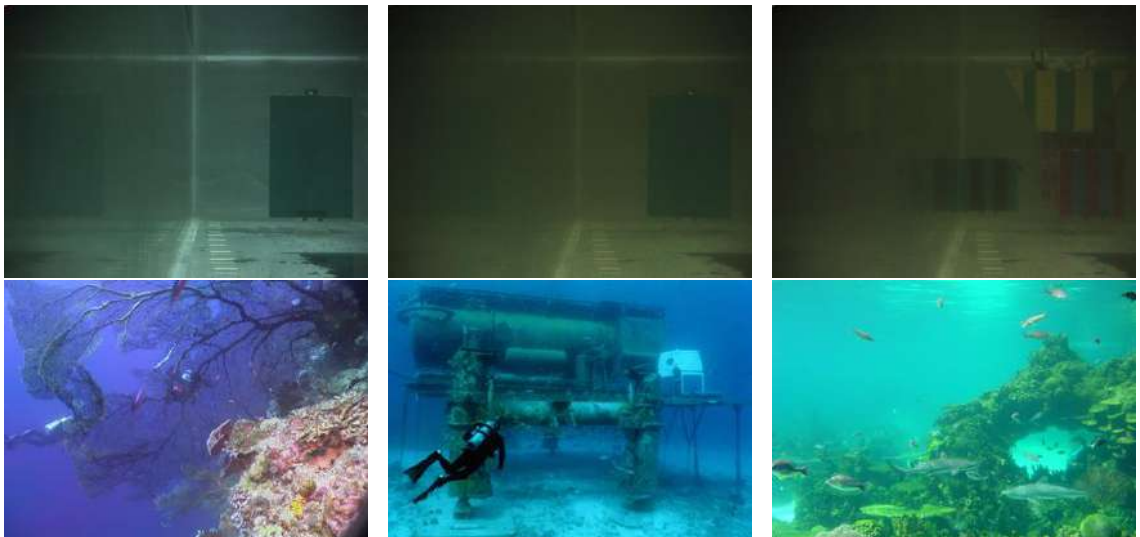


Figure 6.1: Underwater images sample taken from [40] and SUN dataset [60].

Table 6.1: Visual quality using [UCIQE](#) metric (best in bold).

Scene	Original	He	Tarel	Dreus	Ancuti	Ours
<i>ancuti1</i>	0.42465	0.50867	0.41273	0.48448	0.58765	0.65037
<i>ancuti2</i>	0.41477	0.47404	0.43678	0.50476	0.59013	0.56471
<i>ancuti3</i>	0.42631	0.57482	0.46343	0.53477	0.65185	0.62751
<i>galdran</i>	0.49783	0.62876	0.52362	0.64161	0.64333	0.67015
<i>eustice</i>	0.50363	0.61666	0.55290	0.55068	0.63356	0.69135
<i>fish</i>	0.56627	0.67276	0.57636	0.70121	0.66976	0.75171
<i>ocean</i>	0.40795	0.57136	0.43580	0.64708	0.61660	0.66903
<i>reef1</i>	0.61081	0.68628	0.60182	0.63525	0.65471	0.67233
<i>reef2</i>	0.69870	0.72402	0.68316	0.71520	0.71784	0.73244
<i>reef3</i>	0.54392	0.66470	0.57455	0.63816	0.70512	0.67891

[40]. Figure 6.1 illustrates a sample of these subsets. We also added 70 images from the underwater set we built, described in Section 5.1.

During the self-supervised training phase, we apply Equations 4.5 and 4.6 to compute the restored image. This image is normalized for values between $[0,1]$. We restricted $\sum \lambda_X = 1$ to make the [IQM](#) score to lie in between $[0,1]$, where 1 stands for the best and 0 for the worst restoration, respectively. The values of lambdas were defined empirically, being $\lambda_C = 0.25$, $\lambda_A = 0.45$, $\lambda_E = 0.05$ and $\lambda_G = 0.25$. The weights of DehazeNet are initialized with a pre-trained model provided by Cai et al. [13]. This model was designed and trained to restore air images.

To evaluate our trained network, we applied another subset of images often used by the community and available in the work of Ancuti et al. [3]. Although we did not use this dataset during training, it served as a subject of comparison between our approach and other methods. First column of Figure 6.2 shows these images.

We compared our results against four different techniques: [DCP](#) [25] estimates the depth map of the scene by taking the dark channel, which is used in the restoration process; [UDCP](#) [19], a DCP-like prior that does not take into account the red band when computing the dark channel; Tarel and Hautiere [56] perform a linear-complexity function based on median filtering to restore the image; Ancuti et al. [3] restore the image by executing a multi-scale fusion process combining four estimated weight maps.

Table 6.1 shows the [UCIQE](#) scores. We ran all the experiments using the [UCIQE](#) source code provided by the authors. When comparing the restorations using [UCIQE](#) metric, one can see that some methods perform better in different images, but ours outperforms the majority of them. We could achieve the best results in 6 of the 10 images.

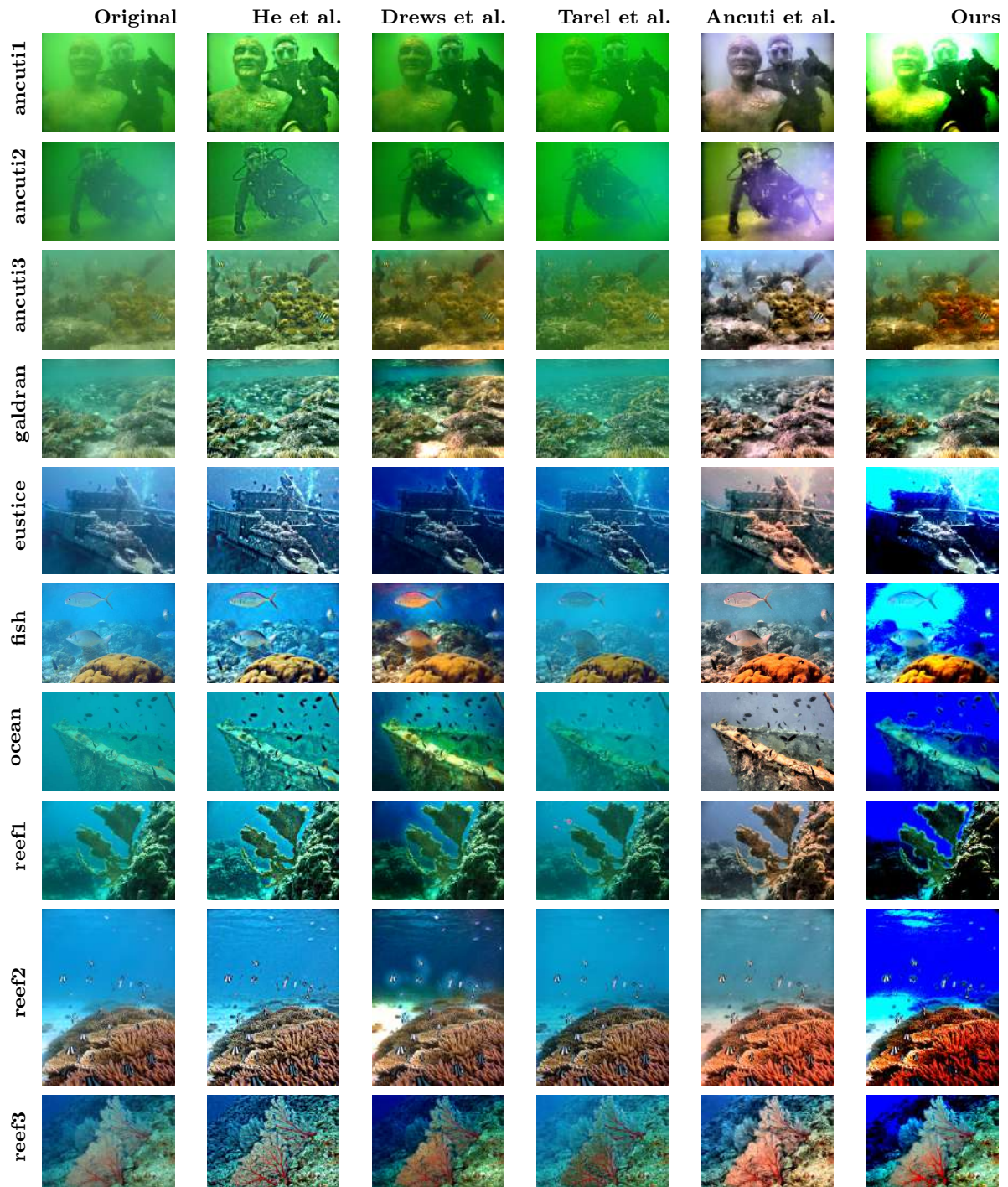


Figure 6.2: UVision18: Images from [3] dataset restored by five different approaches, including ours.

6.3 Experiments on UVision19

The first dataset we built lacks complexity in terms of usable information, such as scene depth and water turbidity control. This leads to a non-successful application of that dataset in underwater images restoration approaches. Aiming to validate the new dataset

and highlight its importance for underwater image restoration techniques, we decided to perform a new training, once again using the air-images-pretrained DehazeNet model as baseline. In total, three training phases were executed. In each part of this experiment, we applied a learning rate of $1e - 08$, decay of $1e - 05$, momentum of 0.95 and batch size of 2.

First, we used 280 images from the real subset to train the CNN model following our quality-based training, i.e., using the underwater images quality metrics we selected to minimize the loss during training. We tested this trained model in the remaining 16 images from the same dataset (Figure 6.3a), all these images are from a scene not included in the training set. After 140 epochs, the network started to perform promising restoration. Although, as we can see in the results displayed in Figure 6.3b, the borders of the objects in the scene were still blurry.

Thus, a supervised training was done, using images of the synthetic subset. We decided to include all the 264 images in the training step, testing the trained model in the same 16 images we used in the first batch of training. After a few epochs, we could see some improvement. Figure 6.3c illustrates the restorations achieved in this phase. Edges of the object present in the scene were less blurry than previous results.

In the third batch of training, we used the best model trained on the second phase, which was a result of a supervised training. From this model, we performed a new quality-based training using the IQM loss function. Figure 6.3d shows the test results from this phase.

6.4 Parametrization

Our convolution network has three convolutional layers: 16 5×5 filters in CONV1, 16 3×3 , 5×5 and 7×7 filters in CONV2, and a 6×6 filter in CONV3. An element-wise maximum operation is applied to the outputs of the first layer, producing four feature maps which are concatenated and fed to the second convolutional layer. We use a 7×7 max-pooling of stride 1 in the concatenated outputs of the second layer and a RELU function in the last layer output.

Hyperparameters Optimization. In an attempt to use the best values for the weights of our quality-driven loss, we performed a grid search algorithm, guided by the UCIQE

metric. We tuned the parameters: λ_C , λ_G , λ_E and λ_A , which specify the contribution factor for the contrast, gray-world prior, border integrity and accutance metrics, respectively. These parameters were assigned values of 0.25, 0.45 or 0.65. We also included two optimizers in the grid search: Adam and SGD, both with learning rate of 0.0005 and decay of 0.00025.

We run the algorithm for 20 epochs each combination of parameters using a 3-fold optimization. As best combination, the grid search returned the value of 0.25 for all IQM weight factors and SGD as the best optimizer. It took two weeks to run the algorithm.

After optimizing these parameters, we performed another self-supervised training in our network. However, the results we obtained were worse than the ones we achieved during the first experiments we performed in *UVision19*. Visualizing Figure 6.3e, the restored images lost most of the 3D structure information, also having their color distribution saturated by the heavy restoration process. This may have happened due to the poor selection of parameters during the grid search algorithm. If we used a broad number of parameter values for the algorithm to select from we could have had a better optimization for the hyper-parameters. Moreover, parameters such as learning rate and decay should have been optimized for a better learning process.

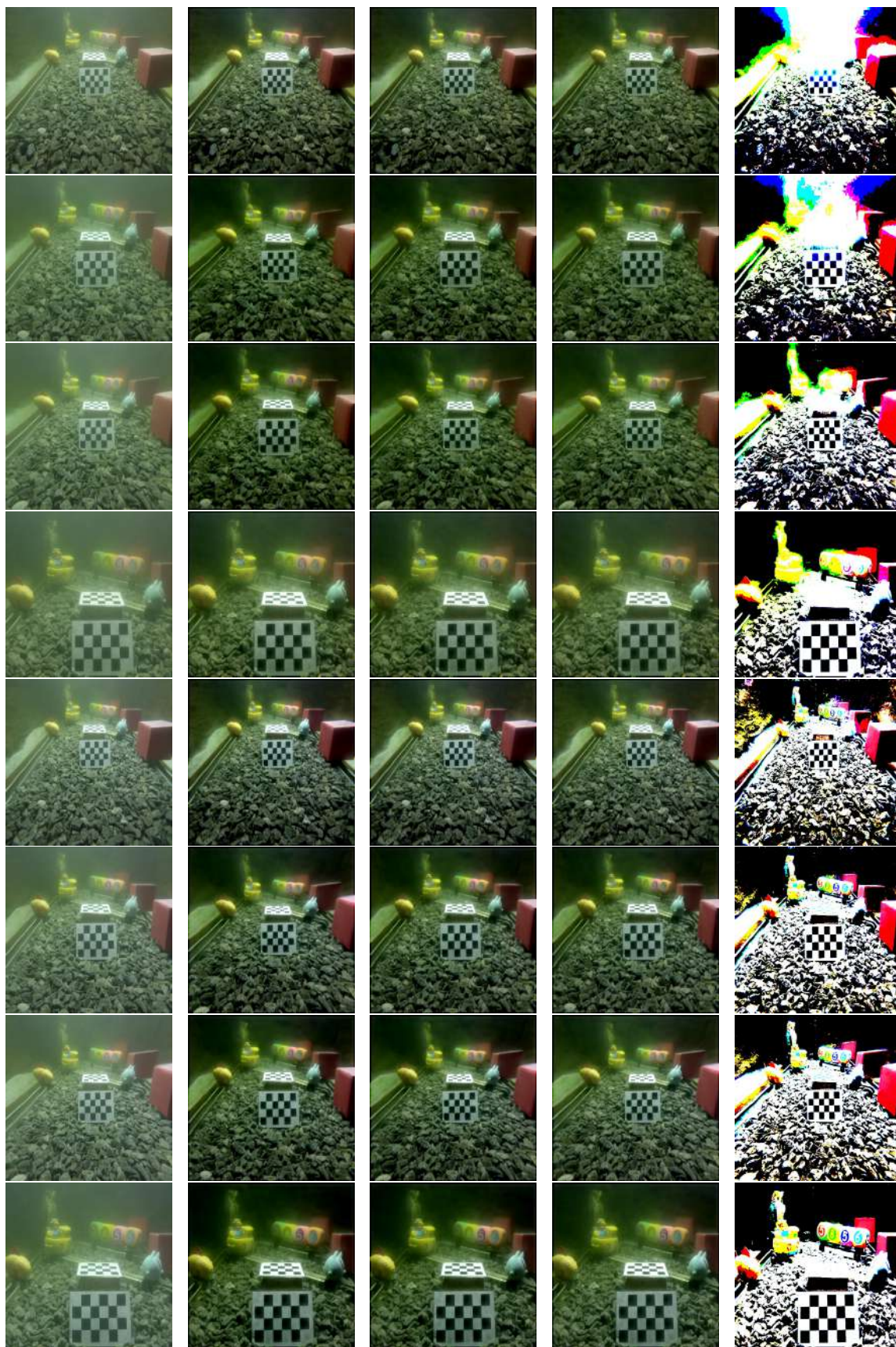
Table 6.2 shows the **UCIQE** results for each of the restorations displayed in Figure 6.3. We can assume that this metric alone cannot be used to visually qualify the restoration of underwater images. By looking at the metric values, the best results (except for Scene 7, by a low) were obtained by the grid search restoration, which are clearly the ones that most degraded the original images.

Our experiments show that our methodology could be applied in underwater image restoration activities. It is clear that it needs improving. Visually, our results are not the best taking into consideration the works we have compared our approach with.

At the same time, studying our datasets may lead us to new ideas in how to modify our pipeline and change the architecture of the convolutional neural network. These modifications could give us satisfiable results, physically plausible and visually pleasing. It is also worth noting that our datasets usage do not limit to underwater images restoration purposes. They could be applied image recovering from various participating medium.

Table 6.2: Visual quality using [UCIQE](#) metric (best in bold) on UVision19 experiments.

Scene	Original	MSE	IQM	Mixed	GridSearch
<i>0</i>	0.47742	0.55919	0.55358	0.54814	0.62626
<i>1</i>	0.45805	0.55462	0.55207	0.54684	0.64429
<i>2</i>	0.45494	0.55511	0.55314	0.54843	0.64382
<i>3</i>	0.47377	0.59151	0.59915	0.59410	0.63794
<i>4</i>	0.49120	0.56670	0.56656	0.56427	0.63818
<i>5</i>	0.47674	0.57339	0.56914	0.56428	0.63299
<i>6</i>	0.47528	0.57028	0.56985	0.56928	0.64242
<i>7</i>	0.49184	0.61793	0.61551	0.61361	0.61662
<i>8</i>	0.51375	0.58490	0.58404	0.58108	0.64768
<i>9</i>	0.50897	0.58975	0.58702	0.58544	0.65984
<i>10</i>	0.50497	0.59293	0.59172	0.58902	0.62803
<i>11</i>	0.52449	0.62825	0.62695	0.62523	0.64993
<i>12</i>	0.55845	0.60835	0.60432	0.60182	0.65126
<i>13</i>	0.56386	0.60692	0.60726	0.60519	0.65309
<i>14</i>	0.56296	0.60872	0.60806	0.60652	0.63348
<i>15</i>	0.57359	0.61675	0.62029	0.62181	0.63655



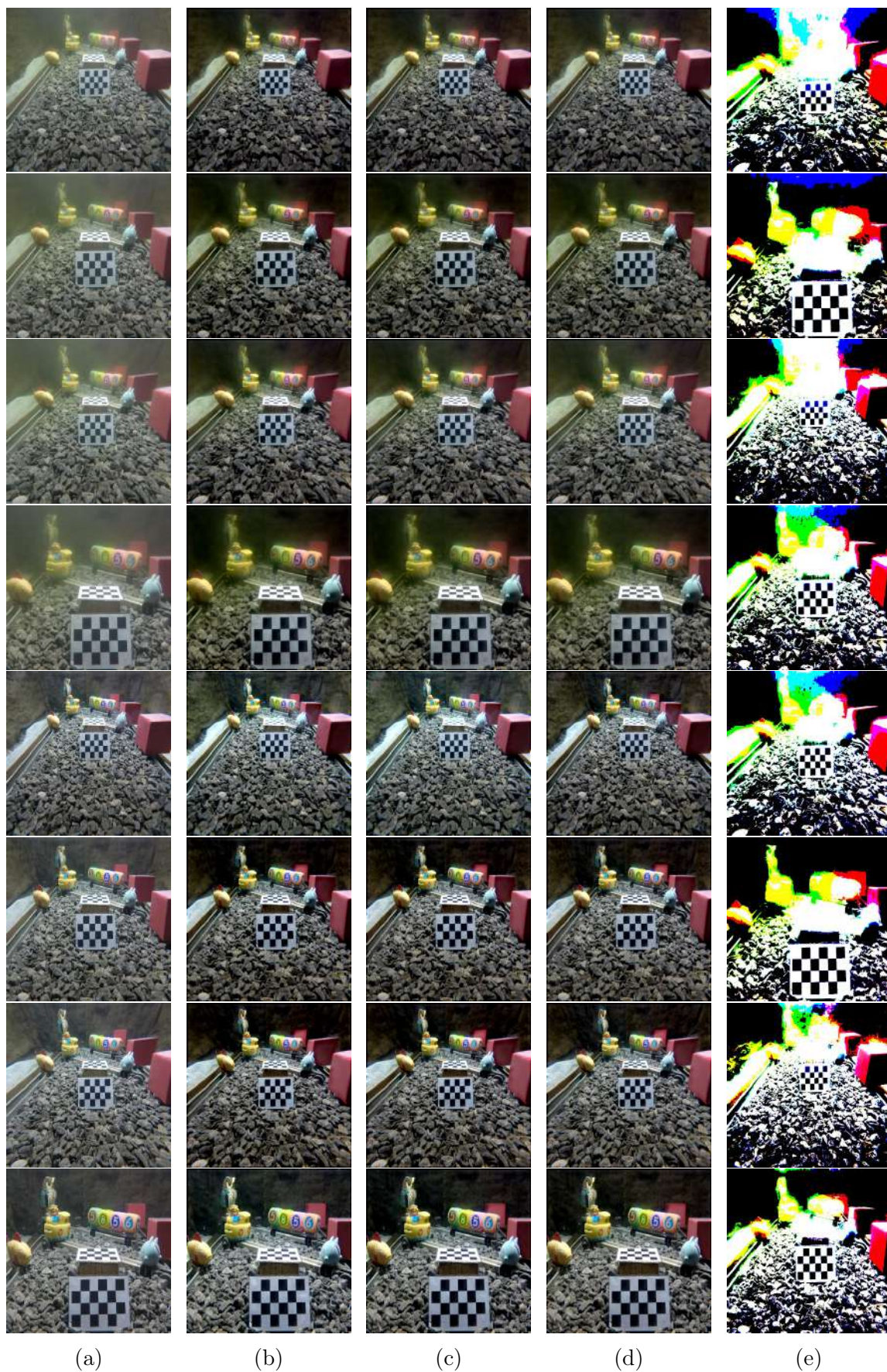


Figure 6.3: Experiments applied to UVision19 dataset images: (a) original image; (b) MSE-based training results; (c) IQM-based training restorations; (d) Mixed training results; (e) Optimized hyper-parameters training restorations.

Chapter 7

Conclusion

In this thesis we discussed about image restoration, an important field of computer vision. Throughout the years, researchers have achieved impressive results when trying to recover information on images acquired from scenes undergoing medium effects as sand, fog, rain or water. Our main focus was in the underwater environment, a complex participative medium where light suffers absorption and scattering effects. The resulting image is a blurred and color-shifted version of the scene we would see if there was no interference in the paths that light follows from source to camera sensor.

We proposed an end-to-end self-supervised approach that after training receives an underwater image as input, estimating the transmission map of the scene, which is then used to compute the background light. Both are applied to the inverse of the underwater image formation model, which combines them to the input image in order to restore information lost by effects such as absorption and scattering of the light that travels from a source to the camera sensor.

Our approach has been complemented and validated by two datasets we built during our research. The first dataset contains a few underwater images taken using a water tank under controlled configurations, where we used green tea to simulate a few turbidity levels. Along with this real subset of images, a simple synthetic dataset was produced using [PBRT](#), a rendering tool based on physical principles such as the absorption coefficient of water. Whereas our second dataset contains thoroughly planned scenes illuminated by a controlled light source. We have again applied portions of green tea in the water to simulate different turbidity levels. Additionally, with the use of depth data we acquired from the scene, we were able to construct a synthetic dataset which can be expanded.

Analyzing our quantitative results using the [UCIQE](#) metric, we can see that our method successfully restores lost scene information. As for the qualitative results, as we do not apply color correction algorithms in the image restoration pipeline, our results

seem unpleasant when compared to other approaches.

7.1 Future Work

During our experiments, some aspects involving the structure and visible features of the imaged scenes caught our attention. We could not, for example, solve the issue of color shifting during the restoration process. Although the network was not trained for that purpose, we could apply some sort of color correction algorithm before or after the image was processed by our system.

Another component we did not tackle was the water refraction indices, which affect the way camera lenses image the scene, causing distortion of objects in the final images. This may or may not affect the restoration process. When we perform camera calibration, this operation gives us the intrinsic parameters, including the distortion coefficients, which refer to characteristics of the camera itself. In a future work, we could use this information to undistort the images, perform their restoration and compare the results with our previous restorations with distorted images. Thus, we could conclude if our network would still estimate the same transmission maps when using structure-corrected images.

While building our second synthetic dataset, we wanted to correlate our RGB images, taken with a GoPRO camera, with our RGB-D images, taken using a Kinect One sensor. To accomplish this, we needed to perform a stereo calibration to get the transformation matrix from one system to another. As we only had two calibration marks per image, the transformation matrix could not be efficiently and correctly estimated. Thereby, the synthesis of artificial underwater images had to be done using only the RGB-D images. For this reason, we propose to further go back to try and produce this synthetic data using both RGB and RGB-D images. This would allow us to contribute with a more complete dataset. In this, the synthetic data would be more accurate with real turbidity levels as the ones we used in our real dataset.

We conclude our research presented in this thesis by stating that underwater image restoration will become a common process in the following years. The field of computer vision and machine learning are evolving exponentially at the same pace of technology. New embedded technology will facilitate the execution of activity as devices are more capable of processing more complex algorithms.

As we discussed before, the images we acquired are not usage-limited to underwater image recovering. They could be applied in a variety of studies which refer to image restoration and those that need depth information in their pipeline execution. We hope the idea we approached throughout this text along with the datasets we delivered may be a relevant contribution to the community of image restoration.

Bibliography

- [1] Radhakrishna Achanta, Sheila Hemami, Francisco Estrada, and Sabine Susstrunk. Frequency-tuned salient region detection. In *Computer vision and pattern recognition, 2009. cvpr 2009. ieee conference on*, pages 1597–1604. IEEE, 2009.
- [2] Tinku Acharya and Ajoy K Ray. *Image processing: principles and applications*. John Wiley & Sons, 2005.
- [3] Cosmin Ancuti, Codruta Orniana Ancuti, Tom Haber, and Philippe Bekaert. Enhancing underwater images and videos by fusion. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 81–88. IEEE, 2012.
- [4] Cosmin Ancuti, Codruta O Ancuti, Christophe De Vleeschouwer, Rafael Garcia, and Alan C Bovik. Multi-scale underwater descattering. In *Pattern Recognition (ICPR), 2016 23rd International Conference on*, pages 4202–4207. IEEE, 2016.
- [5] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:2481–2495, 2017.
- [6] Walysson V Barbosa, Henrique GB Amaral, Thiago L Rocha, and Erickson R Nascimento. Visual-quality-driven learning for underwater vision enhancement. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 3933–3937. IEEE, 2018.
- [7] Wagner Barros, Erickson R Nascimento, Walysson V Barbosa, and Mario FM Campos. Single-shot underwater image restoration: A visual quality-aware method based on light propagation model. *Journal of Visual Communication and Image Representation*, 2018.
- [8] Stéphane Bazeille, Isabelle Quidu, and Luc Jaulin. Automatic underwater image pre-processing. In *in Proceedings of the Characterisation du Milieu Marin (CMM'06)*, 2006.
- [9] Dana Berman, Shai Avidan, et al. Non-local image dehazing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1674–1682, 2016.

-
- [10] Gary Bradski and Adrian Kaehler. *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc.", 2008.
- [11] Gershon Buchsbaum. A spatial processor model for object colour perception. *Journal of the Franklin institute*, 310(1):1–26, 1980.
- [12] Jean Bégaint, Dominique Thoreau, Philippe Guillotel, and Christine Guillemot. Region-based prediction for image compression in the cloud. *IEEE Transactions on Image Processing*, 27(4):1835–1846, 2018.
- [13] Bolun Cai, Xiangmin Xu, Kui Jia, Chunmei Qing, and Dacheng Tao. Dehazenet: An end-to-end system for single image haze removal. *IEEE Transactions on Image Processing*, 25(11):5187–5198, 2016.
- [14] John Canny. A computational approach to edge detection. In *Readings in computer vision*, pages 184–203. Elsevier, 1987.
- [15] Damon M Chandler. Seven challenges in image quality assessment: past, present, and future research. *ISRN Signal Processing*, 2013, 2013.
- [16] Shuai Di, Honggang Zhang, Chun-Guang Li, Xue Mei, Danil Prokhorov, and Haibin Ling. Cross-domain traffic scene understanding: A dense correspondence-based transfer learning approach. *IEEE Transactions on Intelligent Transportation Systems*, 19(3):745–757, 2018.
- [17] Paulo Drews, Erickson R Nascimento, Mario FM Campos, and Alberto Elfes. Automatic restoration of underwater monocular sequences of images. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 1058–1064. IEEE, 2015.
- [18] Paulo Drews, Emili Hernández, Alberto Elfes, Erickson R Nascimento, and Mario Campos. Real-time monocular obstacle avoidance using underwater dark channel prior. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 4672–4677. IEEE, 2016.
- [19] Paulo LJ Drews, Erickson R Nascimento, Silvia SC Botelho, and Mario Fernando Montenegro Campos. Underwater depth estimation and image restoration based on single images. *IEEE computer graphics and applications*, 36(2):24–35, 2016.
- [20] Raanan Fattal. Single image dehazing. *ACM transactions on graphics (TOG)*, 27(3):72, 2008.
- [21] Ke Gu, Guangtao Zhai, Weisi Lin, and Min Liu. The analysis of image contrast: From quality assessment to automatic enhancement. *IEEE Transactions on Cybernetics*, 46(1):284–297, 2016.

- [22] Irwan Prasetya Gunawan and Mohammed Ghanbari. Image quality assessment based on harmonics gain/loss information. In *IEEE International Conference on Image Processing 2005*, volume 1, pages I–429. IEEE, 2005.
- [23] Junwei Han, Dingwen Zhang, Gong Cheng, Nian Liu, and Dong Xu. Advanced deep-learning techniques for salient and category-specific object detection: a survey. *IEEE Signal Processing Magazine*, 35(1):84–100, 2018.
- [24] David Hasler and Sabine Süsstrunk. Measuring colorfulness in natural images. In *Human Vision and Electronic Imaging*, 2003.
- [25] Kaiming He, Jian Sun, and Xiaoou Tang. Single image haze removal using dark channel prior. *IEEE transactions on pattern analysis and machine intelligence*, 33(12):2341–2353, 2011.
- [26] Jules S Jaffe. Computer modeling and the design of optimal underwater imaging systems. *IEEE Journal of Oceanic Engineering*, 15(2):101–111, 1990.
- [27] Ran Kaftory, Yoav Y. Schechner, and Yehoshua Y. Zeevi. Variational distance-dependent image restoration. In *Computer Vision and Pattern Recognition (CVPR), 2007 IEEE Conference on*, pages 1–8, 2007.
- [28] Sergey Krivenko, Mikhail Zriakhov, Vladimir Lukin, and Benoit Vozel. Mse and psnr prediction for adct coder applied to lossy image compression. In *2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, pages 613–618. IEEE, 2018.
- [29] Jon C Leachtenauer, William Malila, John Irvine, Linda Colburn, and Nanette Salvaggio. General image-quality equation: Giqe. *Applied optics*, 36(32):8322–8328, 1997.
- [30] Yann LeCun, Patrick Haffner, Léon Bottou, and Yoshua Bengio. Object recognition with gradient-based learning. In *Shape, contour and grouping in computer vision*, pages 319–345. Springer, 1999.
- [31] Jie Li, Katherine A Skinner, Ryan M Eustice, and Matthew Johnson-Roberson. Watergan: Unsupervised generative network to enable real-time color correction of monocular underwater images. *IEEE Robotics and Automation Letters*, 3(1):387–394, 2018.
- [32] Tsung-Yi Lin, Priyal Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *IEEE transactions on pattern analysis and machine intelligence*, 2018.

- [33] Mingna Liu and Xin Yang. A new image quality approach based on decision fusion. In *2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery*, volume 4, pages 10–14. IEEE, 2008.
- [34] Risheng Liu, Xin Fan, Minjun Hou, Zhiying Jiang, Zhongxuan Luo, and Lei Zhang. Learning aggregated transmission propagation networks for haze removal and beyond. *IEEE transactions on neural networks and learning systems*, 0(99):1–14, 2018.
- [35] Huimin Lu, Yujie Li, Tomoki Uemura, Zongyuan Ge, Xing Xu, Li He, Seiichi Serikawa, and Hyoungseop Kim. Fdcnet: filtering deep convolutional network for marine organism classification. *Multimedia tools and applications*, 77(17):21847–21860, 2018.
- [36] BL McGlamery. A computer model for underwater camera systems. In *Ocean Optics VI*, volume 208, pages 221–232. International Society for Optics and Photonics, 1980.
- [37] Samy Metari and François Deschenes. *A new convolution kernel for atmospheric point spread function applied to computer vision*. IEEE, 2007.
- [38] Curtis D Mobley. *Light and water: radiative transfer in natural waters*. Academic press, 1994.
- [39] Zak Murez, Tali Treibitz, Ravi Ramamoorthi, and David Kriegman. Photometric stereo in a scattering medium. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3415–3423, 2015.
- [40] Erickson Nascimento, Mario Campos, and Wagner Barros. Stereo based structure recovery of underwater scenes from automatically restored images. In *Computer Graphics and Image Processing (SIBGRAPI), 2009 XXII Brazilian Symposium on*, pages 330–337, 2009.
- [41] Erickson R Nascimento. *Estimação Automática de Parâmetros de Modelos para Restauração de Imagens de Cenas Subaquáticas*. In *Dissertação, UFMG. Belo Horizonte-MG*, 2008.
- [42] Øyvind Ødegård, Aksel Alstad Mogstad, Geir Johnsen, Asgeir J Sørensen, and Martin Ludvigsen. Underwater hyperspectral imaging: a new tool for marine archaeology. *Applied optics*, 57(12):3214–3223, 2018.
- [43] Diana Pagliari and Livio Pinto. Calibration of kinect for xbox one and comparison between the two generations of microsoft sensors. *Sensors*, 15(11):27569–27589, 2015.
- [44] Yan-Tsung Peng and Pamela C Cosman. Underwater image restoration based on image blurriness and light absorption. *IEEE Transactions on Image Processing*, 26(4):1579–1594, 2017.

- [45] Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically based rendering: From theory to implementation*. Morgan Kaufmann, 2016. <https://github.com/mmp/pbrt-v3> (visited: 2019-06-25).
- [46] Abdul Rehman and Zhou Wang. Reduced-reference image quality assessment by structural similarity estimation. *IEEE Transactions on Image Processing*, 21(8): 3378–3389, 2012.
- [47] Wenqi Ren, Si Liu, Hua Zhang, Jinshan Pan, Xiaochun Cao, and Ming-Hsuan Yang. Single image dehazing via multi-scale convolutional neural networks. In *European Conference on Computer Vision*, pages 154–169. Springer, 2016.
- [48] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic foggy scene understanding with synthetic data. *International Journal of Computer Vision*, pages 1–20, 2018.
- [49] Davide Scaramuzza, Agostino Martinelli, and Roland Siegwart. A toolbox for easily calibrating omnidirectional cameras. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5695–5701. IEEE, 2006.
- [50] Yoav Y Schechner and Nir Karpel. Recovery of underwater visibility and structure by polarization analysis. *IEEE Journal of oceanic engineering*, 30(3):570–587, 2005.
- [51] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.
- [52] Hamid R Sheikh and Alan C Bovik. Image information and visual quality. In *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages iii–709. IEEE, 2004.
- [53] Mark Sheinin and Yoav Y Schechner. The next best underwater view. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3764–3773, 2016.
- [54] Irwin Sobel and Gary Feldman. A 3x3 isotropic gradient operator for image processing. *a talk at the Stanford Artificial Project in*, pages 271–272, 1968.
- [55] Thomas G Stockham. Image processing in the context of a visual model. *Proceedings of the IEEE*, 60(7):828–842, 1972.
- [56] Jean-Philippe Tarel and Nicolas Hautiere. Fast visibility restoration from a single color or gray level image. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 2201–2208, 2009.

- [57] Emanuele Trucco and Adriana T. Olmos-Antillon. Self-tuning underwater image restoration. *IEEE Journal of Oceanic Engineering*, 31(2):511–519, 2006.
- [58] L-T Wang, Nathan E Hoover, Edwin H Porter, and John J Zasio. Ssim: a software leveled compiled-code simulator. In *Proceedings of the 24th ACM/IEEE Design Automation Conference*, pages 2–8. ACM, 1987.
- [59] Thiemo Wiedemeyer. IAI Kinect2. https://github.com/code-iai/iai_kinect2, 2014 – 2019.
- [60] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3485–3492. IEEE, 2010.
- [61] Takehisa Yamakita, Hiroyuki Yokooka, Yoshihiro Fujiwara, Masaru Kawato, Shinji Tsuchida, Shojiro Ishibashi, Tadayuki Kurokawa, and Katsunori Fujikura. Image dataset of ophiuroid and other deep sea benthic organisms in 2015 extracted from the survey off sanriku, japan, by the research following the great east japan earthquake 2011. *Ecological research*, 33(2):285–285, 2018.
- [62] Miao Yang and Arcot Sowmya. An underwater color image quality evaluation metric. *IEEE Transactions on Image Processing*, 24(12):6062–6071, 2015.
- [63] Lintao Zheng, Hengliang Shi, and Shibao Sun. Underwater image enhancement algorithm based on clahe and usm. In *Information and Automation (ICIA), 2016 IEEE International Conference on*, pages 585–590. IEEE, 2016.
- [64] Jieying Zhu and Nengchao Wang. Image quality assessment by visual gradient similarity. *IEEE Transactions on Image Processing*, 21(3):919–933, 2011.