UNDERWATER IMAGE QUALITY ENHANCEMENT BY USING CONVOLUTIONAL NEURAL NETWORK

YAP JIA JUN

THIS DISSERTATION IS SUBMITTED FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR OF SCIENCE WITH HONOURS

MATHEMATICS WITH COMPUTER GRAPHICS PROGRAMME

FACULTY OF SCIENCE AND NATURAL RESOURCES

UNIVERSITY MALAYSIA SABAH

2024

STUDENT'S DECLARATION

I hereby declare that this dissertation has been composed solely by myself, the work contained here is my own effort except for the citations and work that has been cited along with their respective sources.

YAP JIA JUN

(BS20110220)

19th January 2023

SUPERVISOR VALIDATION

SIGNATURE

1. **SUPERVISOR**
   (PROF MADYA DR ABDULLAH BADE)

____

ACKNOWLEGEMENT

First and foremost, I would like to express my sincere gratitude towards my supervisor, Prof. Madya Dr. Abdullah Bade, for giving me an opportunity to do research and providing me invaluable guidance throughout this final year project. He had offered his precious time and wisdom that enlighten me on the right path in accomplishing the dissertation with insightful comments and suggestions. I am very thankful to my supervisor who assisted me at every stage of the research project.

Besides, I would like to offer my special thanks to every lecturer of my course, Mathematics with Computer Graphics who has educated us the knowledge and skills that were acquired to overcome the challenges that arise in the thesis. Not only that, their spiritual supports had motivated me to carry on and decent thesis on time.

Moreover, I would also like to thank my examiner, Mr. Rechard Lee, for his generous attitude in spending his valuable time in examining and analysing my full thesis. He also gave feedbacks and suggestion to further improve the weakness that lies in the project draft. Hence, second opportunity was given to revise and complete my final thesis.

Furthermore, I am deeply grateful to have my family members with me when I was doing my research in two semesters. It was not a simple task to complete the thesis writing alone. Thus, I would not have accomplished this dissertation without their unwavering support and belief in me.

Last but not least, I would like to extend my sincere thanks to my friends for gibing me full support and willing to exchange the idea of work. I feel grateful to have them in discussion as they always inspired me and advised me in developing a satisfactory system in this project.

ABSTRACT

Image processing is a challenging field of study that focuses on enhancing visual information for human interpretation and implementing autonomous machine perception. Image enhancement, a common method in image processing, involves improving the visual quality of an image. Image enhancement in underwater surveillance is essential for various applications, including oceanic geological exploration, resource exploitation, ecological research. However, the visual perception of underwater imaging is affected by various environmental factors such as attenuation and absorption of light. The primary challenge is that underwater images often appear with haze-like effect, low contrast and sharpness and colour bias. This paper proposes a new deep learning-based underwater image quality enhancement by combining the image formation model and white balance model to improve the overall quality of underwater images in terms of haze removal, improve contrast and sharpness, and colour balance. Experimental results demonstrate the advantages of the proposed method in improving visual quality by eliminating the influence of underwater environmental factors, removed haze, increased contrast and sharpness, restore and balancing colours. These results are further supported by quantitative metric, indicating improvement of 2.80%, 4.67%, 5.22% and 2.69% in entropy, PCQI, UIQM and UCIQE respectively, as compared to image formation model. In terms of future research, it is recommended that the study integrates both models into a unified framework and explores expanded dataset training.

PENINGKATAN KUALITI IMEJ BAWAH AIR DENGAN MENGGUNAKAN ALGORITMA BERASASKAN RANGKAIAN NEURAL KONVOLUSI

ABSTRAK

Pemprosesan imej merupakan bidang kajian yang mencabar yang memberi tumpuan kepada penambahbaikan maklumat visual untuk interpretasi manusia dan pelaksanaan persepsi mesin berautonomi. Penambahbaikan imej, kaedah yang biasa dalam pemprosesan imej, melibatkan peningkatan kualiti visual imej. Penambahbaikan imej dalam pengawasan dalam air adalah penting untuk pelbagai aplikasi, termasuk penerokaan geologi lautan, eksploitasi sumber, penyelidikan ekologi. Walau bagaimanapun, persepsi visual imej bawah air terjejas oleh pelbagai faktor persekitaran seperti pengecilan dan penyerapan cahaya. Cabaran utama adalah imej dalam air sering kelihatan keruh, kontras dan ketajaman yang rendah dan bias warna. Kajian ini mencadangkan satu kaedah penambahbaikan kualiti imej dalam air yang baru berdasarkan pembentukan model imej dan model imbangan putih untuk meningkatkan keseluruhan kualiti imej dalam air dari segi penyingkiran kesan keruh, peningkatan kontras dan ketajaman, serta imbangan warna. Hasil eksperimen menunjukkan kelebihan kaedah yang dicadangkan dalam meningkatkan kualiti visual dengan menghapuskan pengaruh faktor persekitaran daalm air, menghilangkan kesan keruh, meningkatkan kontras dan ketajaman, mengembalikan dan menyeimbangkan warna. Hasil ini disokong oleh metrik kuantitatif, yang menunjukkan peningkatan sebanyak 2.80%, 4.67%, 5.22% dan 2.69% masing-masing dalam entropi, PCQI, UIQM dan UCIQE berbanding dengan model pembentukan imej. Dari segi penyelidikan akan datang, disyorkan agar kajian mengintegrasikan kedua-dua model ke dalam rangkaian bersatu dan meneroka latihan dataset yang diperluaskan.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

α	-	angle of incident light
β	-	angle of refracted light
η	-	refractive index of water
L_T	-	total light intensity received by camera
L_d	-	direct light
L_f	-	forward scattered light
L_b	-	backward scattered light
$J(x)$	-	true intensity value of the scene
$\beta(x)$	-	coefficient of the attenuation of light
d	-	distance from the camera to the scene
$t(x)$	-	direct transmission mapping
$L_\infty(x)$	-	background light
$c \in \{R, G, B\}$	-	three colour channels of red, green and blue
β_C^D	-	attenuation coefficients for direct-transmission light
β_C^B	-	attenuation coefficients for backward scattered light
V_D	-	attenuation coefficients dependencies of direct-transmission
V_B	-	attenuation coefficients dependencies of backward scattered
i_{en}	-	enhanced image
I_{ref}	-	reference image
MAX	-	maximum possible pixel intensity value
MSE	-	mean square error
$l(x, y)$	-	luminance similarity
$c(x, y)$	-	contrast similarity
$s(x, y)$	-	structural similarity
$UICM$	-	colourfulness measure
$UISM$	-	sharpness
$UIConM$	-	the contrast
σ_{ch}	-	standard deviation of chroma
μ_{sat}	-	average saturation
I	-	Image

T^{I}	-	Tensor
B	-	Backward Scatter Estimation
D	-	Direct-Transmission Estimation
T^{DH}	-	Tensor of dehaze image
T^{AWB}	-	Tensor with auto white balance
I^{e}	-	Enhanced Image with original resolution

LIST OF ABBREVIATIONS

2D	-	2 Dimension
3D	-	3 Dimension
RGB	-	Red, Green and Blue
IR	-	Infrared ray
CNN	-	Convolutional neural network
GAN	-	Generative adversarial network
WB	-	White Balance
AWB	-	Auto white balance
ReLU	-	Rectified linear unit
Tanh	-	Hyperbolic tangent
PReLU	-	Parametric rectified unit
ELU	-	Exponential linear unit
SUID	-	Synthetic underwater image dataset
TURBID	-	Dataset from artificial underwater environment
UFO-120	-	Dataset for Simultaneous Enhancement and Super-Resolution
UIEB	-	Underwater Image Enhancement Benchmark
EUVP	-	Enhancing Underwater Visual Perception
MSE	-	Mean Square Error
PSNR	-	Peak signal-to-noise ratio
SSIM	-	Structural similarity index measure
UIQM	-	Underwater Image Quality Measure
UCIQM	-	Underwater Colour Image Quality Evaluation
PCQI	-	Patch-Based Contrast Quality Index
OpenCV	-	Open Computer Vision
UML	-	Unified Modelling Language
PIL	-	Python Image Library
GUI	-	Graphic User Interface

CHAPTER 1

INTRODUCTION

1.1 Overview

Digital image processing involves the manipulation of digital images using a computer. A digital image is defined as a two-dimensional function with spatial coordinates, and the intensity at any coordinates represents the image's grey level. Images are composed of discrete picture elements, commonly referred to as pixels. The field covers a wide spectrum of applications, utilizing the entire electromagnetic spectrum from gamma to radio waves, including sources like ultrasound, electron microscopy, and computer-generated images.

There is no clear boundary between image processing, image analysis, and computer vision. Image processing is sometimes defined as a discipline where both the input and output are images, but this distinction may be limiting. Computer vision aims to emulate human vision and is a branch of artificial intelligence. Processes in this continuum are categorized as low, medium and high level. Low-level processes involve basic operations like noise reduction and contrast enhancement. Medium-level processes include segmentation, description, and classification of objects. Higher-level processing involves recognizing objects and performing cognitive functions associated with vision. The overlap between image processing and image analysis occurs in the recognition of individual regions or objects. Digital image processing encompasses processes with both image inputs and outputs, as well as those extracting attributes from images, including object recognition (Gonzalez and Woods, 2008).

Image enhancement is a common method of digital image processing, which involves improving the visual quality of an image by adjusting its brightness, contrast, sharpness, and colour balance. Image restoration is another application, which involves removing noise, blur, and other distortions from an image to improve its clarity and quality without losing the image details.

Digital image processing has a wide range of applications in various fields, such as medical imaging, remote sensing, surveillance, robotics, and multimedia. In surveillance, it is used for analysing and detecting objects and activities in the image and video footage. Surveillance has been carried out in many places such as caves, dessert, and underwater. Among all the surveillance-based techniques, underwater surveillance technology is less developed till the early twenty-first century. Functionality like waterproof, high pressure withstands, long battery life, brightness, and good focus capability are the basic characteristics that an underwater camera should have. It is more challenging when building an underwater camera compared to a camera that uses for normal outdoor surveillance (Subudhi *et al.*, 2023).

In discovering the underwater world, digital image processing plays an important role in improving the quality and reliability of the information collected from underwater environments. Underwater surveillance has various applications including the protection of rare species of water animals, exploration of underwater, prediction of natural disasters, conservation of marine life and ecosystem, etc. The underwater environment such as low illumination, limited visibility, low brightness, and underwater haze has caused the quality image and video degradation and decolorization and made the surveillance process more challenging.

In the realm of underwater surveillance, the quality of captured images is significantly compromised by the inherent challenges posed by light attenuation and absorption in aquatic environments. Light attenuation, caused by reflection, refraction, and both forward and backward scattering, leads to a notable reduction in light intensity, introducing a pervasive haze-like effect in underwater images. Figure 1.3 demonstrates how the image is affected when taking a picture underwater.

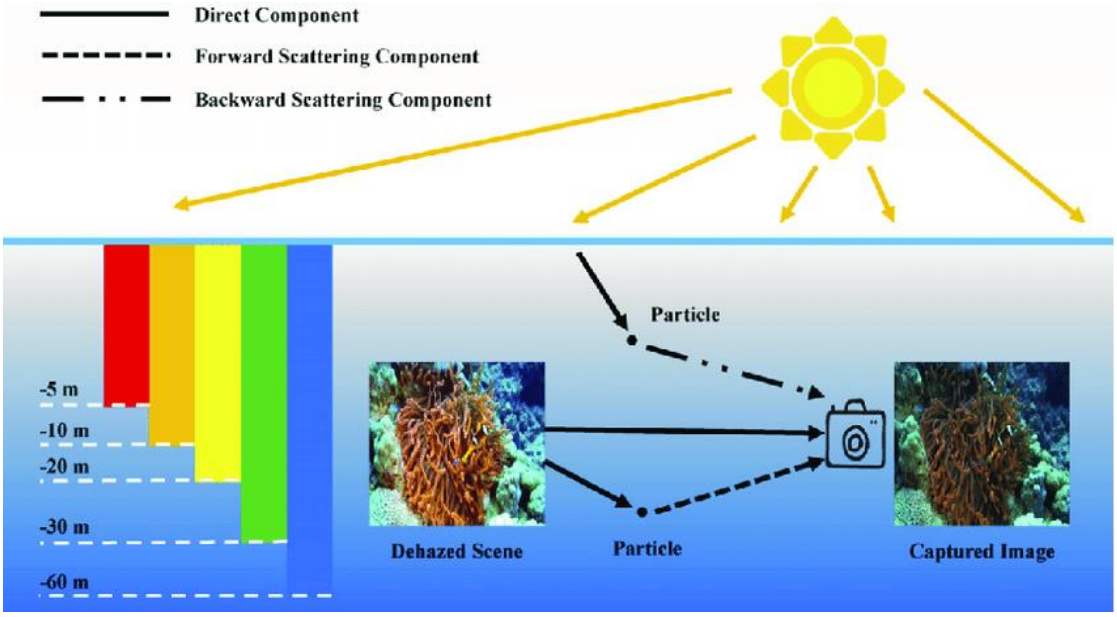


Figure 1.3 Underwater Imaging Model (Source: Tao *et al.*, 2021)

Sunlight is the source of light and is also known as natural illumination. When the light enters the water, this is where refraction for light occurs due to the density of air and water being different. The black dots are the tiny particles and haze present in the water. The formation of the captured image consists of the combination of three types of components: direct component (direct light), where the light is directly reflected from the scene; forward scattering component refers to the light reflected by the scene to the suspended particles in the water and deflected to the camera and backward scattering component refer to the naturally illuminated light scatters by the particle before reaching the camera. These factors cause the captured underwater images in low quality, blurry, haziness, and low illumination (Subudhi *et al.*, 2023).

Furthermore, the absorption of light is wavelength-dependent, influencing the colours that are absorbed as light travels through water. As illustrates in Figure 1.3, each colour has its own depth limit before being absorbed by water. Red, orange, and yellow colours are absorbed when they reach the depth of water at 5m, 10m, and 20m respectively while green and blue colours can travel further down to 60m. Beyond this point, the images will start to become darker due to most of the light has already been scattered and absorbed (Tao *et al.*, 2021). This phenomenon results in a prevalent bluish and greenish tone in underwater images, dictated by the absorption characteristics at different depths.

In order to improve the underwater image quality, this process required a large dataset for researchers to analyse and test different types of image enhancement techniques to obtain the best result of the enhance underwater image. At the same time, it will cost a large amount of time for researchers to manually process all the data. This is why deep learning technique has taken a part in image enhancement to solve the problem. Deep learning is a type of technique under machine learning that teaches a computer to do something that human needs. It consists of many layers of neural network process data which attempt to simulate how the human brain works and enable it to be trained and learn from the data (LeCun *et al.*, 2015). Prior to this, deep learning was mainly used in object detection and image classification by extracting the feature and recognizing the pattern from the image. When it comes to image enhancement, it automatically learns complex features from images and uses the two maps with the degraded image to produce high-quality images. Besides, by training the raw image with their corresponding high-quality image, the computer can calculate the differences such as mean square error between both images and try to minimize the difference to achieve a better-enhanced image. Thus, the deep learning method is one of the important fields to be explored, studied and implement in image processing to bring great improvement and benefit for human beings to obtain better achievement.

1.2 Problem Background

Underwater imaging presents a unique set of challenges that hinder the quality and clarity of captured images. One significant problem is the presence of a haze-like effect caused by the attenuation of light as it travels through water. This effect reduces visibility and obscures details, making it difficult to discern objects in underwater scenes. Additionally, underwater images often suffer from low contrast and sharpness, further diminishing their overall quality and usability for various applications such as scientific research and marine exploration. Another issue contributing to the degradation of underwater images is colour imbalance and degradation resulting from the absorption of light by water molecules. This absorption alters the colour composition of the captured images, leading to unnatural colour casts, such as a greenish or bluish tint, which distorts the true appearance of underwater scenes and complicates image analysis (Subudhi *et al.*, 2023).

Current approaches to overcome these issues primarily rely on traditional image processing techniques. For example, dark channel prior is used for dehazing, histogram equalization to improve contrast and sharpness and white balance to remove the colour bias in the image. While these methods have shown some effectiveness in improving certain aspects of underwater image quality, they often fall short in addressing all the complexities of underwater image degradation comprehensively. Traditional approaches often employ single-purpose methods that lack adaptability across diverse underwater scenes. Manual feature extraction, reliance on heuristic algorithms, and the need for image-specific enhancement parameters further limit the efficacy of these traditional techniques. The shortcomings of single-image processing methods become apparent, as they struggle to restore lost details and sharpness in the intricate and varied patterns present in underwater scenes (Hu *et al.*, 2022).

Despite progress in image processing, several challenges remain in enhancing underwater images. Firstly, current techniques may struggle to fully address the haze-like effect, leaving behind residual haziness and reducing visibility in underwater scenes. Furthermore, efforts to improve contrast and sharpness may unintentionally introduce artifacts or noise, which can degrade the overall image quality. Additionally, colour correction algorithms may find it challenging to accurately restore natural colour balance, resulting in persistent colour imbalances and inconsistencies that impact the fidelity of underwater images.

Another approach to improving underwater image quality involves using underwater imaging models. These models aim to estimate various components affecting image formation underwater, such as backward scatter and direct transmission, and then utilize these estimations to recalculate the actual intensity and colour of the scene. One of the popular and most used underwater imaging models was developed by Jaffe-McGlamery (Yang *et al.*, 2019). However, this approach may still face challenges in fully capturing the complexities of underwater light propagation. Factors such as varying water conditions, particulate matter, and surface reflections can introduce additional uncertainties and limitations to the modelling process.

1.3 Problem Statement

As outlined in the problem background, underwater image enhancement faces problem in degradation of underwater image quality due to light attenuation and absorption, influenced by reflection, refraction, and scattering phenomena. These factors contribute to underwater images appearing hazy, with low contrast and sharpness, and exhibiting colour bias. The primary challenge is to devise a comprehensive solution that effectively addresses these degradation factors. The central focus is on developing a more versatile and adaptive solution capable of addressing the complexities inherent in diverse underwater scenes.

1.4 Aim

This project aims to implement a deep learning-based image enhancement method for underwater colour images by combining dual Neural Network models to produce high-quality result.

1.5 Objective

The objectives of this project are:

i. To implement underwater image formation model by removing haze-like effect in the underwater images.
ii. To develop a fusion of white balance model and underwater image formation model to increase sharpness, contrast and colour balance in the degraded underwater colour images.

1.6 Scope

i. In this project, only static image will be used for enhancement purpose.
ii. Deep learning algorithm is the main method to carry out underwater image enhancement process.
iii. Dehazing and colour restoration is the main aspect to be overcome and improve in this proposed method.
iv. Only online underwater image dataset will be used for training process.

v. Real-time processing is not included in this project.

1.7 Justification

The deep learning-based technique has become popular for researchers to study and apply in existing applications to achieve better results in a short time without human effort to handle the process. A high-quality underwater image is required for researchers in scientific research of marine life, exploration of underwater, and preserving underwater ecosystems. Through the enhanced image, they are able to get information such as the actual colour of the species underwater, hidden details, and the terrain underwater. Therefore, any problems such as disasters or issues underwater can be easily detected when scientists obtain a clear and true image of underwater. When the deep learning model is trained, it can process a large dataset in a short time and provide accurate results in the long run.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter discusses an overview of underwater image enhancement, its concept, method, issues, limitations, etc. It begins with the fundamentals of digital image processing, where a digital image is formed and processed using a computer. Image enhancement is a fundamental of image processing used to increase the visual quality, perception, and interpretability of digital images. Contemplative knowledge about underwater image formation is important so that appropriate techniques and algorithms can be designed and used to enhance images. The available methods of underwater image enhancement can be divided into two primary classes: Traditional method and deep learning-based method. Pros and cons are present in these two main techniques that could facilitate the application of underwater image enhancement in fulfilling different demands. As a result of its outstanding performance in computer vision, deep learning-based technology for enhancing underwater images is growing rapidly. Hence, this project is reviewing the deep learning-based method and the output quality of underwater images.

2.2 Digital Image

A digital image represents visual information in a format that can be stored, processed, and displayed using digital devices. It comprises a grid of tiny picture elements called pixels, each containing colour and brightness information. In mathematical definition, an image is defined as a two-dimensional function, $f(x, y)$, where x and y are coordinates of the plane, and the amplitude of f at any pair of coordinates is known as the intensity of the image at that point. A digital image is an image when x, y, and the amplitude value of f are all finite and discrete quantities (Gonzalez and Woods, 2008). Digital images are widely used in various fields, including photography, computer graphics, medical imaging, and more.

2.3 Digital Image Processing

Digital image processing can be described as the field of study that focusses on the analysis, manipulation, and enhancement of digital images by applying computer algorithms. In the process involves applying mathematical operations, algorithms and computer-based methods to modify the pixels value of digital photographs. The results can be photos or a set of representative features or properties of the source images that can be used for a variety of purposes. The intention of digital image processing is to enable humans to obtain a high-quality image or a subset of the original image's basic features. Imaging system or sensors are not able to automatically capture and extract useful targets from the image. It is different with human visual system, which is capable of adapting itself to various circumstances. Digital image processing has significant applications in various fields, including medicine, surveillance, remote sensing, robotics, computer graphics, and multimedia. It enables tasks such as medical diagnosis, object recognition, image-based modelling, and visualization. However, nearly no field of technical work is unaffected by digital image processing in some way. The advancements in digital image processing techniques and algorithms have greatly contributed to the development of advanced imaging systems and applications (McAndrew, 2015).

2.4 Attenuation of Light in Water

In general, when light enters the water medium, it is subjected to reflection, refraction, absorption, and scattering by the water's surface, the water itself, and any suspended particles present. All of these processes will reduce the intensity or brightness of light as it travels through water, and this is known as attenuation of light (Nababan *et al.*, 2021).

2.4.1 Reflection of Light

Light undergoes reflection when it travels from one medium to another medium with different density. The reflection will occur when light encounters the boundary between two different mediums. So, when natural sunlight encounters the water surface, a portion of it is reflected back. The reflection is affected by the incidence angle as well as the refractive index of the water and the surrounding medium. As illustrated in Figure 2.1, angle of incidence, α is the angle when the light initially strikes the surface, measured from a line perpendicular to the flat and calm water surface while refractive index of water is determined by the wavelength of the light and the purity of water. The angle of reflection, γ is equal to the angle of incidence. The intensity of reflected light can be determined by Fresnel's law. In most situations, only a small fraction of the sunlight is reflected when it hit the water surface, unless the angle at which it strikes is very close to the horizon. Reflection contributes to the overall attenuation of light in water.

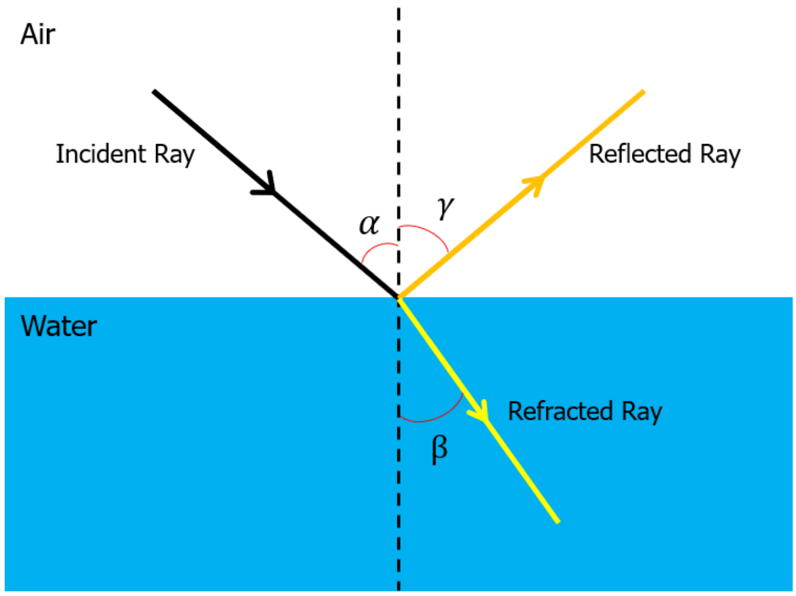


Figure 2.1 Reflection and refraction of light to water

2.4.2 Refraction of Light

Refraction of light refers to the bending or change in direction that occurs when light passes from one transparent medium to another due to light travels at different speeds in different materials. As depicted in Figure 2.1, when light transitions from air to water, a portion of the light is reflected at the water's surface, while the remaining light continues into the water, experiencing a decrease in speed and altering its trajectory. This change in the path of transmission is a result of the difference in refractive indices between air and water. The degree of bending is described by Snell's law,

$$\frac{\sin\alpha}{\sin\beta} = \eta \tag{2.1}$$

where:

α is the angle of incident light,

β is the angle of refracted light,

η is the refractive index of water.

2.4.3 Adsorption of Light

After the light goes through the surface of the water, the water takes in the electromagnetic radiation from the sun rather than the light. Electromagnetic radiation is a type of energy which made up of electromagnetic waves that are defined by their wavelength and frequency. The human eye is limited to perceiving a narrow range of electromagnetic waves known as visible light, which is just a fraction of the entire electromagnetic spectrum. The electromagnetic spectrum represents various forms of electromagnetic radiation. Figure 2.2 provides an illustration of the complete electromagnetic spectrum, highlighting the part of visible light that is perceivable by humans and appears as white in colour. However, white light is actually a combination of multiple wavelengths of light, each representing a different colour within the visible light spectrum. As shown in the figure 2.2, red light emits electromagnetic radiation with the longest wavelength, around 700 nanometres (nm), while violet light emits electromagnetic radiation with the shortest wavelength, approximately 400 nm. The greater the energy of electromagnetic radiation, the shorter its wavelength.

Back to absorption of light in water, the absorption is mainly caused by dissolved substances such as organic matter, minerals, and chlorophyll. Water adsorbs almost all of the infrared ray (IR) within 10 centimetres from the water surface. Since the energy of visible red light has more energy than IR, it can penetrate deeper than IR but yet is more readily absorbed by water than other visible colour of light. Figure 2.3 illustrates the depth that can be penetrated by the different colour lights before fully adsorb by water. The wavelength of the visible light that has greatest penetration up to 227 metres is at around 420 nm, which is on the boundary between blue and violet light. Therefore, at this depth, a photographer only able to capture picture mostly in blue colour tone. Besides, the amount of light energy absorbed is determined by the absorption coefficient, which quantifies the conversion of radiant energy to heat and chemical energy(Downing, 2008). As shown in Figure 2.4, around 400 nm wavelength of the visible light (blue light) has the lowest absorption coefficient of pure water. This justify that amount of radiant energy been absorbed is the lowest at that point and prove again why most of the underwater images are appear in blue colour when captured in the deep water. It is clear that adsorption is the main contribution the attenuation of light in water.

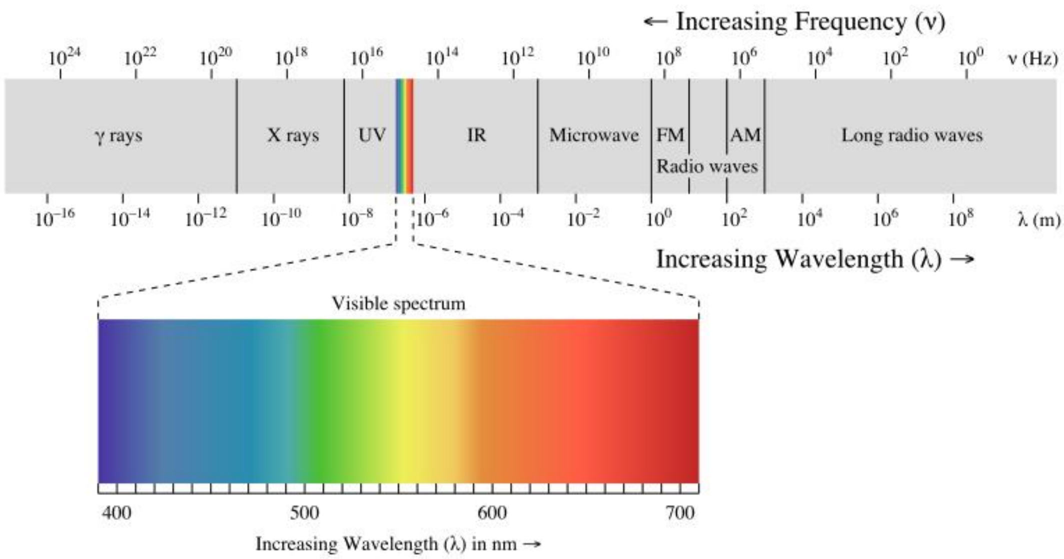


Figure 2.2 The electromagnetic spectrum with visible light highlighted (Source: https://commons.wikimedia.org/wiki/File:EM_spectrum.svg)

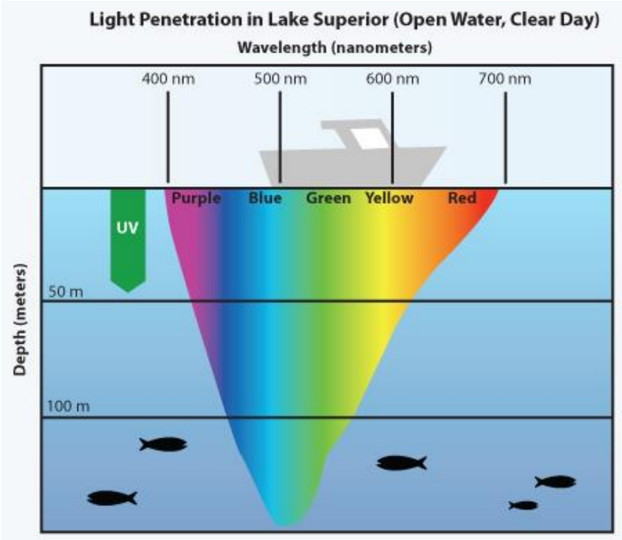


Figure 2.3 The Light Penetration in Lake Superior (Source: http://manoa.hawaii.edu/exploringourfluidearth/physical/ocean-depths/light-ocean)

Figure 2.4 The absorption coefficient of pure water (Source: Downing, 2008)

2.4.4 Scattering of Light

Water contains suspended particles and molecules that scatter light in different directions without changing their wavelength. This scattering can cause light to deviate from its original path depending on the size and concentration of the particles, resulting in a decrease in the overall intensity of light. The scattering coefficient, $b(\lambda)$, is determined by the percentage of energy scattered from a light particle per unit distance travelled in a scattering medium, in cm^{-1} (Downing, 2008). Thus, scattering also will cause the attenuation of light when it travels through water.

2.5 Underwater Image Acquisition

Advancements in sensor technology and high-speed communication strategies have greatly improved the capabilities of surveillance systems. Understanding the acquisition procedure of an underwater image is crucial for further image processing. The process of taking pictures is shown in Figure 2.5, where $l\ (x, y)$ represents the optical energy incident on the object plane in the real world and $r\ (x, y)$ denotes its reflection coefficient. Assuming a perfect and noiseless medium, $I\ (x, y)$ is the image captured by a camera sensor (Subudhi *et al.*, 2023).

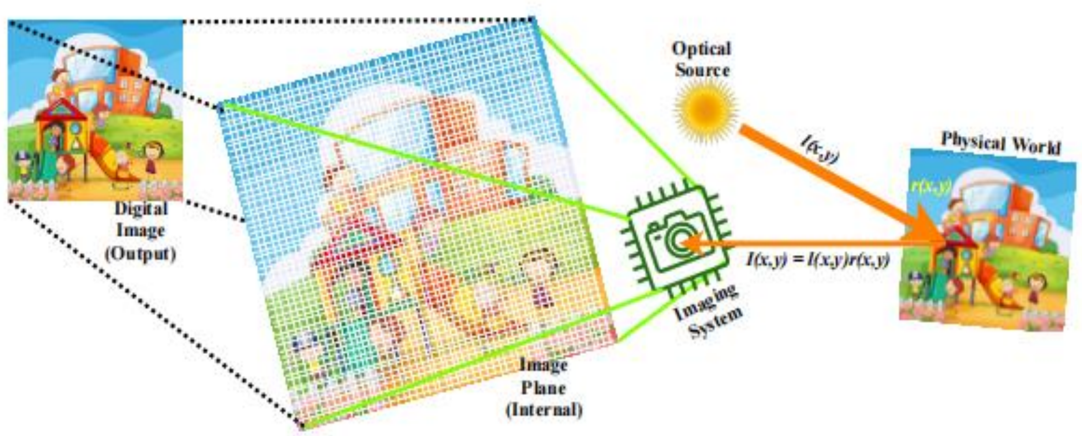


Figure 2.5 Image formation in camera (Source: Subudhi *et al.*, 2023)

2.6 Underwater Imaging Model

Underwater imaging models are mathematical models that aim to simulate the effects of underwater condition on the formation of image. An image is formed when natural light reflected from the object or scene to the digital camera and stored in the form of digital bit string. If imaging is carried out under an ideal medium, where there is no any effect on the reflected light, the exact amount of optical energy reflected from the scene will be sense by the camera and a perfect image is formed (Subudhi *et al.*, 2023). However, the process of formation of an underwater image is much more challenge compared to outdoor image due to the complexity of the environment. It can be affected by the medium, light adsorption and attenuation, scattering, colour distortion, etc.

2.6.1 Jaffe-McGlamery Underwater Imaging Model

The popular and most used underwater imaging model was developed by Jaffe-McGlamery. Figure 2.6 illustrates the light propagation process from a light source to a camera based on Jaffe-McGlamery underwater imaging model. There are several paths to travel the light from source to the scene and to the camera's image plane. The pathways are categorized into three types: direct light, the light directly reflected from the scene without scattered by the suspended particle; forward scattered light, the light reflected from the scene and scattered by the particle in the water; back scattered light, the light reflected from the source and scattered by the particle in the water (Yang *et al.*, 2019).

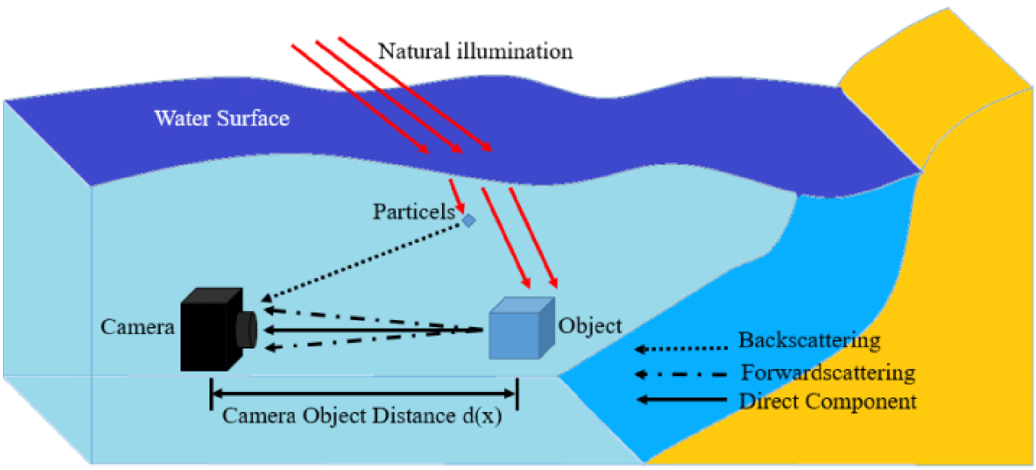


Figure 2.6 Jaffe-McGlamery Underwater Imaging Model (Source: Yang *et al.*, 2019)

Based on Jaffe-McGlamery underwater imaging model, the received light intensity by a camera is the sum of the three pathways

$$L_T = L_d + L_f + L_b \tag{2.2}$$

where:

L_T is the total light intensity received by camera,

L_d is the direct light,

L_f is the forward scattered light,

L_b is the backward scattered light.

2.6.2 Narasimhan Imaging Model

In practical imaging scenarios, the medium through which light travels is often less than ideal, leading to absorption and scattering of light before it reaches the camera. This scattering phenomenon introduces additional optical energy into the surrounding environment (Subudhi *et al.*, 2023). Consequently, the light captured by the camera sensor is not solely derived from the reflected light originating from the scene, which undergoes attenuation within the medium, but also includes contributions from the ambient or surrounding light (Narasimhan & Nayar, 2002). In this model will explain more detail of the formation of underwater image. For direct light L_d, since the light is reflected from the scene to camera without being scattering, considering only attenuation of light. The attenuation of optical energy is assumed to be an exponentially decaying process. Thus, direct light L_d is denoted as

$$L_d = J(x)e^{-\beta(x)d} \tag{2.3}$$

where:

$J(x)$ is the true intensity value of the scene,

$\beta(x)$ is the coefficient of the attenuation of light,

d is the distance from the camera to the scene.

For forward scattered light L_f, it is similar with direct light but the light is disturbed by the underwater suspended particle at small-angle. Therefore, it will cause the image blur and can be expressed as a convolution

$$L_f = L_d \cdot g(x) \tag{2.4}$$

where $g(x)$ is the point spread function (Schechner & Karpel, 2004). However, L_f is omitted in the final formation of underwater image due to it does not contribute significantly to the degradation of an image especially when the camera is nearby the captured scene.

For backward scattered light L_b, it is cause by the scattering of environmental illumination by suspended particle. Let $dL(z,x)$ be the infinitesimally small radiance at distance z and the light with wavelength x. $dL(z,x)$ can be given as

$$dL(z,x) = k\beta(x)e^{-\beta(x)z}dz \tag{2.5}$$

By integrating equation 2.5 from 0 to d. The total radiance can be written as

$$L(d,x) = k\beta(x)\left[-\beta(x)e^{-\beta(x)z}\right]_0^d \tag{2.6}$$

$$L(d,x) = k(1 - e^{-\beta(x)d}) \tag{2.7}$$

When d is set to be infinity, the value of backscatter at infinity (veiling light) can be obtained. For $d \to \infty, L(d,x) \to L(\infty,x) = L_\infty(x)$, then

$$\lim_{d\to\infty} L(d,x) = k(1-0) = k \tag{2.8}$$

$$L_b = L_\infty(x)(1 - e^{-\beta(x)d}) \tag{2.9}$$

According to the equation 2.2, the final light intensity observed by the camera is a combined effect of the direct transmission light and back scattered light and can be rewritten as equation 2.11

$$dL(z,x) = k\beta(x)e^{-\beta(x)z}dz \tag{2.10}$$

$$L_T = J(x)e^{-\beta(x)d} + L_\infty(x)(1 - e^{-\beta(x)d}) \tag{2.11}$$

Assuming $e^{-\beta(x)d}$ as $t(x)$, the equation 2.11 can be simplified as

$$L_T = J(x)t(x) + L_\infty\big(1 - t(x)\big) \tag{2.12}$$

where:

$t(x)$ is the direct transmission mapping,

$L_\infty(x)$ is the background light.

2.6.3 A Revised Image Formation Model

Common methods for enhancing underwater images often simplify the calculation by treating the $t(x)$ of three channels of an RGB image as one mapping, neglecting the variation in the attenuation coefficient β in $t(x)$ between different colour channels. As a result, the enhanced images may still exhibit colour bias. In a study conducted by Akkaynak & Treibitz (2018), they performed numerous underwater experiments and introduced a revised model, expressed in equation 2.13. This revised model considers the optical imaging properties of the underwater environment and adjusts the attenuation coefficient accordingly.

$$L_c(x) = J_c(x)e^{-\beta_C^D(V_D)d} + L_c^\infty\left(1 - e^{-\beta_C^B(V_B)d}\right) \tag{2.13}$$

where:

$c \in \{R, G, B\}$ denotes the three colour channels of red, green and blue,

β_C^D denotes the attenuation coefficients for direct-transmission light,

β_C^B denotes the attenuation coefficients for backward scattered light,

d denotes the transmission distance between scene and camera,

$V_D = \{d, \rho, E, S_c, a, b\}$ and $V_B = \{E, S_c, a, b\}$ denote the attenuation coefficients dependencies of direct-transmission and backward scattered.

2.7 Underwater Image Enhancement

In recent years, the fields of image processing and underwater vision have focused on issues related to the enhancement of underwater images. Due to the complexity of the underwater environment and lighting conditions, improving underwater images presents a significant challenge. Typically, wavelength-dependent absorption and scattering, including forward and backward scattering, degrade underwater images. Additionally, marine snow, which refers to organic particles in the water, introduces noise and further enhances the effects of scattering. These factors result in reduced visibility, decreased contrast, and colour casts, which limit the practical applications of underwater images in fields such as marine biology, archaeology, and marine ecology. Earlier methods tackled this problem by utilizing multiple underwater images or employing polarization filters. However, more recent algorithms focus on enhancing underwater images using information solely from a single image (Schechner & Karpel, 2004)

The exploration of the world's oceans is still limited due to poor visibility conditions. As our planet is predominantly covered by water (about 70% of its surface), there is a growing interest in uncovering what lies beneath the sea. To facilitate this exploration, underwater image enhancement techniques are crucial. These techniques aim to produce clear and high-quality images that enable the monitoring of marine species, underwater landscapes, and underwater flora. Achieving this clarity is essential for gaining insights into the underwater world and understanding its diverse ecosystems (Sahu *et al.*, 2014).

Overall, the development of effective underwater image enhancement methods holds great significance in expanding our knowledge of the underwater environment and its inhabitants. By improving the quality of underwater images, researchers and scientists can explore and analyse underwater scenes with greater accuracy and detail.

2.8 Techniques of Underwater Image Enhancement

In current study, existing techniques for enhancing underwater images are categorised as either traditional or based on deep learning. In traditional approaches, model-based and non-model methods are included. Non-model enhancement techniques are able to improve the visual effect of an image by modifying the pixel values without taking into account the imaging principle. On the other hand, model-based enhancement is also known as the image restoration method which the process of image enhancement is based on imaging model by estimating transmission and perform inverse calculation to obtain the original image.

The rapid progress of deep learning technology has led to significant advancements in underwater image enhancement. The use of deep learning algorithms for enhancing underwater images is rapidly evolving and showing promising developments. Convolution neural networks (CNN) and generative adversarial networks (GAN) are two common types of deep learning network used in image enhancement task (Hu *et al.*, 2022).

2.8.1 Traditional Underwater Image Enhancement

The traditional image enhancement methods encounter limitations when directly applied to underwater image enhancement due to the distinct optical conditions present underwater. These methods typically target specific issues like dehazing, colour correction, brightness and contrast individually but lacking the ability to address multiple challenges simultaneously. To overcome these limitations, non-physical model enhancement algorithms have been proposed. These include histogram-based methods, retinex-based method and fusion-based method. Histogram-based methods utilize statistical analysis of image pixel intensities to adjust the contrast and improve overall image quality. Retinex-based methods focus on restoring the image's illumination and reflectance components, which can greatly enhance visibility and details in underwater scenes. Fusion-based methods involve combining multiple images or different processing techniques to produce a final enhanced image that addresses various issues.

In contrary, physical model-based methods aim to simulate the physical processes of light propagation and absorption in water, which are crucial for underwater image enhancement. These methods rely on an underwater imaging model, and one widely used model is the Jaffe-McGlamery underwater imaging model, as mentioned earlier. Once the underwater imaging model is established, the next step involves obtaining the unknown parameters in the imaging model. This can be achieved by leveraging prior knowledge and employing other techniques. Subsequently, the degraded images within the established model are solved to obtain the enhanced images. There are several mainstream methods employed within the physical model-based approach. One approach involves image restoration based on light polarization, where the polarization properties of light are utilized to recover the true underwater scene. Another approach utilizes prior information of the underwater environment and image statistics to restore the degraded images. These methods use statistical regularities and assumptions about the underwater environment to enhance image visual quality. Additionally, integral imaging-based methods are utilized, which capture multiple perspectives of the underwater scene and reconstruct a high-resolution image by fusing these perspectives (Hu *et al.*, 2022).

In conclusion, non-physical model enhancement methods provide flexibility, real-time processing, and potential generalization, but may lack accuracy and robustness in extreme conditions and rely on training data. Physical model-based enhancement methods offer accurate representations and robustness, but can be computationally complex and require parameter tuning. The choice of these methods depends on the requirements and goal of the underwater imaging task.

2.8.2 Deep Learning-Based Underwater Image Enhancement

CNN and GAN are the most popular types of deep learning networks for underwater image enhancement tasks. Traditional model-based methods in underwater image enhancement require estimating the transmission graph and parameters of the underwater image based on prior knowledge and other strategies. However, these estimated values often lack adaptability. To address this, deep learning combined with the physical model leverages the powerful feature extraction capabilities of CNNs to solve parameter values, such as the transmission mapping in the imaging model. In this approach, CNN replaces the assumptions or prior knowledge used in traditional methods, such as the dark channel prior theory. Besides, the enhanced underwater image also can directly output after convolution, pooling, deconvolution, and other operations by input the raw image into the CNN network model with its learning ability. This eliminates the constraints imposed by model assumptions or prior conditions and enables direct learning of the mapping relationship between the original underwater image and the clear underwater image (Hu *et al.*, 2022).

Goodfellow *et al.* (2014) introduced Generative Adversarial Networks (GANs), which use a game-like learning framework with a generator and a discriminator to improve the quality of generated outputs. The generator aims to produce images that closely resemble the real images, making it challenging for the discriminator to distinguish between the real and generated images whereas the discriminator is responsible for determining whether an image is genuine or generated. Through an iterative process, the generator learns to produce increasingly realistic images as it tries to deceive the discriminator. This process involves feeding low-quality images into the generator and obtaining high-quality generated images as output. The discriminator evaluates the generated images along with actual samples and provides a probability value indicating the likelihood of an image being real. GANs have proven to be highly effective in various applications, including image generation, image enhancement, restoration, and style transfer.

CNNs are mainly used for supervised learning tasks as it trained to learn the mapping between the input underwater images and their corresponding enhanced versions. Alternately, GANs are trained in an unsupervised manner with a generator and a discriminator and do not require paired of input-output data. The choice between CNN and GAN depends on the availability of corresponding training data and the desired level of generative and discriminative capabilities for underwater image enhancement. However, CNNs are more computationally efficient compared to GAN. The training and inference processes for CNNs are typically faster, making then suitable for real-time applications, where quick processing is crucial.

2.9 Underwater Haze Removal by Deep Learning

In the field of underwater image enhancement, the foremost challenge that demands attention is haze removal which plays a pivotal role in restoring clarity and visibility to underwater scenes. Thus, significant contributions have been made by various researchers using deep Convolutional Neural Network (CNN) architectures. Cai *et al.* (2016) introduced DehazeNet, a pioneering work focused on underwater dehazing. DehazeNet uses a CNN to estimate the transmission map, which is subsequently used to restore haze-free images using an atmospheric scattering model. The architecture of DehazeNet is specifically designed to incorporate established assumptions and priors in image dehazing.

Addressing the colour distortion and visibility issues prevalent in underwater images due to light absorption and scattering, Wang et al. (2017) proposed a comprehensive underwater enhancement scheme. They utilized a UIE-net, a CNN-based model, for both colour correction and haze removal. Perez *et al.* (2017) also employed a CNN to dehaze underwater images in their enhancement scheme. Similarly, Pan *et al.* (2018) utilized a CNN algorithm to estimate and refine the transmission map incorporated with adaptive bilateral filter in their underwater image dehazing scheme.

Another approach to underwater image dehazing was introduced by Yang and Sun (2018) They proposed a proximal dehaze-net, utilizing CNNs to learn proximal operators. Li *et al.* (2019) tackled the computational complexities of deep learning architectures by developing the U45 dataset, a publicly available underwater test dataset. They addressed colour casts, low contrast, and haze-like effects of underwater degradation by developing a fused adversarial network for enhancing underwater images. Furthermore, they introduced a multi-term loss function for effective colour corrections, resulting in visually pleasing enhanced results.

Many developed techniques based on physical imaging models are often limited in their applicability, being specific to certain images and lacking generalizability. These methods incorporate simplified versions of the image formation model, resulting in insignificant image enhancement effects. To address this, Chen *et al.* (2021) proposed an innovative approach for enhancing underwater images by improving an existing deep learning-based method. They achieved this by combining a revised image formation model with a convolutional neural network (CNN). The CNN incorporates rectified linear unit activation functions and dilated convolutions to enhance the network's fitting ability. The outcome is a dehazed image that preserves the details of the underwater scene. However, the method still lacks of accuracy in terms of colour correction.

2.10 Underwater Colour Restoration by Deep Learning

Following haze removal, another aspect needs to be focused in underwater image enhancement is colour restoration. Overcoming colour distortion caused by light absorption and scattering is paramount to accurately representing the vibrant colours of the underwater environment. According to the findings Li *et al.* (2017), a technique was introduced to address colour correction in underwater images using the CycleGAN framework. In a separate study, Katherine *et al.* (2019 devised a two-stage neural network architecture to tackle image depth estimation and colour correction as distinct processes. Furthermore, Fu and Cao (2020) presented a neural network that effectively merged global and local information to enhance underwater image. Liu *et al.* (2019) developed a deep residual network combined with CycleGAN was utilized to create artificial underwater images. These synthetic images were then employed to train a Convolutional Neural Network (CNN) model, specifically emphasizing improvements in colour correction and resolution enhancement.

Addressing the challenge of colour restoration, Lu *et al.* (2018) employed a deep CNN along with depth estimation in a light field imaging approach. Hu *et al.* (2018) proposed a transmission estimation network (T-network) and a global ambient light estimation network (A-network) for underwater image enhancement, emphasizing colour correction and addressing halo artifacts. Yu *et al.* (2018) introduced a conditional generative adversarial network for underwater image restoration, incorporating a gradient penalty term and perceptual loss to improve visual quality through colour restoration and overall enhancement.

To tackle diverse underwater conditions, Uplavikar *et al.* (2019) developed an enhancement technique that learned domain-agnostic features, achieving enhanced underwater images even with varying viscosities. Li *et al.* (2019) proposed an image enhancement scheme based on a CNN model utilizing underwater scene priors, with a primary focus on colour restoration by directly reconstructing latent underwater images.

Another approach for colour restoration involves the use of white balance methods, which are capable of removing colour casts caused by scene illumination. In the field of deep learning, Afifi and Brown (2020) proposed a deep neural network that is trained end-to-end and uses a CNN architecture to correct white balance. The network maps an input image to two extra white balance parameters that match to indoor and outdoor light and obtained the output with accurate colour correction. This method holds potential for application in underwater colour restoration by training the model with a dataset comprising white balance settings corresponding to underwater illumination.

2.11 Architecture of Convolutional Neural Network

The convolutional neural network (CNN) is a hierarchical architecture that creates a funnel-shaped network followed by a fully connected layer. For a good CNN model, the training time should be as low as possible on the same time have high accuracy and better performance. A CNN model consists of types of layers that connected to one another and incorporate with other components such as activation function, optimization function and loss function (Teuwen and Moriakov, 2020).

2.11.1 Type of Layers

Layers in CNN are classified into three types which are convolutional layers, pooling layers, and fully linked layers. Each layer has a distinct purpose in image processing. Figure 2.7 demonstrates that convolutional and pooling primarily serve the purpose of extracting features to learn the pattern from the input while the fully connected layer is used for classification towards corresponding output. In image enhancement task, the main focus lies solely on extracting features, details and mapping from the original images and enhance it, rather than on classification. Consequently, deep learning-based image enhancement model typically incorporate only convolutional and pooling layers during their construction.

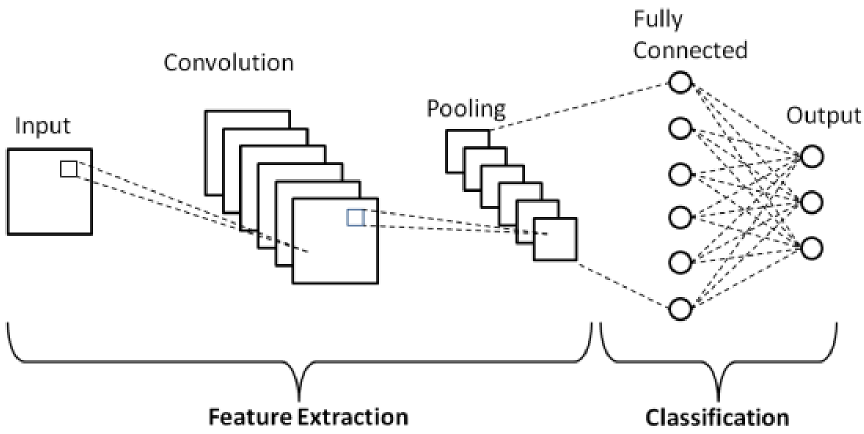


Figure 2.7 Type of Layers in CNN Architecture (Source: https://www.upgrad.com/blog/basic-cnn-architecture/)

A. Convolutional Layer

Convolutional layer is the initial layer or the input layer of the CNN. It consists of multiple filters or kernels that convolve across the input image, capturing the spatial patterns and correlations. These layers may be stacked to learn hierarchical representation of the input.

B. Pooling Layer

Pooling layer's primary function is to extract image features and decrease the feature maps acquired from the convolutional layer while preserving image details. A smaller input size helps to minimise CNN computation and prevent overfitting. Max pooling and average pooling are common pooling operations. Max pooling selects the maximum value from the pooling kernels, whereas average pooling computes the mean value (Chen, 2019).

C. Fully Connected Layer

Fully connected layers establish connections between every neuron in the previous layer and every neuron in the subsequent layer. They help in capturing high-level abstractions and making final predictions. Generally, these layers are positioned towards the end of the network architecture.

2.11.2 Activation Function

Every layer in the neural network has the responsibility of selecting a specific activation function for its neurons. The activation function uses a straightforward mathematical operation to compute the output based on the input of each neuron. The purpose of the activation function is to introduce nonlinearity into the neural network, enabling it to effectively address complex problems. Rectified linear unit (ReLU) function, sigmoid function, hyperbolic tangent (Tanh) function, exponential linear unit (ELU) function, and parametric rectified linear unit (PReLU) function are typical activation functions used in CNN. The choose of the activation function is depending on the prediction problem to solve (Chen, 2019).

2.11.3 Optimization Function

When training a neural network, it is required to use an optimization technique to update the neural network parameters, such as weights and bias in each layer of the network, in order to minimise the value of the loss function. The gradient descent function is the one of the common optimization functions. However, it has several evident flaws: it is simple to get the local optimal value but difficult to reach the global optimal value. As a result, many optimization methods are updated using the gradient descent algorithm.

An article presents an optimization technique dubbed "Adam." The Adam optimization algorithm is a variant of the stochastic gradient descent technique. It is now frequently utilised in deep learning applications, particularly in computer vision and natural language processing. The Adam algorithm differs from the typical stochastic gradient descent approach. Stochastic gradient descent uses a single learning rate (alpha) to update all weights, and the learning rate does not change during training. Adam computes an independent adaptive learning rate for various parameters by calculating the gradient's first- and second-moment estimations. Overall, Adam is a good optimization function for the deep learning model.

2.11.4 Loss Function

Loss function is the one of the most important components of neural network, as it along with the optimization function are directly responsible for the fitting the model to the given data. A loss function quantifies the difference between the network's expected output and the actual output (ground truth). During the training process, the CNN aims to minimize this loss to improve its performance and accuracy. CNN utilises several loss functions, including mean squared error (MSE), binary cross-entropy, and categorical cross-entropy. The selection of the loss function is depending upon the specific task at hand, the objective of the challenge, and the desired behaviour of the CNN.

2.12 Underwater Image Dataset

Underwater image dataset is a sample of underwater images use to train deep learning model and test the algorithm performance. Due to the underwater environment, it is a difficult task for researcher to get real underwater image as the dataset for research and training purpose. Besides, different datasets are available for various applications and settings, catering to specific targets or serving general tasks such as reconstruction or colour correction. These datasets vary widely in their characteristics and are designed to address specific needs within their respective domains.

For example, TURBID, focuses on testing and evaluating enhancement and restoration methods. It consists of 82 underwater images captured in a human-made artificial underwater environment. The TURBID dataset provides a standardized benchmark to assess the performance of algorithms under specific conditions (Duarte *et al.*, 2016).

In addition to TURBID, several other datasets contribute to the advancement of underwater image processing. For testing enhancement and restoration techniques, the Synthetic Underwater Image Dataset (SUID) provides 900 synthetic underwater images and 30 ground truth images (Hou *et al.*, 2020).

Furthermore, the enhancing underwater visual perception (EUVP) dataset is presented by Islam *et al.* (2020) which is a large-scale dataset of a paired and an unpaired collection of underwater images that are captured using seven different cameras over various visibility conditions during oceanic explorations and human-robot collaborative experiments.

However, synthetic images do not provide the same level of fidelity and realism as real-world images when it comes to training the deep learning models. Thus, an underwater image enhancement benchmark dataset (UIEB) has been proposed. It comprises 950 real-world images and 890 of which have the corresponding references images for benchmarking. Rest 60 underwater images without reference image are challenging data to be evaluated. This dataset has been used to train an underwater image enhancement network called WaterNet as a benchmarking of training CNN for underwater image enhancement (Gonzalez-Sabbagh & Robles-Kelly, 2023). A list of reviewed datasets is tabulated in Table 2.1.

Table 2.1 List of Underwater Image Datasets.

Dataset	Description	Underwater Data Included
TURBID (Duarte *et al.*, 2016)	Testing and evaluating underwater image enhancement and restoration methods	82 underwater images with ground truth
SUID (Hou *et al.*, 2020)	Evaluating enhancement and restoration methods	900 synthetic underwater images and 30 ground truth images
UIEB (Li *et al.*, 2020)	Underwater image enhancement and restoration	950 real-world underwater images, 860 of these with reference images and 60 challenging images
EUVP (Islam *et al.,*	Large-scale dataset of a paired and an unpaired collection of	2185 training pairs and 130 validation images

2020)	underwater images	

2.13 Evaluation Metrics

In underwater image enhancement, the type of metrics validation plays a crucial role. A suitable quality evaluation metric indicates the quality of images.

2.13.1 Mean Square Error

Mean square error (MSE) is the most common evaluation metric to measure image quality. A traditional full reference metric can be obtained by calculating the cumulative squared error between the resultant enhanced image and the reference image. The lower the MSE value, the better the quality of the images (Sara *et al.*, 2019). MSE is denoted as,

$$MSE = \frac{1}{MN}\sum_{i=0}^{M}\sum_{j=0}^{N}\left(I_{en}(i,j) - I_{ref}(i,j)\right)^2 \tag{2.14}$$

where:

M, N are the numbers of rows and columns in the input images,

i, j are the coordinates of a pixel in the input images,

i_{en} is the enhanced image and

I_{ref} is the reference image.

2.13.2 Peak Signal-to-Noise Ratio

Peak signal-to-noise ratio (PSNR) is another often applied metric in image processing to quantify the quality of reconstructed images relative to original or reference images. It computes the maximum signal power to noise power ratio and is defined by the mean square error. PSNR is expressed in decibels (dB) and the higher the values indicate the better image quality. PSNR is defined as,

$$PSNR = 20\log_{10}\left(\frac{MAX}{\sqrt{MSE}}\right) \tag{2.15}$$

where:

MAX is the maximum possible pixel intensity value and

MSE is the mean square error.

2.13.3 Structural Similarity Index Measure

Structural similarity index measure is a method that measures the similarity between two images by considering the change of perceptions in structural information of the images. SSIM is determined using the luminance, contrast, and structure of an image. The resultant similarity value is in decimal form and ranges between -1 and 1, with 1 indicating perfect similarity, 0 indicating no similarity and -1 indicating perfect anti-correlation. SSIM can be expressed as

$$SSIM(x,y) = l(x,y)\cdot c(x,y)\cdot s(x,y) \tag{2.16}$$

where:

$l(x,y)$ is the luminance similarity,

$c(x,y)$ is the contrast similarity and

$s(x,y)$ is the structural similarity.

Luminance similarity is used to compare the brightness of two images, contrast similarity is used to compare the difference in brightness between brightest and darkest region of two images, and structure similarity is used to compare the local structure patches and pattern of two images. These three components are defined as,

$$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \tag{2.17}$$

$$Sc(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \tag{2.18}$$

$$s(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \tag{2.19}$$

where:

μ_x, μ_y are the local means of image patches x and y,

σ_x, σ_y are the standard deviations of pixels in patch x and y,

σ_{xy} is the convariance of patched x and y, and

C_1, C_2, C_3 are small constants to stabilize the division while the denominator is close to zero.

2.13.4 Underwater Image Quality Measure

Underwater Image Quality Measure (UIQM) is a metric for evaluating the overall quality of underwater images in a manner equivalent to the human visual system. UIQM includes three quality attributes, including the underwater image quality colourfulness measure (UICM), the underwater image sharpness measure (UISM), and the underwater image contrast measure (UIConM). As the value of UIQM higher, the quality of the enhanced underwater image better. All the attributes are linearly combined together and UIQM is given by

$$UIQM = c_1 \times UICM + c_2 \times UISM + c_3 \times UIConM \tag{2.20}$$

where:

$UICM$ is the colourfulness measure,

$UISM$ is the sharpness,

$UIConM$ is the contrast, and

c_1, c_2, c_3 are the weighted coefficients.

2.13.5 Underwater Colour Image Quality Evaluation

Underwater Colour Image Quality Evaluation (UCIQE) is a metric designed to access the quality of colour images captured underwater. It aims to measure the colour degradation and distortion in water due to light attenuation, scattering and colour cast. A weighted summation of the standard deviation of chroma, the contrast of luminance and the average saturation is calculated by UCIQE to measure underwater image quality (Yang and Sowmya, 2015). The higher the UCIQE values, better the quality of the underwater image and balance between chroma, contrast and saturation. UCIQE can be denoted as

$$UCIQE = c_1 \times \sigma_{ch} + c_2 \times contrast + c_3 \times \mu_{sat} \tag{2.21}$$

where:

σ_{ch} is the standard deviation of chroma,

$contrast$ is the contrast of luminance,

μ_{sat} is the average saturation, and

c_1, c_2, c_3 are the weighted coefficients.

2.13.6 Patch-Based Contrast Quality Index

Patch-Based Contrast Quality Index (PCQI) is a metric for evaluating the quality of contrast changes in the image by employing an adaptive representation of local patch structure. PCQI consists of three independent components which are mean intensity, signal strength and signal structure used to define any image patch in a novel and adaptable manner. It is calculated by the averaged of the summation of the three components to provide a single score of an entire image. The higher the PCQI score indicates a better quality of contrast in the image (Wang *et al.*, 2015). The formulation for PCQI is shown in Equation 2.22.

$$PCQI = \frac{1}{M} \sum_{j=1}^{M} q_i(x_j, y_j) \cdot q_c(x_j, y_j) \cdot q_s(x_j, y_j) \tag{2.22}$$

where:

M is the total number of patches,

$q_i(x_j, y_j)$ is the mean intensity,

$q_c(x_j, y_j)$ is the signal strength,

$q_s(x_j, y_j)$ is the signal structure.

2.13.7 Entropy

Entropy is a fundamental concept in information theory that measures the amount of uncertainty or randomness in a system. In the context of image processing, entropy is often used as a measure of the amount of information content and variability in pixel values of an image. Entropy evaluation typically applied to 8-bit grayscale images and the value range from 0 to 8. A higher entropy value means that the enhanced image has more detail and variability in the pixels (Tsai *et al.*, 2007). The formulation for Entropy, $\mathrm{H}(S)$ is shown in Equation 2.22.

$$\mathrm{H}(S) = -\sum_{i=1}^{n} p(S_i) \cdot \log_2 p(S_i) \tag{2.23}$$

where:

$p(S_i)$ is the probability of appearance of pixel value i,

2.14 Discussion

Based on the reviews, Underwater image degradation is a result of light attenuation when it enters the water. This attenuation arises from various processes such as reflection, refraction, absorption, and scattering of light, which occur when light strikes the water surface and permeates it. To simulate these underwater conditions, the Jaffe-McGlamery underwater image model was proposed. However, this model only considered the imaging process under ideal medium conditions.

Aiming for a more practical approach, the Narasimhan imaging model was introduced, taking into account additional underwater conditions such as the attenuation coefficient and the distance between the camera and the scene. However, this model didn't take into consideration the three RGB colour channels of images. Consequently, a revised model of image formation was presented, taking into account the optical imaging capabilities of the underwater environment and including the attenuation coefficient across RGB colour channels.

Traditional underwater image enhancement methods performed well in processing specific images for certain tasks, but failed to automate the overall enhancement process. This is due to the unique properties of each image, rendering the parameter settings for one image potentially unusable for others. Additionally, these methods required more time to improve the image quality compared to deep learning-based methods, which can offer a better time-quality ratio for image enhancement. Deep learning methods can automate the image enhancement process as they learn and store models capable of enhancing images with different properties.

In this project, CNN has been selected over GAN to implement the deep learning network. Based on the architecture, CNN use a combination of convolutional and pooling layers that can automatically and adaptively learn spatial hierarchies of features, making them adept at understanding the complex structures within images. It also operates within a supervised learning framework, which allows for more direct control and clear interpretation of the learning process, and can offer more precise and reliable results when there are abundant labelled training data available, as in this project. GANs, while they have shown promise in various applications, can be harder to train due to their game-theoretic nature and may not offer the same level of control and predictability that our project requires.

Among various deep learning image enhancement methods, a technique that combines a revised image formation model with deep learning was chosen. This method primarily focuses on haze removal tasks, while an auto white balance model addresses colour restoration and removal of colour cast.

Numerous underwater datasets have been analysed in research to train deep learning models. This project prioritizes datasets with strong dehazing and colour correction reference images. Additionally, as the proposed method aims to restore colour using the auto white balance approach, datasets yielding good results in white balance are preferred. Among these, the UIEB dataset was found to be the most suitable for training the proposed deep learning model.

Several metric evaluations have been introduced in research, with MSE, PSNR, SSIM, PCQI and entropy being the most popular. However, these metrics only evaluate the overall quality of the images and are not specific to underwater images. Therefore, only SSIM, entropy and PCQI were selected to assess the overall quality including haze removal, contrast and sharpness of the enhanced images. Additionally, UIQM and UCIQE were chosen as they popular for underwater task evaluation. They assess the overall quality of underwater images and colour balance. These metrics will be used as benchmarks in this project.

CHAPTER 3

METHODOLOGY

3.1 Introduction

In this chapter, the proposed method of implementing the convolutional neural network (CNN) algorithm by combining underwater image formation model and white balance model to enhance the quality of underwater colour image will be discussed.

This chapter encompasses the project framework, system architecture, experiment setup and workflow of the proposed method. This project framework is a comprehensive study of this project's structural design, which is divided into two phases, namely Phase I and Phase II. Phase I or also known as the research phase emphasizes reading and understanding the topic of the related materials and the work contributions by researchers in this field. Phase II or also known as the development phase is highlighted on implementation and analysis of the proposed technique studied in this project.

Moreover, the system architecture is the conceptual method designed to achieve the objectives of this project. The development of the system will be described in the system design. There are 2 stages in which the system architecture can be categorized into. These are training stage and testing stage. These stages will be discussed in detail by the guidance of formulae and figures.

3.2 Project Framework

A framework design for a deep learning-based underwater image quality enhancement system is constructed as illustrated in Figure 3.1. Phase I consists of three stages which are Preliminary Research, Literature Review and Framework Design while Phase II comprises three stages which including Implementation, System Evaluation and Conclusion. This framework could insights on the main goal of this project, thus ensuring that this project achieves the objectives that have been declared in chapter 1.

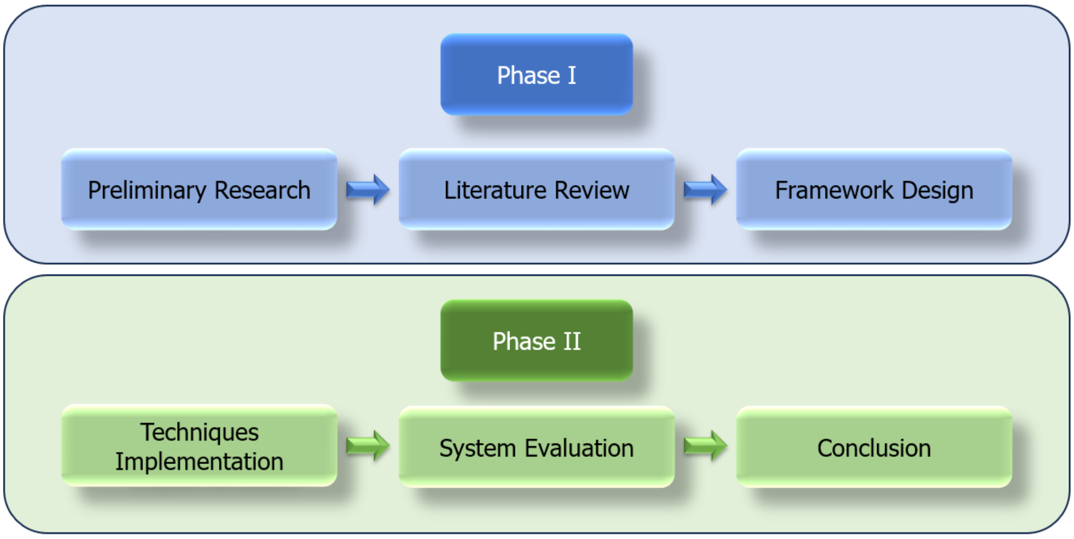


Figure 3.1 The overall framework of this project

3.2.1 Phase I

In phase I, the preliminary research is first conducted in this project. During this stage, the findings of current issues of underwater image enhancement are written in the problem background. Several limitations are identified from the problem background that further delivered as the problem statement. The aim of this project is then determined to encounter the drawbacks declared in the problem statement. It is then followed by achieving the main goal of carrying out this project through objectives that are developed by scopes.

Next, Literature Review will be the second stage in Phase I. This stage demands a bunch of researching, reading and understanding the topic chosen for this project. All the relevant materials that corresponded to underwater image enhancement are being explored. All the materials are obtained from journals, articles, theses, books, conference papers, etc. this is an essential stage that contributed to the initial ideas on how to accomplish the project through the analysis of existing techniques, history, application, technologies, etc.

The last stage of Phase I is the Framework Design that depicts the entire structure of this project in which the existing techniques are chosen and stated out.

3.2.2 Phase II

The initial stage in Phase II is the Technique Implementation. This stage is where the selected techniques are applied to achieve the goals of this project. The chosen methods are implemented by using Visual Studio Code Version 1.79.0 with python 3.10.8 (64-bit) language.

The middle stage of phase II is the system evaluation. Some analysis would be carried out in this stage toward the system performances to identify the problem that occurred. Furthermore, the enhancement of the system would also be performed to obtain a better result. The system is then undergoing a few experiments according the benchmarks.

Lastly, a conclusion of the whole project will be written to summarize all of the important aspects as well as further research. It also includes how the system can contribute to this era of technology and how people interact with this system.

3.3 System Architecture

The system architecture design can be split into two phases: training and testing, and three sections which are the input, process, and output. In training phase, the input are the raw images and references images from UIEB and EUVP dataset. Next, the process method consists of data preparation, image formation model, white balance model and initialize training procedure. The output of the training phase are the pre-trained image formation model and pre-trained white balance model. In testing phase, the input of the proposed system are the test images from UIEB dataset. Then, the methods comprised of tensor conversion, image formation model (which comprised of backward scatter estimation module, direct-transmission module and reformation calculation), auto white balance model (which formed by an encoder and decoder with auto white balance setting), image conversion and colour mapping that executed in sequence. The image formation model is to reconstruct the degraded underwater image and white balance (WB) model will responsible for restore and balance the colour of images. Lastly, enhanced underwater images with original resolution are produced as the final output of the system. Figure 3.2 depicts the system architecture design of deep learning-based underwater image enhancement system. Figure 3.3 and 3.4 illustrates the system architecture design of image formation model and white balance model.

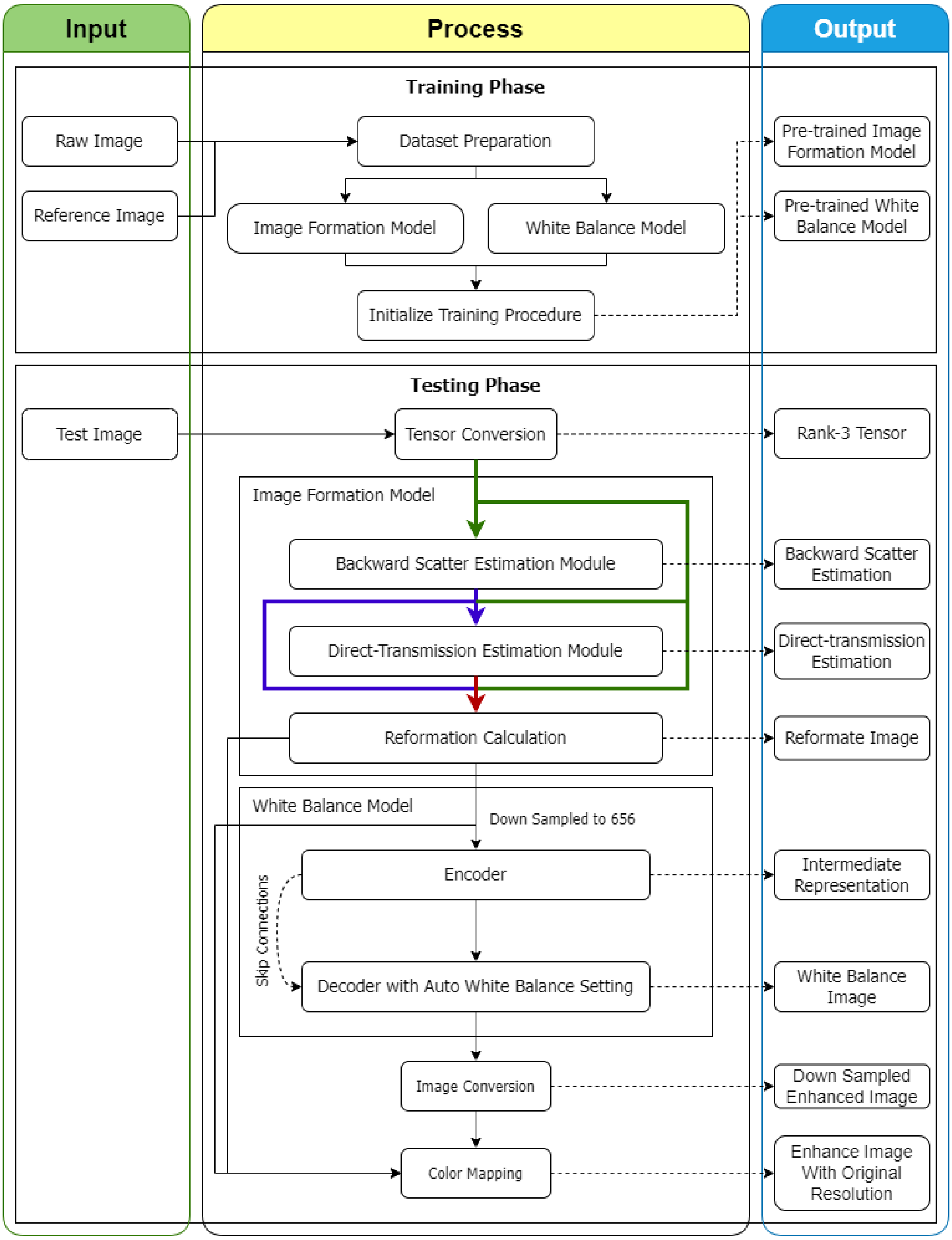


Figure 3.2 System Architecture design of deep learning-based underwater image enhancement system

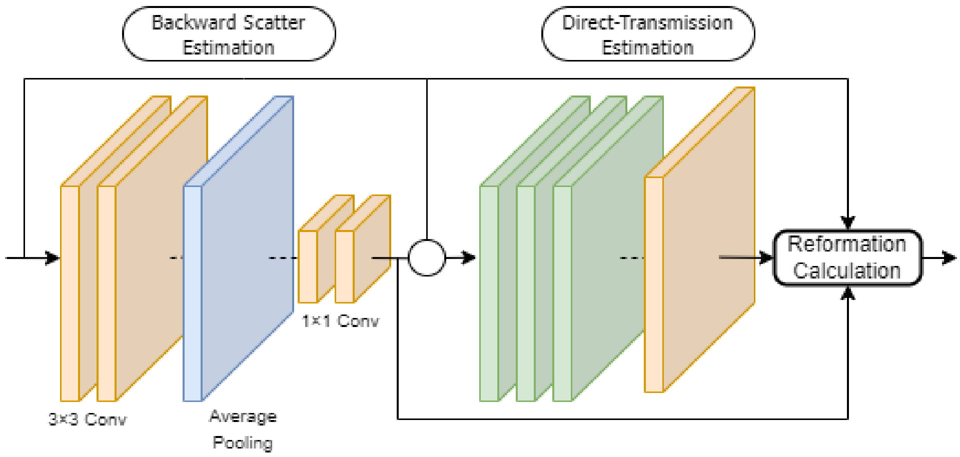


Figure 3.3 System Architecture design of Image Formation Model

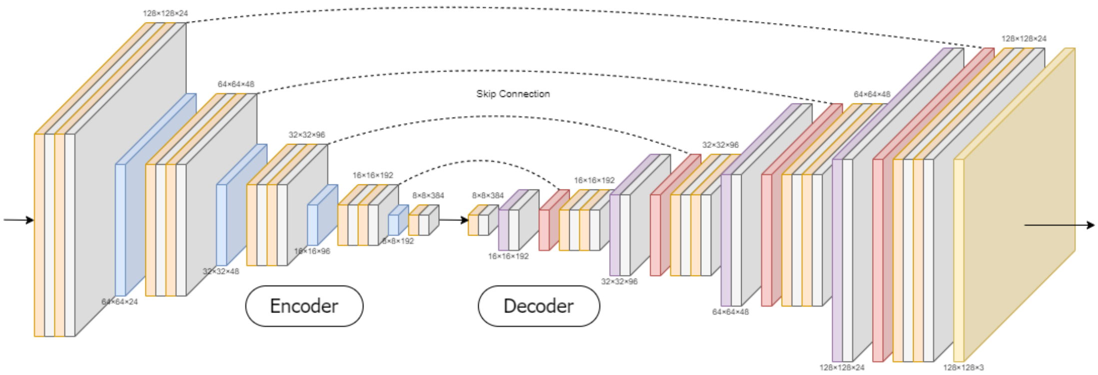


Figure 3.4 System Architecture design White Balan Model

3.4 Training Phase

In the training process of the image formation model, each raw image is paired with a corresponding reference image, which serves as the target for the training. Prior to inputting them into the deep learning model, these raw images undergo data preparation to form pairs with their respective target images. Once prepared, the training procedure begins by initializing the neural network parameters and utilizing the Adam optimizer for optimization.

The objective during training is to minimize the discrepancy between the enhanced image generated by the model and the target image. This discrepancy in training image formation model is quantified using the mean square error (MSE) loss function, as expressed in equation (3.1). The MSE loss function calculates the average squared difference between the pixel values of the enhanced and target images, providing a measure of how well the model is performing in terms of pixel-level accuracy.

$$MSE = \frac{1}{MN}\sum_{i=0}^{M}\sum_{j=0}^{N}\left(I_{en}(i,j) - I_{ref}(i,j)\right)^2 \tag{3.1}$$

Similarly, for the white balance model, the training process employs the Adam optimizer to optimize the neural network parameters. However, in contrast to the image formation model, the loss function utilized here is the mean absolute error (MAE) based on pixel values, as depicted in equation (3.2). The MAE loss function computes the average absolute difference between the pixel values of the enhanced image and the target image. This choice of loss function is suitable for white balance adjustment tasks as it focuses on the magnitude of errors without being sensitive to the direction of deviations.

$$MAE = \frac{1}{MN}\sum_{i=0}^{M}\sum_{j=0}^{N}\left|I_{en}(i,j) - I_{ref}(i,j)\right| \tag{3.1}$$

Upon completion of the training phase, the outputs consist of pre-trained models for both the image formation and white balance models. These pre-trained models are then utilized in the testing phase to evaluate their performance on new, unseen data, allowing for the assessment of how well they generalize to real-world scenarios. Overall, the training procedure ensures that the deep learning models learn to accurately generate enhanced images and perform white balance adjustments by minimizing the discrepancies between their outputs and the corresponding target images.

3.5 Testing Phase

3.5.1 Input

The input of the proposed system will be the digitized underwater colour images that retrieved from online dataset and classified in train and test images. The underwater image enhancement benchmark (UIEB) dataset provides 950 real-world underwater images, 890 of which have the corresponding target images that can be used as train images and the rest 60 underwater images as test images (Li *et al.*, 2019). The python imaging library (PIL) will be applied to retrieve the image pixels in the form of matrix arrays.

3.5.2 Process

Convolutional neutral network will be implemented in the system to enhance underwater image quality in terms of dehazing and colour restoration. The deep learning algorithm is designed based on the formation of underwater image and auto white balance technique.

A. Tensor Conversion

The conversion of image to tensor is necessary for the image before entering CNN model because CNN are designed to process and analyse tensor data. Tensors are multi-dimensional arrays that can represent various type of data, including images. By converting an image to tensor, the compatibility of the data format with the CNN model is being ensured. The most common format for representing images as tensors is the channel-last format, also known as the NCHW format where N represents the number of images in a batch, C represents the number of channels or colour channels, H and W represents the height and width of the image respectively. This data format is commonly used in some frameworks like Pytorch.

Algorithm 1 : Tensor conversion

IN : Image I

OUT : Tensor T^I

Set up tensor transformation
for I in original image directory **do**
    Load I using pillow package
    Apply tensor transformation on I
    Unsqueeze T^I to add another dimension at the front to get NCHW format
endfor

7 **return** tensor T^I

Figure 3.5 Algorithm of Tensor Conversion in the system

B. Backward Scatter Estimation Module

After obtaining the tensor of the input image, it will be passed to convolutional neutral networks that designed for formation of underwater image undergo deep learning training. The deep learning model is designed to fit multiple components in the revised image formation model. Let recall back the equation 2.13 of revised model

$$L_c(x) = J_c(x)e^{-\beta_C^D(V_D)d} + L_c^\infty\left(1 - e^{-\beta_C^B(V_B)d}\right) \tag{3.1}$$

where:

$L_c(x)$ is the total light intensity received by camera

$J_c(x)$ is the scene with true intensity value,

$e^{-\beta_C^D(V_D)d}$ and $e^{-\beta_C^B(V_B)d}$ is the direct transmission mapping,

L_c^∞ is the backward scattered light

In order to compute the true intensity value of the scene, the value of direct transmission mapping and backward scattered light is needed. The first part of image formation model is the estimate the backward scattered light from the input image, that is, L_c^∞ in Equation 3.1. The backward scatter estimation module includes two groups of convolution layer with 3×3 kernel size, one adaptive average pooling layer and another two group of convolution layer with 1×1 kernel size. All the convolution layer designed with 3 input channels and 3 output channels. For the activation function of the convolution operation, the parametric rectified linear unit (PReLU) is chosen. The detailed structure is shown in Figure 3.4.

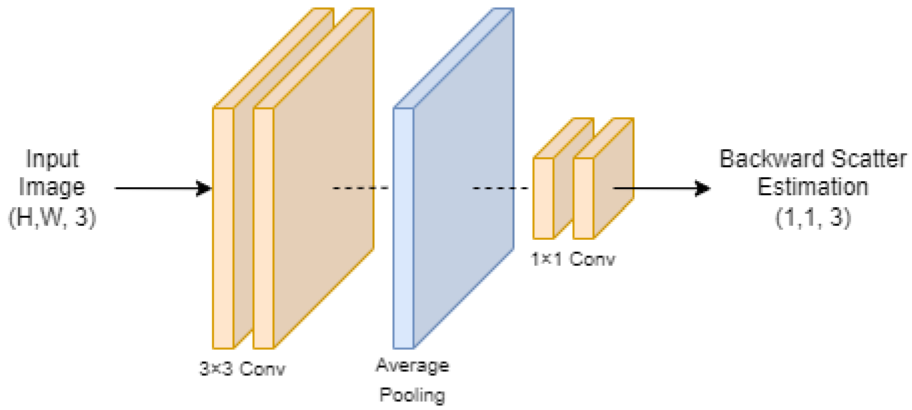


Figure 3.6 Detailed structure of Backward Scatter Estimation Module

---

Algorithm 2 : Backward Scatter Estimation Module

IN : Tensor T^I

OUT : Backward Scatter Estimation B

Initialize the backward scatter estimation module

Define the architecture of the module

- Apply the first convolutional layer with 3 input channels, 3 output channels and a kernel size of 3×3 and perform padding to maintain input size
- Apply the PReLU activation function
- Apply the second convolutional layer with 3 input channels, 3 output channels and a kernel size of 3×3 and perform padding to maintain input size
- Apply the PReLU activation function
- Perform adaptive average pooling to reduce the spatial dimensions of the tensor to 1×1
- Apply the third convolutional layer with 3 input channels, 3 output channels and a kernel size of 1×1
- Apply the PReLU activation function
- Apply the fourth convolutional layer with 3 input channels, 3 output channels and a kernel size of 1×1
- Apply the PReLU activation function

Pass the input tensor T^I through the module layers in a forward direction

return Backward Scatter Estimation B

Figure 3.7 Algorithm of Backward Scatter Estimation Module in the system

C. Direct-Transmission Estimation Module

The second part of image formation model is to estimate the direct transmission mapping from the input image, that is, $e^{-\beta_C^D(V_D)d}$ in Equation 3.1. The direct-transmission estimation module includes three groups of dilated convolutional layers with 3×3 kernel size and one group of normal convolutional layer with same kernel size. The first dilated convolutional layer designed with 6 input channels and 8 output channels and another two dilated convolutional layers designed with 8 input and output channel. The last layer consists of 8 inputs channels and 3 output channels. The concatenation of backscatter estimation with the input image will then pass into this module to compute the direct transmission mapping. The detail structure is shown in Figure 3.6.

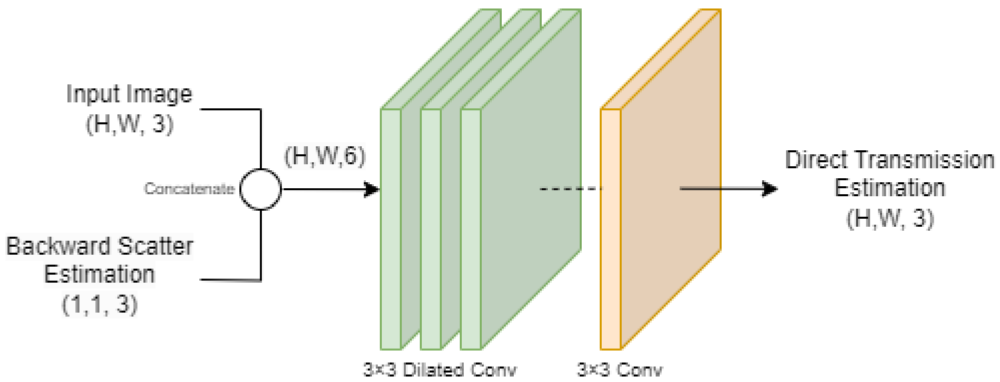


Figure 3.8 Detailed structure of Direct-Transmission Estimation Module

---

Algorithm 3 : Direct-Transmission Estimation Module

IN : Tensor T^I
Backward Scatter Estimation B

OUT : Direct-Transmission Estimation D

---

1 Initialize the direct-transmission estimation module

2 Define the architecture of the module

3 Apply the first convolutional layer with 6 input channels, 8 output channels, a kernel size of 3×3 and dilation of 1 and perform padding to maintain input size

4 Apply the PReLU activation function

5 Apply the second convolutional layer with 8 input channels, 8 output channels, a kernel size of 3×3 and dilation of 2 and perform padding to accommodate the dilation rate

6 Apply the PReLU activation function

7 Apply the third convolutional layer with 8 input channels, 8 output channels, a kernel size of 3×3 and dilation of 5 and perform padding to

accommodate the dilation rate

Apply the PReLU activation function

Apply the fourth convolutional layer with 8 input channels, 3 output channels and a kernel size of 3×3and

Apply the PReLU activation function

$\text{Input} \leftarrow \text{Concatenate}(T^I, B)$

Pass the input through the module layers in a forward direction

return direct-transmission estimation D

Figure 3.9 Algorithm of Direct-Transmission Estimation Module in the system

D. Reformation Calculation

To obtain the true intensity value of original scene, rearrange the equation 3.1 subject to $J_c(x)$.

$$J_c(x) = (L_c(x) - L_c^{\infty})e^{\beta_C^D d} + L_c^{\infty} e^{(\beta_C^D - \beta_C^B)d} \tag{3.2}$$

According to the research from Akkaynak & Treibitz (2018), the difference between β_C^D and β_C^B is very small and approximated as 0 at distances greater than 3m. Thus, equation 3.2 can be rewrite in the form of equation 3.3.

$$J_c(x) = (L_c(x) - L_c^{\infty})e^{\beta_C^D d} + L_c^{\infty} \tag{3.3}$$

where:

$c \in \{R, G, B\}$ denotes the three colour channels of red, green and blue,

β_C^D denotes the attenuation coefficients for direct-transmission light,

β_C^B denotes the attenuation coefficients for backward scattered light,

d denotes the transmission distance between scene and camera,

$V_D = \{d, \rho, E, S_c, a, b\}$ and $V_B = \{E, S_c, a, b\}$ denote the attenuation coefficients dependencies of direct-transmission and backward scattered.

The proposed method in the project using equation and combined with deep learning techniques to reformat the degraded underwater image to increase the overall quality of the image including dehazing.

Algorithm 4 : Reformation Calculation

IN : Tensor T^I
Backward Scatter Estimation B
Direct-Transmission Estimation D
OUT : Tensor of dehaze image T^{DH}

1 Initialize the reformation calculation module
2 $T^F \leftarrow ((T^I - T^B) \times T^D + T^B)$
3 Apply clamping to the T^F to ensuring that the pixel values are within the range of 0 to 1
4 **return** tensor of dehaze image T^{DH}

Figure 3.10 Algorithm of Reformation Calculation in the system

E. Encoder

After obtaining the reformation image in tensor form, it will further pass to auto white balance model to remove the colour cast of the scene's illumination. In auto white balance model consists of an encoder and decoder to produce an output image with auto white balance. The model is trained with raw images and references images with corrected by white balance setting. For the encoder part, it can be described as a reconstruction function to reverses the camera-rendered underwater colour image back to its corresponding raw colour image with the current WB setting applied. The output of this encoder is an intermediate representation. The encoder framework is illustrated in Figure 3.11.

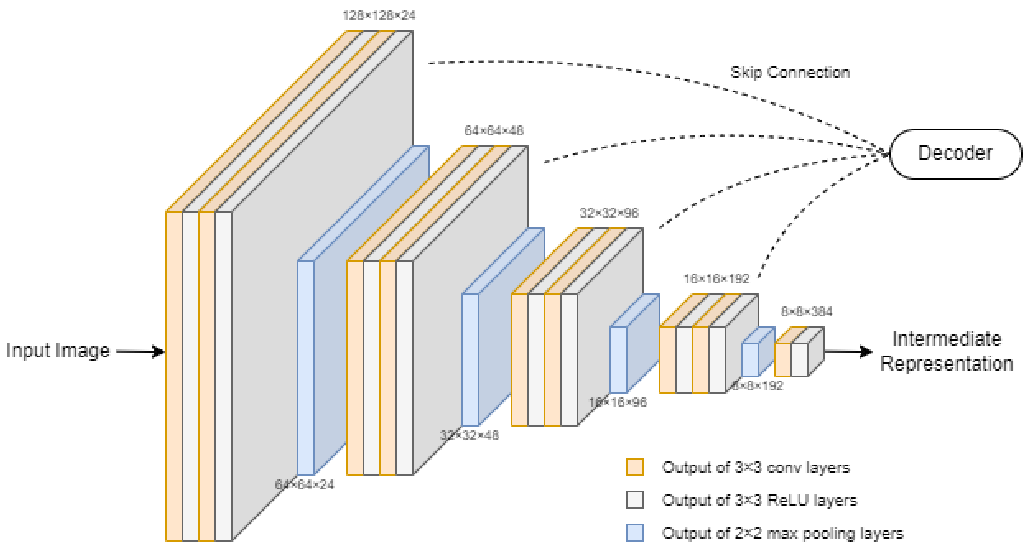


Figure 3.11 Framework of Encoder

Algorithm 5 : Encoder

IN : Tensor of dehaze image T^{DH}

OUT : Intermediate representation

Initialize the double convolutional block (DoubleConvBlock) module

Define the architecture of the DoubleConvBlock module

    Apply the first convolutional layer with accepts n_i input channels, n_o output channels, a kernel size of 3×3 and perform padding to maintain input size

    Apply the ReLU activation function with inplace is true

    Apply the second convolutional layer with accepts n_o input channels, n_o output channels, a kernel size of 3×3 and perform padding to maintain input size

    Apply the ReLU activation function with inplace is true

Initialize the downscale block (DownBlock) module

Define the architecture of the DownBlock module

    Perform max pooling with kernel size of 2 to reduce the spatial dimensions of the tensor by half

    Apply the DoubleConvBlock module

Initialize the downscale bottleneck block (DownBottleBlock) module

Define the architecture of the DownBottleBlock module

    Perform max pooling with kernel size of 2 to reduce the spatial dimensions of the tensor by half

    Apply the convolutional layer with accepts n_i input channels, n_o output channels, a kernel size of 3×3 and perform padding to maintain input size

    Apply the ReLU activation function with inplace is true

Initialize the encoder module

Define the architecture of the encoder module

    Apply the DoubleConvBlock module with $n_i \leftarrow 3$ and $n_o \leftarrow 24$

    Apply the DownBlock module with $n_i \leftarrow 24$ and $n_o \leftarrow 48$

    Apply the DownBlock module with $n_i \leftarrow 48$ and $n_o \leftarrow 96$

    Apply the DownBlock module with $n_i \leftarrow 96$ and $n_o \leftarrow 192$

    Apply the DownBottleBlock module with $n_i \leftarrow 192$ and $n_o \leftarrow 384$

Define the forward function of the encoder module

    $x \leftarrow \mathrm{DoubleConvBlock}(T^F)$

    $x_1 \leftarrow \mathrm{DownBlock}(x)$

    $x_2 \leftarrow \mathrm{DownBlock}(x_1)$

$x_3 \leftarrow \mathrm{DownBlock}(x_2)$

$x_5 \leftarrow \mathrm{DownBlock}(x_4)$

Pass T^F through the encoder module layers in a forward direction

return Intermediate representation

Figure 3.12 Algorithm of Encoder in the system

F. Decoder

The designed white balance model consists of multi-scale skip connections between the encoder and decoder which to transfer the intermediate representations from encoder to decoder. Once getting the intermediate representation from encoder, decoder scheme will act as an unknown camera rendering function that accepts it and renders it with the target auto WB setting to a RGB colour space encoding. The output of this decoder is the final underwater images with auto WB setting. The decoder framework is illustrated in Figure 3.13.

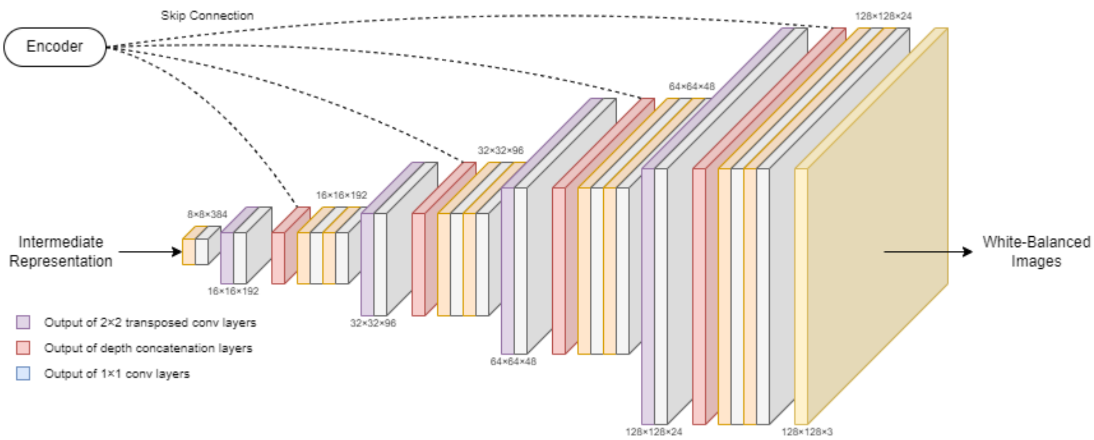


Figure 3.13 Framework of Decoder

Algorithm 5 : Decoder

IN : Intermediate representation x_5
Intermediate representation x_4
Intermediate representation x_3
Intermediate representation x_2
Intermediate representation x_1

OUT : Tensor with auto white balance T^{AWB}

Initialize the upscale bottleneck block (UpBottleBlock) module

Define the architecture of the UpBottleBlock module

  Apply the convolutional layer with accepts n_i input channels, n_i output channels, a kernel size of 3×3 and perform padding to maintain input size

  Apply the ReLU activation function with inplace is true

  Apply the transposed convolutional layer with accepts n_i input channels, n_o output channels, a kernel size of 2×2 and a stride of 2

Initialize the upscale block (UpBlock) module

Define the architecture of the UpBlock module

  Apply the DoubleConvBlock module from encoder

  Apply the transposed convolutional layer with accepts n_i input channels, n_o output channels, a kernel size of 2×2 and a stride of 2

Define the forward function of the UpBlock module

  $x \leftarrow \text{Concatenates}(x_2, x_1)$

  $x \leftarrow \text{DoubleConvBlock}(x)$

Initialize the output block (OutputBlock) module

Define the architecture of the OutputBlock module

  Apply the DoubleConvBlock module from encoder

  Apply the convolutional layer with accepts n_i input channels, n_o output channels, a kernel size of 1×1

Define the forward function of the OutputBlock module

  $x \leftarrow \text{Concatenates}(x_2, x_1)$

Initialize the decoder module

Define the architecture of the decoder module

  Apply the UpBottleBlock module with $n_i \leftarrow 384$ and $n_o \leftarrow 192$

  Apply the UpBlock module with $n_i \leftarrow 192$ and $n_o \leftarrow 96$

  Apply the UpBlock module with $n_i \leftarrow 96$ and $n_o \leftarrow 48$

  Apply the UpBlock module with $n_i \leftarrow 48$ and $n_o \leftarrow 24$

  Apply the OutputBlock module with $n_i \leftarrow 24$ and $n_o \leftarrow 3$

Define the forward function of the decoder module

  $x_{awb} \leftarrow \text{DoubleConvBlock}(x_5)$

  $x_{awb} \leftarrow \text{DownBlock}(x_{awb}, x_4)$

  $x_{awb} \leftarrow \text{DownBlock}(x_{awb}, x_3)$

  $x_{awb} \leftarrow \text{DownBlock}(x_{awb}, x_2)$

  $T^{AWB} \leftarrow \text{DownBlock}(x_{awb}, x_1)$

Pass x_5 through the encoder module layers in a forward direction

33 **return** tensor with auto white balance T^{AWB}

Figure 3.14 Algorithm of Decoder in the system

G. Image Conversion

After obtain the output tensor with AWB, we need to convert it back to image form in order to show it. It is typically the reverse process of tensor conversion.

Algorithm 6 : Image Conversion

IN : Tensor T^{AWB}

OUT : Down Sampled Enhanced Image

1 Set up image transformation

2 Squeeze T^{AWB} to remove another dimension at the front to get CHW format

3 Apply image transformation on T^{AWB}

4 **return** Down Sampled Enhanced Image

Figure 3.15 Algorithm of Image Conversion in the system

H. Colour Mapping

To ensure a consistent run time for any size of input images, all input images have been down sampled to a specific size. The deep learning model is applied on the down sampled images to produce down sampled enhanced image. In order to produce the enhanced image with original resolution, global colour mapping function between the down sampled input and down sampled output is applied to original input. The equation of colour mapping is shown below.

$$I^e = M\psi\left(I^i\right) \tag{3.3}$$

where:

I^e is enhanced image with original resolution,

M is the mapping function between the down sampled input and down sampled output,

ψ is a polynomial kernel function that maps the image's RGB vectors to a higher 11-dimensional space.

Algorithm 7	: Colour Mapping
IN	: Input Image I Down sampled dehaze image $I_{\downarrow}^{DH}$ Down sampled auto white balance image $I_{\downarrow}^{AWB}$
OUT	: Enhanced Image with original resolution I^{e}
1	Define the polynomial kernel function ψ
2	Get mapping function between $I_{\downarrow}^{DH}$ and $I_{\downarrow}^{AWB}$
3	$M \leftarrow polynomial\ fitting(\psi(I_{\downarrow}^{DH}), I_{\downarrow}^{AWB})$
4	Apply mapping function m to I
5	$I^{e} \leftarrow M\psi(I)$
6	**return** enhanced image with original resolution I^{e}

Figure 3.16 Algorithm of Colour Mapping in the system

3.5.3 Output

The tensor conversion is applied to RGB channel test image I and train image I return a rank-3 tensor T^{I}. Then, the tensor is passed into image formation model to undergo back scatter and direct transmission module to estimate backward scatter estimation B and direct-transmission estimation D and reformation calculation to compute the dehaze image in tensor form T^{DH}. This is then followed by the encoder which produce an intermediate representation and decoder in auto white balance model to generate the output with auto WB setting T^{AWB}. The tensor with auto WB setting is then converted to image form. However, the output now is been down sampled. Finally, the output is used to compute the colour mapping function and applied to original image to obtain the enhanced image with original resolution I^{e}. To conclude, the final output of this system generates an enhanced underwater image with dehazing and colour restoration result from CNN deep learning algorithms. In addition, the final output is colour image consists of 3 colour channels (RGB).

3.6 Flow Chart

A flow chart is illustrated to describe the algorithm of deep learning-based underwater image enhancement in the proposed system. As depicted in Figure 3.17, the flowchart of proposed system starts from loading the input image, go through a series of image processing techniques and CNN models that have been detailed in section 3.5.2 and produce an enhanced image as final output in the end.

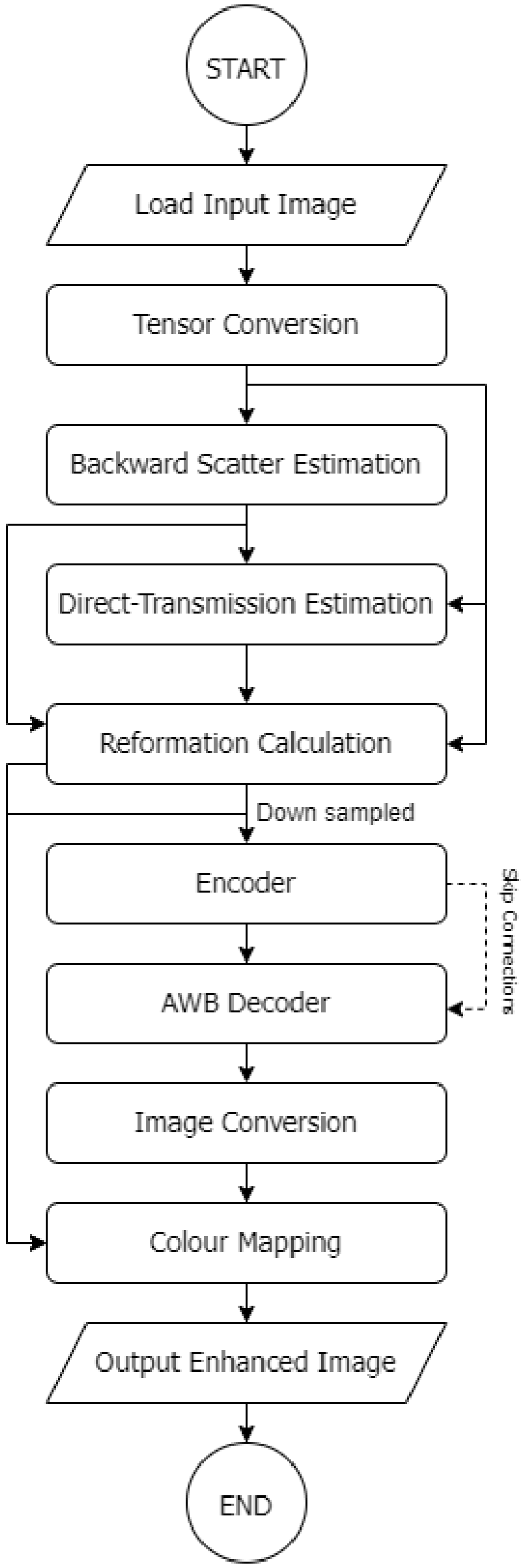


Figure 3.17 Flowchart of the proposed system

3.7 Experimental Setup

There are a few materials that will be utilized for the experimental setup to conduct the proposed system. These include a laptop, coding software with python language and underwater image dataset as the main apparatus and materials for this project. The details of the setup will be shown in Table 3.1 below.

Table 3.1 List of apparatus and material used for experiment setup with their properties and applications.

Materials	Unit	Properties	Function
Laptop	1	• Model: HP Pavilion Laptop 15ec0xxx • Processor: AMD Ryzen 7 4800H with Radeon Graphics 2.90 GHz • RAM: 16.00GB • System type: 64-bit operating system, x64-based processor	An electronic digital device to store images and medium to conduct the experiment with python and integrated software
Visual Studio Code	1	• Version: 1.80.0 • System type: 64-bit • Integrated Development Environment (IDE) for python	Software to perform and integrate computation, visualization and provide programming environment
Online Dataset	2	• Underwater Image Enhancement Benchmark (UIEB) dataset consists of 890 training pair images and 60 challenge images • Enhancing Underwater Visual Perception (EUVP) dataset consists of 2185 training pair images and 130 validation images for underwater scenes	Consist of real-world underwater images and corresponding high-quality reference images that will be adopted as experiment subjects for training and testing.

3.8 Evaluation Metrics

To validate the experimental result conducted in this project, parameters were chosen for evaluation purposes. The properties of these benchmarking are tabulated in Table 3.2.

Table 3.2 List of parameters to conduct benchmarking upon result of proposed system.

Parameters	Description	Benchmark
Structural Similarity Index Measure (SSIM)	SSIM defines the measurement of the luminance, contrast and structure of two images and compute the similarity value between the images. The SSIM value is in decimal form and lies between -1 and 1, with 1 indicating perfect similarity, 0 is no similarity and -1 is perfect anti-correlation. The lower the value means enhanced image consists significant difference from the original image. $SSIM(x, y) = l(x, y) \cdot c(x, y) \cdot s(x, y)$	Similarity of original image and enhance image
Entropy	Entropy defines measure of the amount of information content and variability in pixel values of an image. The value is range from 0 to 8. A higher entropy value means that the enhanced image has more detail and variability in the pixels. $\mathrm{H}(S) = -\sum_{i=1}^{n} p(S_i) \cdot \log_2 p(S_i)$	Information and variability in image
Underwater Image Quality Measure (UIQM)	UIQM defines the measurement that combined multiple quality attributes such as underwater image quality colourfulness measure (UICM), the underwater image sharpness measure (UISM) and the underwater image contrast measure (UIConM) to evaluate the overall quality of underwater image. As the value of UIQM higher, the overall quality of the underwater image better. $UIQM = c_1 \times UICM + c_2 \times UISM + c_3 \times UIConM$	Quality of sharpness and contrast in underwater image

Underwater Colour Image Quality Evaluation (UCIQE)	UCIQE defines the measurement of the colour degradation and distortion in water due to light attenuation, scattering and colour cast. It calculates the weighted summation of the standard deviation of chroma, the contrast of luminance and the average saturation of the underwater images. The higher UCIQE value indicates the better quality of the enhanced underwater colour image. $$UCIQE = c_1 \times \sigma_{ch} + c_2 \times contrast + c_3 \times \mu_{sat}$$	Quality of colour balance in the underwater image
Patch-Based Contrast Quality Index (PCQI)	PCQI defines the measurement of quality of contrast changes in the image by employing an adaptive representation of local patch structure. It calculates the average of the summation of the mean intensity, signal strength and signal structure to provide a single score of an entire image. The higher the PCQI score indicates a better quality of contrast in the image. $$PCQI = \frac{1}{M}\sum_{j=1}^{M} q_i(x_j, y_j) \cdot q_c(x_j, y_j) \cdot q_s(x_j, y_j)$$	Quality of contrast in the image

3.4 Summary

In this chapter, the project framework of this research is thoroughly described in two phases. In the research phase, numerous of study and research needs to be done to interpret and collect all the relevant information regarding the underwater image enhancement. The techniques that have been explored in research phase will be potentially chosen as the algorithm to build and execute the proposed system during development phase. The system architecture of the proposed system is divided into three major parts which are input, process and output. The main deep learning-based underwater image enhancement algorithms will be included in the process part. The project is able to achieve with the help of the list of experimental setups. Lastly, the evaluation of the selected techniques and the result of the whole system will be benchmarked by SSIM, Entropy, UIQM, UCIQE and PCQI.

CHAPTER 4

SYSTEM DESIGN AND IMPLEMENTATION

4.1 Overview

This chapter encompasses the design and implementation of the proposed system in which the general work process of the system framework and the algorithm of the deep learning based underwater image quality enhancement system will be presented. All the algorithms are implemented by utilizing the Python programming language through the IDE of Visual Studio Code (Version 1.80.0). There are also import of external libraries that assist the accomplishment of the proposed algorithm, which include OpenCV, PIL, PyTorch, NumPy, PyQT5 and Matplotlib.

4.2 Use Case Diagram

Based on the illustration of the use case diagram in Figure 4.1, there is only one primary actor which is the user that interacts with the proposed system. Firstly, the user can load a digital image or a folder of image through open the file explorer into the system. The input image will display as well as update the details of the image. Then, the image is ready for enhance using the trained models. The output image will be displayed and its benchmark value will then be computed and display. Besides, in order to see the difference more significantly, the display image can be zoomed and panned by clicking at the navigator of the image. For the pre-trained model, user able to change and import another model by clicking the tool button to open file dialog. Furthermore, to view the difference in benchmark value of input and output image, the compare benchmark button is built to display the comparison and also the visual of both images. Lastly, the user can save a copy of the enhance underwater image in device.

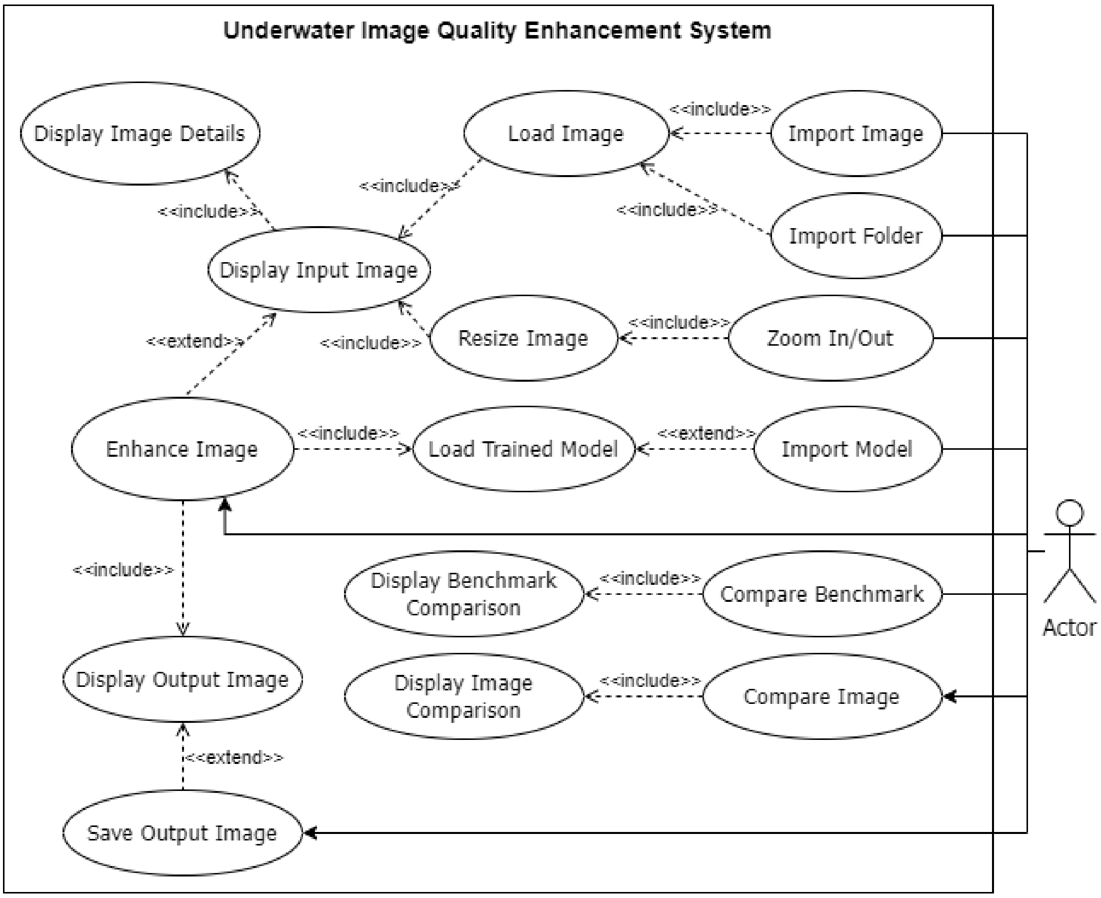


Figure 4.1 Use Case Diagram of Underwater Image Quality Enhancement System

4.2.1 Use Case Description

According to the use case diagram modelled above, most use cases are further described in Table 4.1 – 4.8.

Table 4.1 Use case description for Load Image

Use Case	Description
Actor	User
Pre-condition	Digital image in PNG, JPEG, JPG, TIF and TIFF format
Basic Flow of Events	1. User click the New button under file in menu bar. 2. Execute the file explorer. 3. Select an image.

	4. The image is load and read. 5. Input image is displayed on the GUI.
Post-condition	Input image is displayed

Table 4.1 shows the description of use case diagram for Load Image, which is the one of the initial actions triggered by actor to import test image into the system.

Table 4.2 Use case description for Load Folder

Use Case	**Description**
Actor	User
Pre-condition	Folder which contains digital image in PNG, JPEG, JPG, TIF and TIFF format
Basic Flow of Events	1. User click the New button under folder in menu bar. 2. Execute the file explorer. 3. Select a folder. 4. The images in the folder are load and read. 5. The list of images is displayed on the GUI
Post-condition	Folder of input image is displayed

Table 4.2 shows the description of use case diagram for Load Folder, which is another initial action triggered by actor to import a list of test images into the system.

Table 4.3 Use case description for Zoom In/Out

Use Case	**Description**
Actor	User
Pre-condition	Input image is imported and displayed
Basic Flow of Events	1. User click on + button for zoom in and – button for zoom out or drag the zoom slider. 2. The image is resized and display. 3. User can drag the white rectangle on navigator to pan and view different part of the image.
Post-condition	The resized image is displayed

Table 4.3 shows the description of use case diagram for Zoom In/Out, that allow user to have a closer and clearer vision on the displayed image with functions of panning and zooming.

Table 4.4 Use case description for Import Model

Use Case	**Description**
Actor	User
Pre-condition	The pre-trained model is same with the model for enhance image.
Basic Flow of Events	1. User click on the tool button. 2. Execute the file explorer. 3. Select the pre-trained model. 4. The model is load. 5. The path of model is displayed in a label.
Post-condition	The model is loaded and displayed.

Table 4.4 shows the description of use case diagram for Import Model, that allow user change and import pre-trained model that will be used for enhancing.

Table 4.5 Use case description for Enhance Image

Use Case	**Description**
Actor	User
Pre-condition	The input image and model are imported and displayed.
Basic Flow of Events	1. User press on the Enhance button. 2. The imported model is load. 3. The input image is converted into tensor. 4. The input tensor is then pass to Image Formation Model and undergo: ✓ Backward Scatter Estimation ✓ Direct-Transmission Estimation ✓ Reformation Calculation 5. The output tensor of Image Formation Model is down-sampled and pass to White-Balance Model as input and undergo:

	✓ Encoder ✓ Decoder 6. The output tensor is converted back into image. Perform colour mapping on the down-sampled image and the original resolution image. 7. The output image is displayed.
Post-condition	The output enhanced image is displayed

Table 4.5 shows the description of use case diagram for Enhance Image, in which the deep learning technique is applied to enhance the input image.

Table 4.6 Use case description for Compare Benchmark

Use Case	**Description**
Actor	User
Pre-condition	Output image is enhanced and displayed
Basic Flow of Events	1. User click on the Compare Benchmark button. 2. A new window is displayed. 3. The benchmark value of input and output image is computed and displayed. 4. The good and bad value is indicated with arrow and colour.
Post-condition	All benchmark value is updated

Table 4.6 shows the description of use case diagram for Compare Benchmark, where the benchmark of system is computed and display in a new window for ease of comparison.

Table 4.7 Use case description for Compare Image

Use Case	**Description**
Actor	User
Pre-condition	Output image is enhanced and displayed
Basic Flow of Events	1. User click on the double frame button. 2. The frame of display image will separate into two:

	one for input image and one for output image. 3. The input and output image are displayed within two separate frames.
Post-condition	The input and output image are displayed together

Table 4.7 shows the description of use case diagram for Compare Image, where the display image frame is divided into two to display input and output image for ease of visual comparison.

Table 4.8 Use case description for Save Output Image

Use Case	**Description**
Actor	Actor
Pre-condition	Output image is enhanced and displayed
Basic Flow of Events	1. User click on Save button. 2. Execute file explorer. 3. Enter filename and choose the compression format. 4. The enhanced image is saved in the selected path.
Post-condition	The output is saves within the selected path

Table 4.8 shows the description of use case diagram for Save Output Image, which is an optional function provided by system to keep a copy of output.

4.3 Class Diagram

Figure 4.4 shows the UML Class Diagram of the Underwater Image Quality Enhancement System.

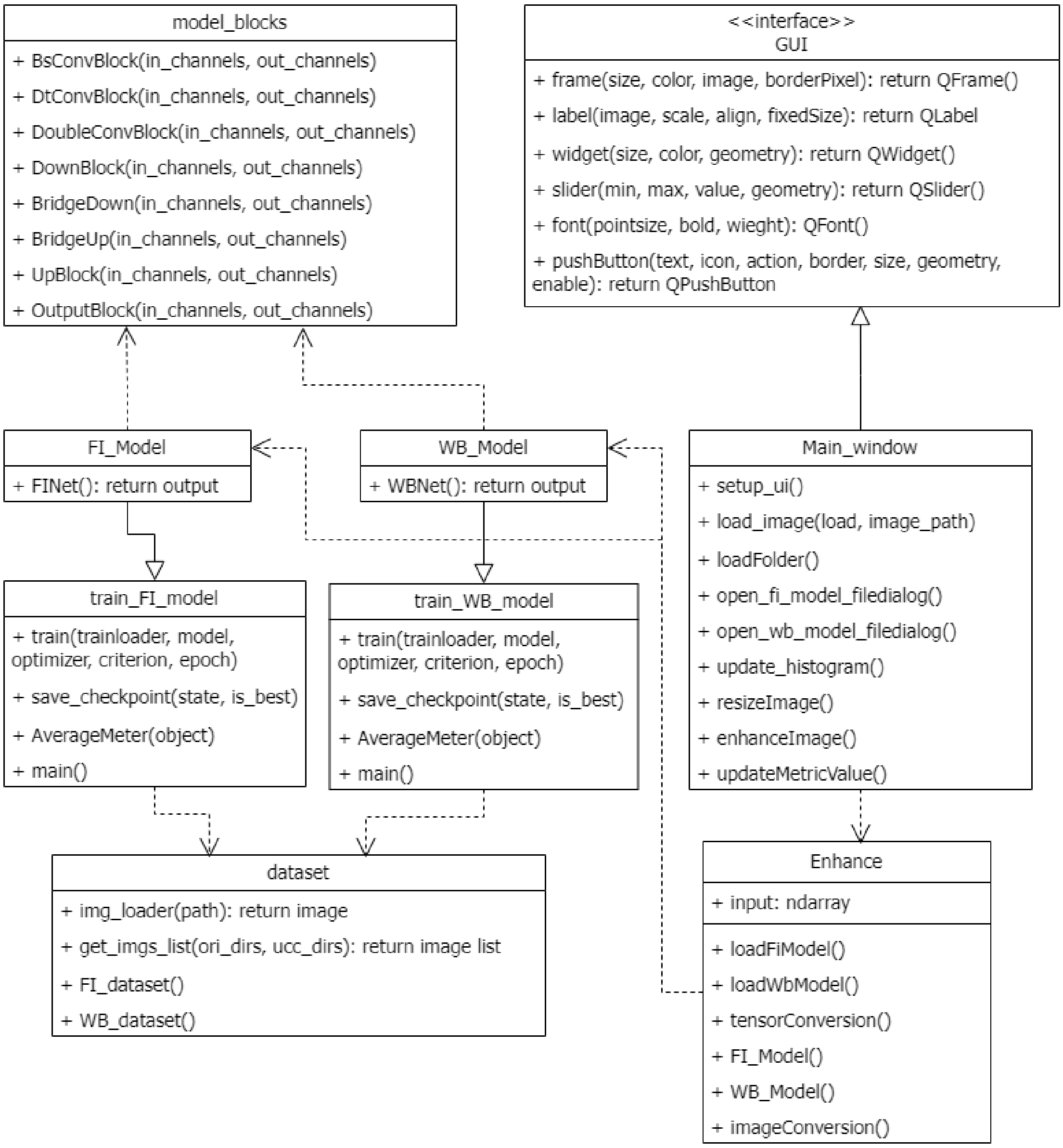


Figure 4.4 UML Class Diagram of Underwater Image Quality Enhancement System

In this project, the classes can categories into two part which is for training and testing. The model_blocks, FI_model, WB_model, train_FI_model, train_WB_model and dataset classes are used for training the deep learning models. The model_blocks is the class for FI_model and WB_model to build their layers in convolutional neural network. The dataset class is used to load the training dataset for training the two models while the process of training is under the class train_FI_model and train_WB_model. For testing part, Main_window is the main class that compose the deep learning technique and integrate actions with GUI, which will be executed to run the proposed system. GUI is the class inherited from

Main_window that is responsible to build the user interface that allow user interaction and visualize the output. Last but not least, enhance is class for processing the algorithm of deep learning to enhance underwater image quality.

4.4 Activity Diagram

Figure 4.3 show the activity diagram of the Underwater Image Quality Enhancement System. Firstly, the user needs to click on either the load image button or load folder button from the GUI menu bar. This action will open the system file explorer for user to choose the image file and obtain the path of the image file. The system will read the image and display it in GUI as well as the details of the image. After that, user can enhance the image with the default pre-trained model by presses the enhance button or user can import other pre-trained models to enhance by presses the tool button. The path of the model will be displayed in GUI. After press the enhance button, the image will go through a series of process with two pre-trained models and the final output will be displayed in GUI. At the same time, the metric evaluation for the benchmark will be updated simultaneously. Once image imported or enhanced, user can choose to zoom in/out the image by dragging the slider of clicking the button under the navigator frame. Furthermore, user also able to view the comparison of metric evaluation between original image and enhanced image by clicking the compare benchmark button. For visual appearance comparison, user can click on the change view button to separate the display image frame into two. Last but not least, save button is user to save the final output image of underwater image quality enhancement.

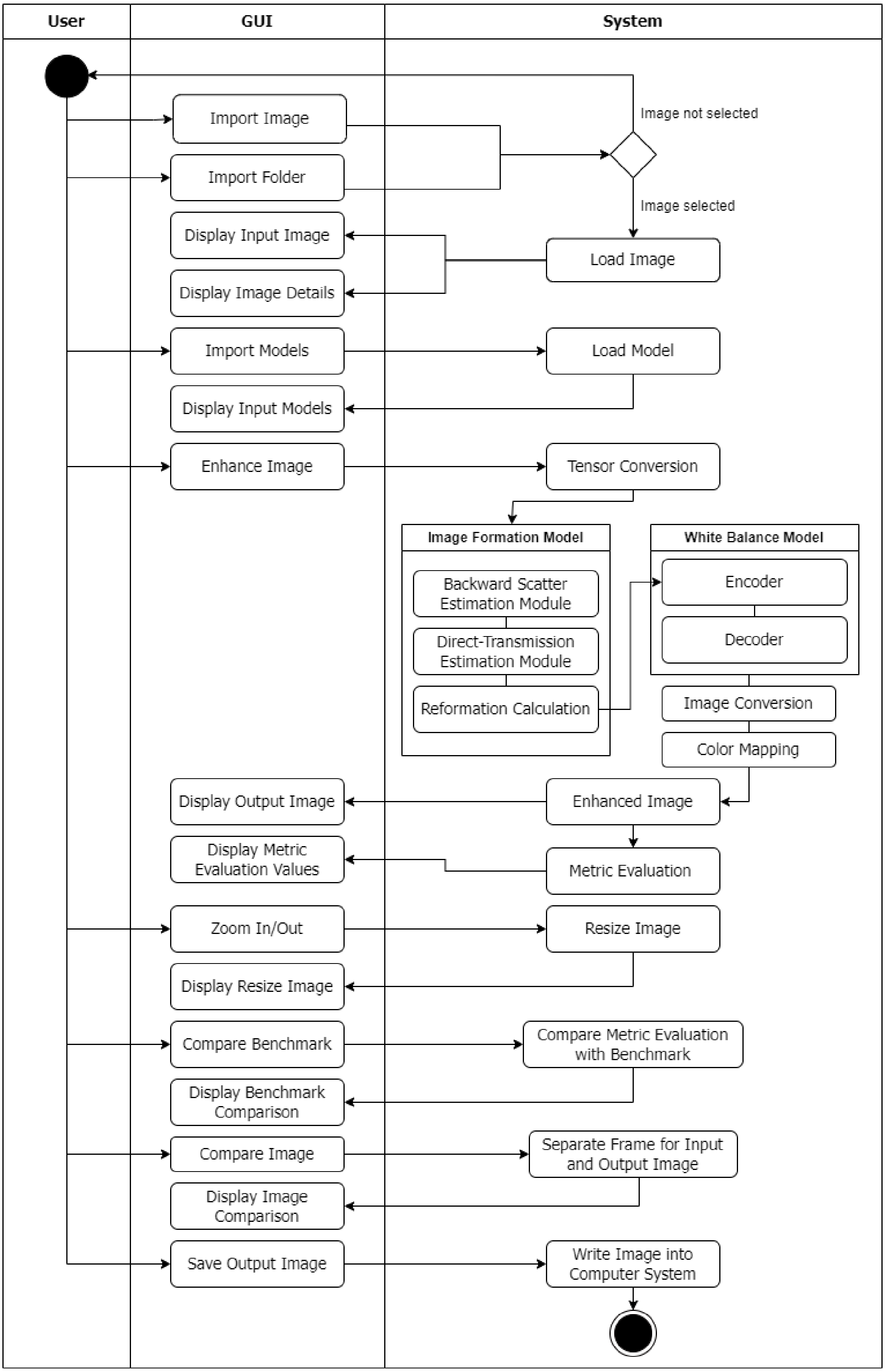


Figure 4.3 Activity diagram of Underwater Image Quality Enhancement System

4.5 State Machine Diagram

Figure 4.2 show the state machine diagram of the Underwater Image Quality Enhancement System. Firstly, the system will be in idle state and waiting for an image to be input. During input an image, the image is validated to ensure that it is in the correct format. If image is invalid, system will back to idle state. Next, if the image is imported successfully, the system is ready to enhance the input image with proposed method. Once the process of enhancement is done, the system will display the output and metric and overall comparison is able to compute. A new window will pop out for user to view the comparison. Finally, the enhanced image is able to save by user into device. Thus, these are all the states in the proposed system.

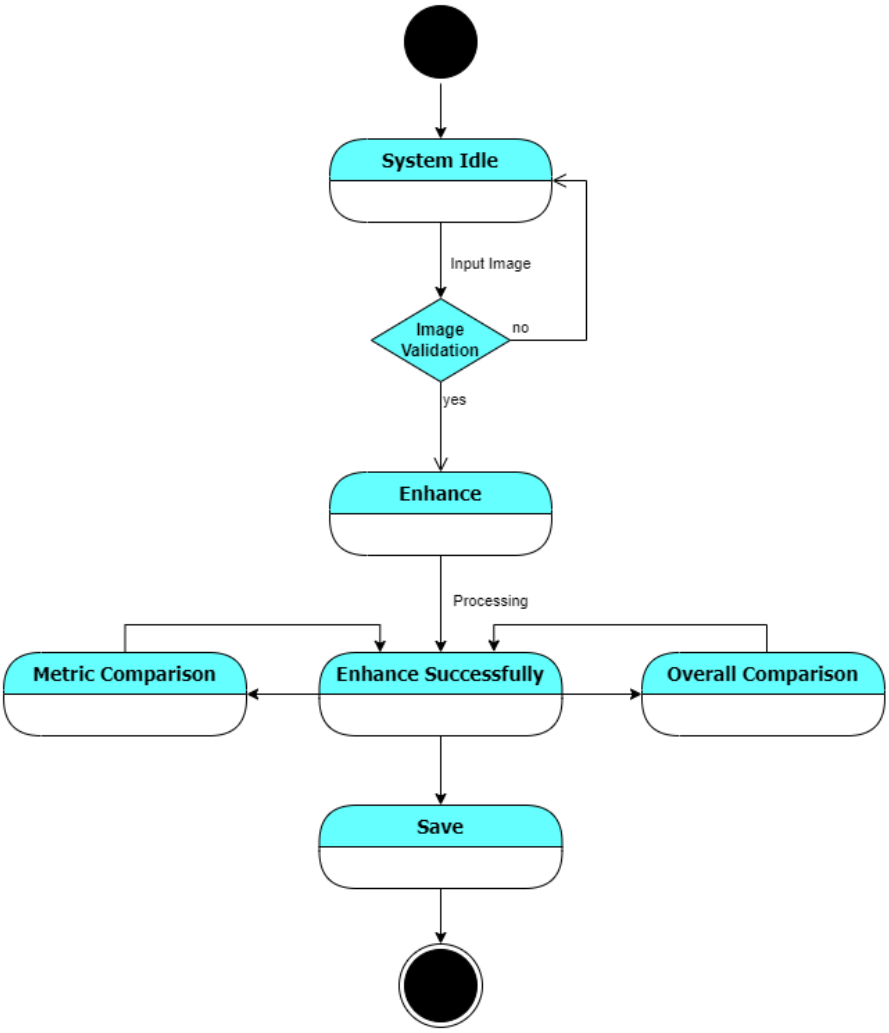


Figure 4.2 State machine diagram of Underwater Image Quality Enhancement System

4.6 Experiment on Training Model

In order to verify the effectiveness of the underwater image quality enhancement method proposed in this project, a series of experiment of training the CNN models are conducted in Visual Studio Code using PyTorch deep learning framework. The computer used has AMD Ryzen 7 4800H processor with Radeon Graphics 2.90 GHz.

In order to train image formation model, the batch size used in the deep learning training algorithm is 1, the learning rate is 0.001, number of workers is 1 and the number of epochs is 3000. On the other hand, the parameter used in training white balance model is 16 of batch size, 0.0001 of learning rate, 4 number of workers and the number of epochs is 3000.

The source of the images in the deep learning training experiments is the UIEB dataset and EUVP dataset, which collects a large number of underwater images taken in real scenes. Both of the dataset also contains corresponding colour-enhanced images obtained by a variety of existing algorithm and manual evaluation of volunteers. The details of the both datasets have been discussed in previous section.

4.7 Graphical User Interface (GUI)

Graphical User Interface (GUI) of the system serves as a platform to allow user interaction with the proposed system. Thus, a simple and user-friendly interface system has been implemented. The application begins with a start page that has a start button which navigates to the main window of the system.

Figure 4.5 shows the Start page of the GUI, in which the basic information is displayed. User will need to click the 'START' button in order to continue to the main page of the system. After user click the button, a loading state will show in the Start page as displayed in Figure 4.6. After the loading is done, the Start page will close automatically and the main page of the system will display.

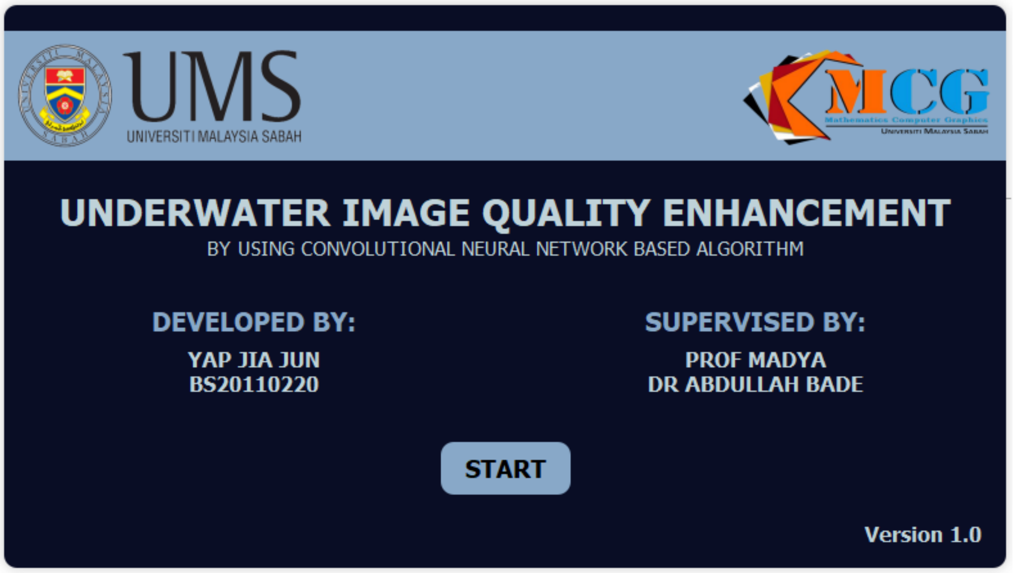


Figure 4.5 GUI Start Page

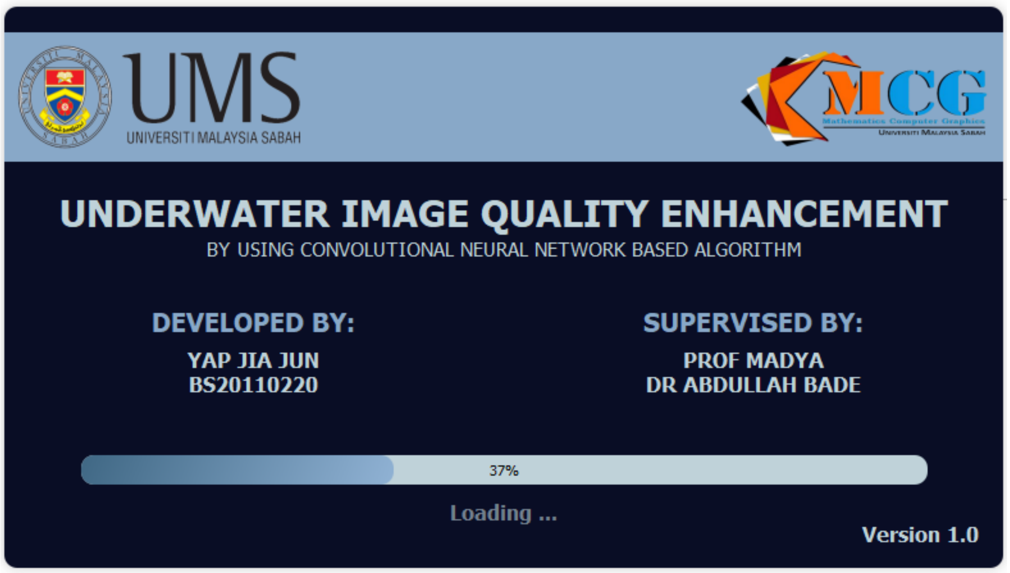


Figure 4.6 GUI Start Page – Loading State

With the reference to depicted Figure 4.7, the Main Page has the same theme colour as Start page and design with few sections. First of all, the top section is the menu bar which consists of File, Folder, Edit and Help. Below it is the system's logo and name. Next, move on to the main section which is in the middle part where it is the place to display the input and output image. Whenever user input image or enhance the image, the image will be displayed in the same part. It consists of two modes which are single display mode or double display mode. Single display mode will only show the after image while double display mode will show the original and after enhanced image on the same time for comparison. Then, the part below it is where the metric evaluation value of the image will be shown together with the dimension of the image. Furthermore, the left part consists of navigator, histogram and button for changing the display mode. Navigator is where user able to zoom in or out by using button or slider and pan the image to view any part of the image. Histogram is the part to display the histogram of the image with three colour channels. Lastly, the right section comprises of basic function, import model, processing status and enhance button. There are three basic functions which are overall comparison (to display all the output for comparison), metric comparison (comparing the metric evaluation of original image and enhanced image) and reset (reset everything into initial state). Besides, user able to use the default pre-trained model or import new pre-trained models to use for enhance the image. During the process of enhancement, the status of each step will be display in the processing status section to tell user. The most important button is the enhance button which is for user to press to enhance the input image.

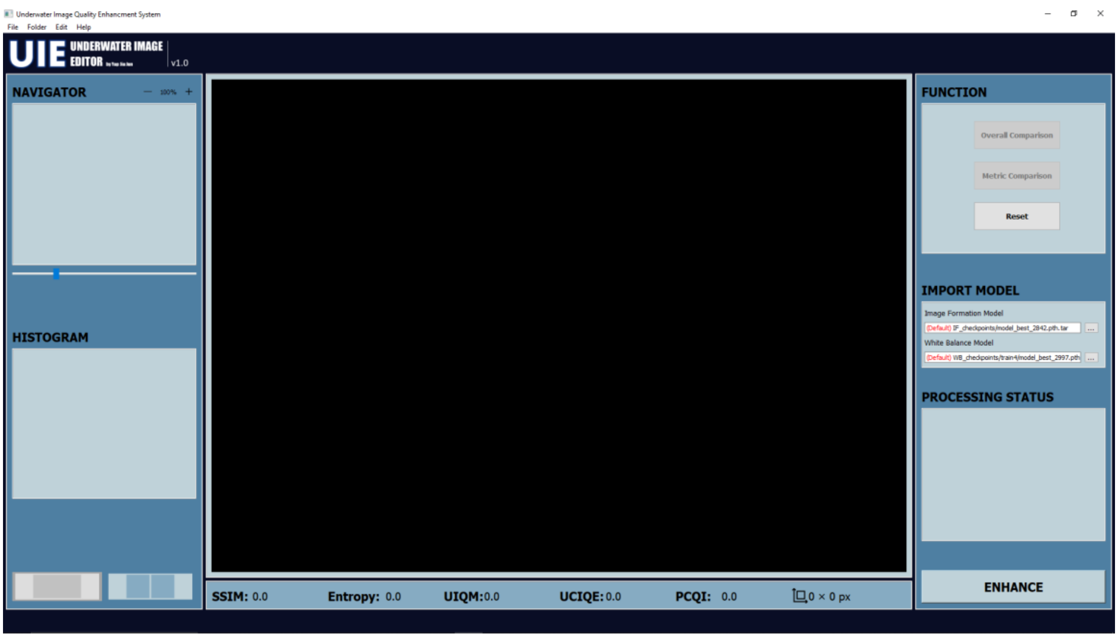


Figure 4.7 GUI Main Page

To use this application, user first need to choose to import single image or a folder of image by clicking the new button under the file or folder. This will prompt user to choose an input image or folder through the file explorer. Then, the chosen input image will be displayed within the middle frame. On the same time, all of the details of the image will also display on their respective frame. Next, user can choose to import a new pre-trained model by clicking the tool button beside it or using the default pre-trained model. Then, enhance button is ready to be initialized to process the input image with current pre-trained models. The processing status will be updated during the enhancement and after the enhancement is complete, the output will be display by replacing the original image. After that, user can view the different type of comparison by utilize the function given in the system. Finally, user can save the enhanced image in specific path of the device through file explorer.

Table 4.9 Description of icon push buttons in GUI

Icon	Function
—	To zoom out image
+	To zoom in image
	To change display image mode to single
	To change display image mode to double
...	File explorer will be executed to allow user to import pre-trained models.

Table 4.10 Description of text push buttons in GUI

Name	Function
File - New	File explorer will be executed to allow user to import test image as the input image of the system
File - Save	File explorer will be executed to allow user to save single output image into path selected
Folder - New	File explorer will be executed to allow user to import folder of test image.
Folder – Save	File explorer will be executed to allow user to save all output image of the folder into path selected
Edit – Open Image Strip	To open the image strip to choose the image in the selected folder
Edit – Close Image Strip	To close the image strip

Overall Comparison	A new window of all the outputs with metric evaluation will be displayed
Metric Comparison	A new window of metric evaluation of original image and enhanced image will be displayed
Reset	Reset everything into initial state
Enhance	To enhance image as proposed method.

Table 4.11 Description of slider in GUI

Parameter	Label Name	Range	Function
%	Navigator	[10,400]	To zoom in/out image

All the controls are explicitly detailed and tabulated in Table 4.9 – 4.11 for icon push button, text push button and slider in the GUI. A sample output of GUI is captured as illustrated in Figure 4.6, where the original and enhanced image are displayed together will their benchmarking values within the GUI. Figure 4.7 and 4.8 displays the window of overall comparison and metric comparison of the sample output.

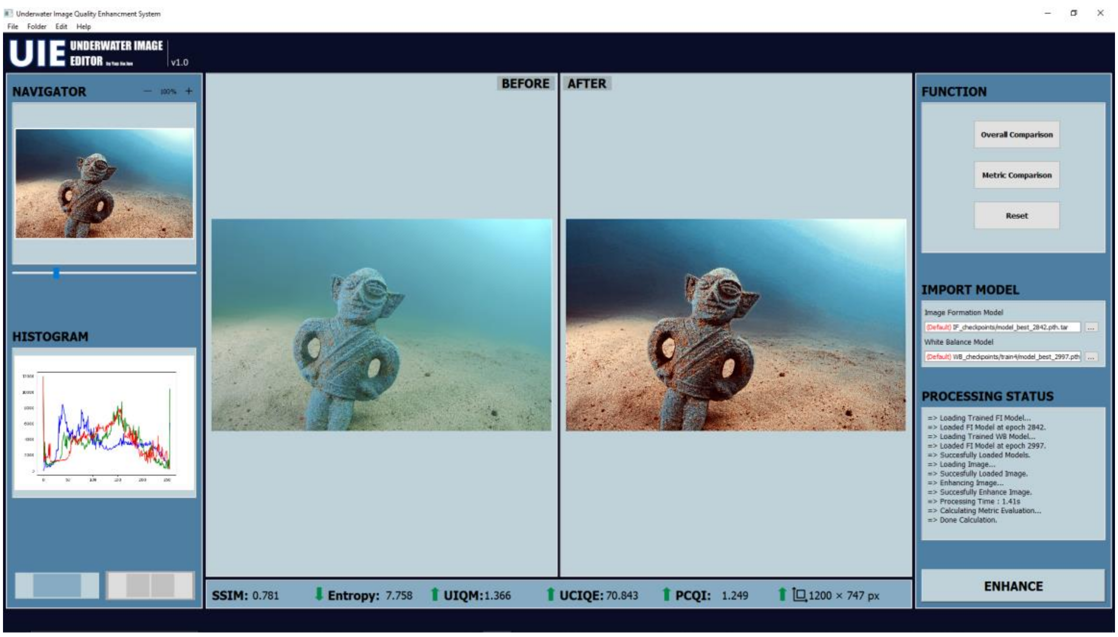
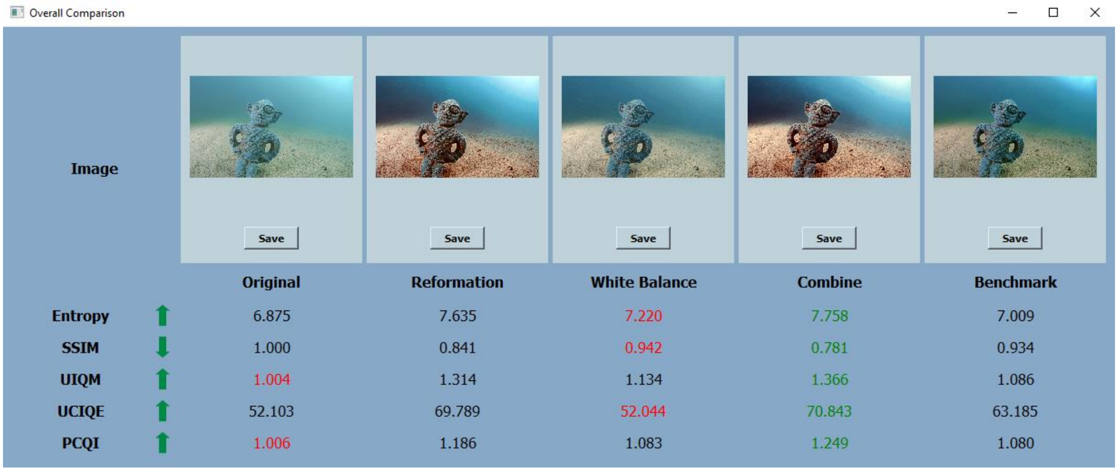


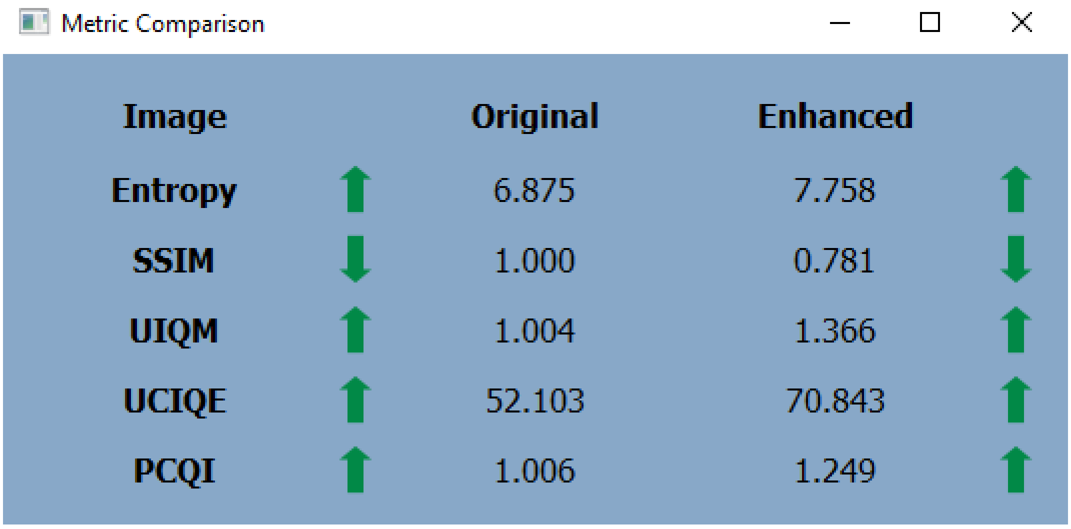
Figure 4.8 GUI with Sample Output



Overall Comparison

Image		Original	Reformation	White Balance	Combine	Benchmark
		Save	Save	Save	Save	Save
Entropy	↑	6.875	7.635	7.220	7.758	7.009
SSIM	↓	1.000	0.841	0.942	0.781	0.934
UIQM	↑	1.004	1.314	1.134	1.366	1.086
UCIQE	↑	52.103	69.789	52.044	70.843	63.185
PCQI	↑	1.006	1.186	1.083	1.249	1.080

Figure 4.9 GUI Overall Comparison



Metric Comparison

Image		Original	Enhanced	
Entropy	↑	6.875	7.758	↑
SSIM	↓	1.000	0.781	↓
UIQM	↑	1.004	1.366	↑
UCIQE	↑	52.103	70.843	↑
PCQI	↑	1.006	1.249	↑

Figure 4.10 GUI Metric Comparison

4.8 Pseudocode

The proposed system is comprised of deep learning-based enhancement techniques that have their own purpose in improving the underwater image quality enhancement. To get the conceptual of how these techniques are implemented, this subtopic is dedicated to discussing the pseudocode of these techniques as well as GUI actions.

4.8.1 Enhance Image

After test image is imported, user can proceed to enhance image with deep learning algorithm by pressing the enhance button. As depicted in figure 4.11, the algorithm of deep learning techniques will be implemented to perform enhancement. The deep learning models used will further described in the following subsection. The system will pass the input image into the deep learning models as a tensor. Before entering second model which is white balance model, the image will be down sampled to 656 to ensure the consistence run time. After undergo all models, the output tensor will be converted back into image which is in down sampled. Then, system will apply colour mapping to produce the output with original resolution. Lastly, system will compute the all the metric evaluation values of output image for benchmarking and the output image and values will be displayed in GUI

Algorithm 4.1 : Pseudocode for Enhance Image

IN : Input Image
Pretrained Image Formation Model
Pretrained White Balance Model

OUT : Output Enhanced Image

```
START
        Set a maximum value of down-sampled size to 656
        Get the state of both pretrained models
        Load image formation model and white balance model with the state
        Convert input image to numpy array
        Transforms numpy array to tensor
        Pass tensor through image formation model
        Convert the output tensor to PIL image
        Resize the image to maximum 656
        if width % 2 ** 4 == 0
             then new width = width
        else new width = width + 2 ** 4 - width % 2 ** 4
        end if
        if height % 2 ** 4 == 0
             then new height = height
        else new height = height + 2 ** 4 - height % 2 ** 4
```

```
        end if
        New size = (width, height)
        Resize the image to new size
        Convert the resize image to numpy array
        Transforms numpy array to tensor
        Pass tensor through white balance model
        Convert the output tensor to PIL image
        Apply colour mapping to the output image
        Reshape the resize image and output image to [-1,3]
        Compute the mapping function by fitting linear regression model
        between resize image and output image
        Apply the mapping function to output image from image formation
        model
        Clip out-of-gamut pixel to the range 0 to 1 of the image.
        Return output enhance image
END
```

Figure 4.11 Pseudocode of compare benchmark in system

4.8.2 Compare Benchmark

Figure 4.12 illustrates the pseudocode of compare benchmark among all the outputs in the system. When user pressed the compare benchmark button in the GUI, it will be initiated by creating a new window to show all the intermediate output as well as metric evaluation value for each output with indicator showing which values are better.

Algorithm 4.2 : Pseudocode for Compare Benchmark

IN : Input Image
Output from image formation model
Output from white balance model
Final output of proposed combine method
Ground Truth of the input image

OUT : A new window for overall comparison

```
START
        for all the images
                Calculate SSIM, Entropy, PCQI, UIQM and UCIQE of each image
```

```
    end for
    for all metric evaluation
        Get the maximum and minimum value among all the values
        for all values
            if value == maximum value
                then set the text colour to green
            if value == maximum value
                then set the text colour to red
            end if
        end for
    end for
    Set properties of the new window
    Set up frames to display all the images
    Set up labels to display all the values of metric evaluation
    Set up buttons to save each image
    Return a new window for overall comparison
END
```

Figure 4.12 Pseudocode of compare benchmark in system

CHAPTER 5

RESULTS AND DISCUSSION

5.1 Overview

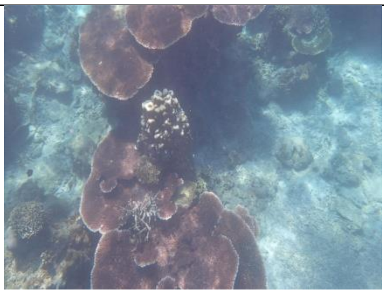
In this chapter, the underwater image quality enhancement system will undergo the chosen benchmarking to evaluate its effectiveness with regards to the objectives of this project. There are total of five benchmarking test which consists of SSIM, Entropy, PCQI, UIQM and UCIQE measure will be conducted to validate the proposed system. Specifically, these benchmarking are used to evaluate the implementation of deep learning models for image dehaze and colour restoration.

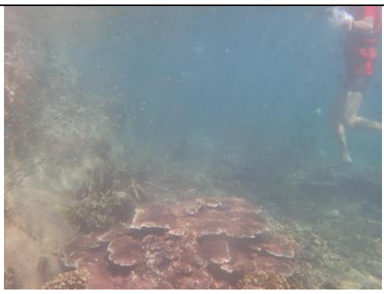
5.2 Input Image

The inputs of the system are acquired from UIEB dataset which is a dataset that used for evaluation of underwater image quality enhancement. A total of 55 images is chosen for testing. The input images are in PNG format and can be categorized into three groups: hazy images, images with low contrast and sharpness, and images with a bluish or greenish effect, with 20, 10, and 25 images respectively in each group. The list of sample input images that are suitable for evaluation of enhancement is chosen and tabulated in Table 5.1, Table 5.2 and Table 5.3.

Table 5.1 List of sample input images with hazy effect

No.	Test Image	Details
1.		Dataset: 115 Resolution: 640×480
2.		Dataset: 9 Resolution: 640×480
3.		Dataset: 46 Resolution: 640×480
4.		Dataset: 73 Resolution: 640×480
5.		Dataset: 336 Resolution: 640×480
6.		Dataset: 2 Resolution: 2000×1124

7.		Dataset: 871 Resolution: 640×480
8.		Dataset: 869 Resolution: 640×480
9.		Dataset: 862 Resolution: 640×480
10.		Dataset: 846 Resolution: 640×480
11.		Dataset: 84 Resolution: 640×480
12.		Dataset: 815 Resolution: 640×480

13.		Dataset: 75 Resolution: 640×480
14.		Dataset: 646 Resolution: 1280×720
15.		Dataset: 56 Resolution: 640×480
16.		Dataset: 346 Resolution: 1024×768
17.		Dataset: 247 Resolution: 640×480
18.		Dataset: 133 Resolution: 640×480

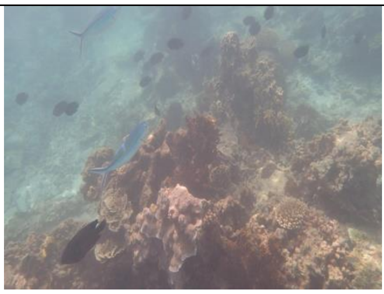
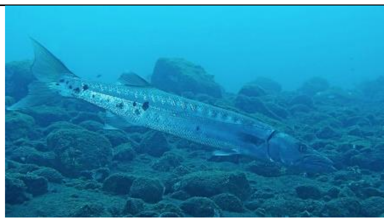
19.		Dataset: 134 Resolution: 640×480
20.		Dataset: 126 Resolution: 640×480

Table 5.2 List of sample input images with low contrast and sharpness

No.	Test Image	Details
1.		Dataset: 17 Resolution: 800×1150
2.		Dataset: 70 Resolution: 600×371
3.		Dataset: 277 Resolution: 1200×747

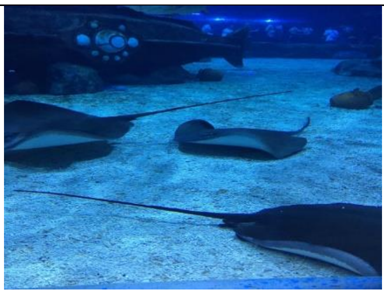
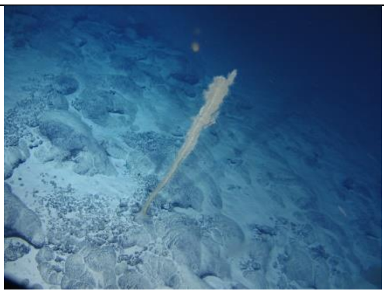
4.		Dataset: 880 Resolution: 1280×720
5.		Dataset: 803 Resolution: 640×480
6.		Dataset: 10103 Resolution: 640×360
7.		Dataset: 9607 Resolution: 1280×720
8.		Dataset: 37 Resolution: 640×480
9.		Dataset: 582 Resolution: 1200×750
10.		Dataset: 12191 Resolution: 1280×720

Table 5.3 List of sample input images with greenish and bluish effect

No.	Test Image	Details
1.		Dataset: 10226 Resolution: 640×360
2.		Dataset: 3201 Resolution: 1280×720
3.		Dataset: 52 Resolution: 1280×720
4.		Dataset: 7277 Resolution: 1280×720
5.		Dataset: 183 Resolution: 241×209
6.		Dataset: 1 Resolution: 800×450
7.		Dataset: 249 Resolution: 1280×720

8.		Dataset: 250 Resolution: 1280×960
9.		Dataset: 266 Resolution: 640×480
10.		Dataset: 287 Resolution: 736×552
11.		Dataset: 289 Resolution: 736×552
12.		Dataset: 296 Resolution: 1280×720
13.		Dataset: 304 Resolution: 1024×768

14.		Dataset: 451 Resolution: 500×333
15.		Dataset: 452 Resolution: 500×333
16.		Dataset: 26 Resolution: 791×537
17.		Dataset: 4 Resolution: 640×480
18.		Dataset: 532 Resolution: 500×375
19.		Dataset: 533 Resolution: 500×375

20.		Dataset: 561 Resolution: 1440×1080
21.		Dataset: 63 Resolution: 640×480
22.		Dataset: 71 Resolution: 1024×685
23.		Dataset: 734 Resolution: 1280×720
24.		Dataset: 816 Resolution: 640×480
25.		Dataset: 892 Resolution: 320×240

5.3 Result and Analysis

In this section, the results obtained from the project experiment are depicted and discussed. Additionally, benchmarking tests such as SSIM and Entropy were used to validate hazy images, while PCQI and UIQM were employed to validate contrast and sharpness. UCIQE was utilized to determine colour balance and restoration of underwater images. All these tests were conducted to compare and validate the techniques applied in the proposed system.

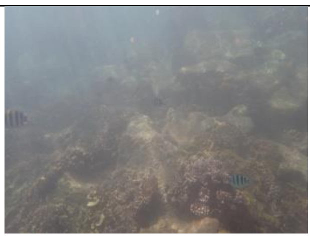
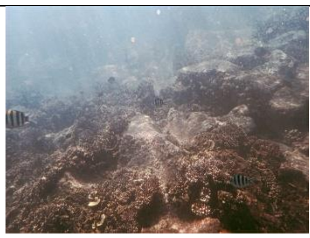
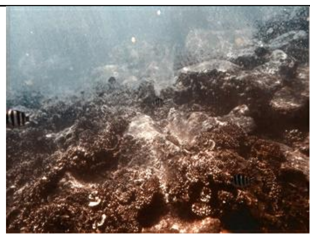
5.3.1 Summary of Evaluation on Proposed System

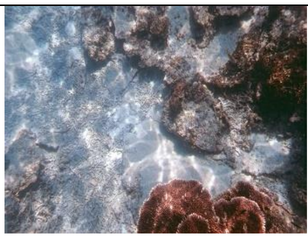
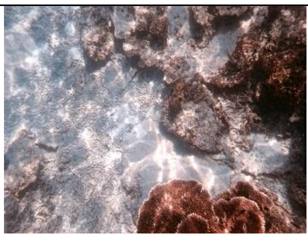
The test is conducted to evaluate the effect of implementing deep learning models for underwater image quality enhancement. There are two deep learning models as mentioned earlier which are image formation model and white balance model. The proposed method is to produce the output from combination of the two models. In the analysis of results, the effectiveness is evaluated in terms of visual quality and quantitative metrics.

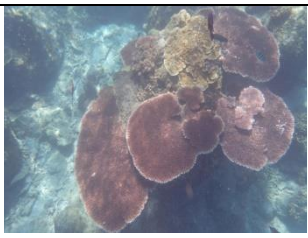
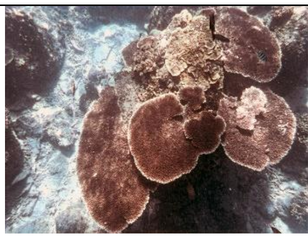
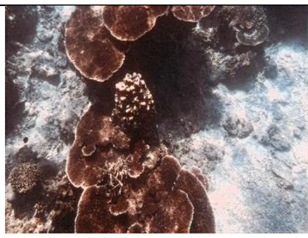
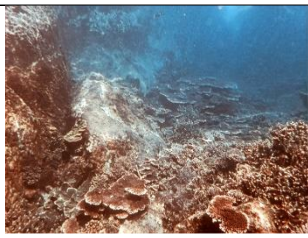
A. Underwater Hazy Image

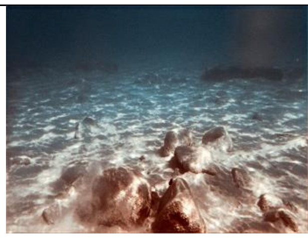
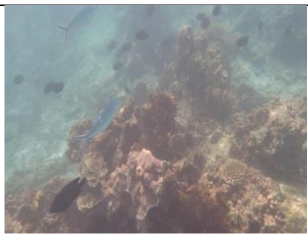
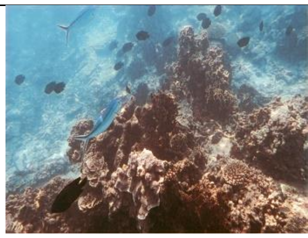
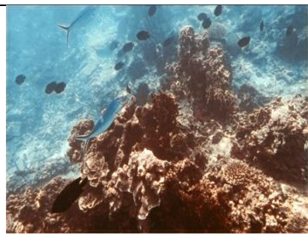
The first test was conducted for the group of underwater hazy images. To assess the effectiveness of the proposed method in haze removal, the output from the image formation model and original image were used for comparison. Table 5.4 presents the visual quality results of the proposed system.

Table 5.4 Visual Quality Results of Proposed System for Hazy Images

No.	Original	Image Formation Model	Combined
1.			

2.			
3.			
4.			
5.			
6.			
7.			
8.			

9.			
10.			
11.			
12.			
13.			
14.			
15.			

16.			
17.			
18.			
19.			
20.			

As depicted in Table 5.4, all the outputs are presented, with the leftmost column containing: the original image, the output from the Image Formation Model, and the output from the proposed method. From the result, it is clear that both the image formation model and proposed method effectively reduced the hazy effect in the images. However, a closer look reveals that proposed method provides better visual clarity compared to the image formation model. While the image formation model still leaves some haze, proposed method produces clearer outputs. This is probably due to the architecture of the deep learning model has reached its enhancement limit.

There could be biased and inaccurate deduction made by human visual comparison upon the results. Therefore, it is crucial to apply appropriate benchmarking test on the results to validate the objective of this project. SSIM is used to measure the structure similarity of two images while entropy is used to measure the amount of information content and variability in pixel values of an image.

A lower SSIM value shows that there are significant differences in how the images look, especially in their structure. On the other hand, a higher entropy value means that the enhanced image has more detail and variability in the pixels. If the entropy value is high, it suggests that the changes made to the image could be seen as improvements. Table 5.5 presents the quantitative metric results of the proposed system by using SSIM and Entropy.

Table 5.5 Quantitative Metric Results of Proposed System for Hazy Images

No.	Methods	Metric	
		SSIM	Entropy
1.	Original	1.000	6.442
	Image Formation Model	0.741	7.453
	Combined (Proposed)	**0.492**	**7.591**
2.	Original	1.000	6.028
	Image Formation Model	0.739	7.098
	Combined (Proposed)	**0.601**	**7.442**
3.	Original	1.000	5.676
	Image Formation Model	0.725	6.986
	Combined (Proposed)	**0.606**	**7.350**
4.	Original	1.000	6.384
	Image Formation Model	0.739	7.493
	Combined (Proposed)	**0.690**	**7.601**
5.	Original	1.000	6.040
	Image Formation Model	0.756	7.152
	Combined (Proposed)	**0.663**	**7.377**
6.	Original	1.000	6.409
	Image Formation Model	0.683	7.557

	Combined (Proposed)	**0.630**	**7.608**
7.	Original	1.000	6.224
	Image Formation Model	0.698	7.282
	Combined (Proposed)	**0.566**	**7.521**
8.	Original	1.000	7.060
	Image Formation Model	0.833	7.666
	Combined (Proposed)	**0.753**	**7.836**
9.	Original	1.000	6.961
	Image Formation Model	0.710	7.764
	Combined (Proposed)	**0.631**	**7.819**
10.	Original	1.000	6.611
	Image Formation Model	0.715	7.544
	Combined (Proposed)	**0.645**	**7.699**
11.	Original	1.000	5.806
	Image Formation Model	0.732	7.109
	Combined (Proposed)	**0.598**	**7.535**
12.	Original	1.000	6.868
	Image Formation Model	0.706	7.686
	Combined (Proposed)	**0.577**	**7.713**
13.	Original	1.000	5.615
	Image Formation Model	0.723	6.890
	Combined (Proposed)	**0.592**	**7.310**
14.	Original	1.000	7.352
	Image Formation Model	0.723	**7.875**
	Combined (Proposed)	**0.638**	7.868
15.	Original	1.000	5.918
	Image Formation Model	0.729	7.120
	Combined (Proposed)	**0.631**	**7.438**
16.	Original	1.000	6.031
	Image Formation Model	0.906	6.931
	Combined (Proposed)	**0.868**	**7.105**
17.	Original	1.000	6.181

	Image Formation Model	0.735	7.200
	Combined (Proposed)	**0.635**	**7.409**
18.	Original	1.000	5.989
	Image Formation Model	0.723	7.212
	Combined (Proposed)	**0.525**	**7.584**
19.	Original	1.000	6.516
	Image Formation Model	0.731	7.524
	Combined (Proposed)	**0.627**	**7.678**
20.	Original	1.000	6.572
	Image Formation Model	0.768	7.453
	Combined (Proposed)	**0.568**	**7.622**

Referring to the test values obtained and tabulated in Table 5.5, it is observed that proposed method has obtained more lowest score (bold text) of SSIM and highest score of entropy value in each output. This suggests that the proposed method is more effective at enhancing underwater images in terms of haze removal and produced high quality results compared to the image formation model. To further imply that which method outperforms the underwater image enhancement, the test values of each output are summarised by their mean value to compare overall performance.

Table 5.6 Summary of Quantitative Metric Results of Proposed System for Hazy Images

Metrics	SSIM	Entropy
Original	1.0000	6.3342
Image Formation Model	0.7408	7.3498
Combined (Proposed)	**0.6268**	**7.5553**

Based on mean values tabulated in table 5.6, it is evident that the proposed method achieves lowest SSIM value of 0.6268 and highest entropy value of 7.5553. In summary, the proposed method is able to improve in haze removal effectiveness by approximately 15.38% in SSIM and 2.80% in entropy as compared to image formation model. This further demonstrates that the output from the proposed method significantly improves upon the original image and contains more detail and variability in the pixels.

A clearer picture is able to captured by averaging the score of benchmarking test conducted upon the various models adopted in deep learning algorithm for the test inputs used. The average score is essential to represent their overall performance and ease the analysis for deduction as displayed in Table. Hence, bar chart is utilized to better visualize the comparison of quantitative result of original image and the outputs from image formation model and proposed method. Figure 5.1 and 5.2 displays the bar chart summarizing SSIM and Entropy results of proposed system for hazy images.

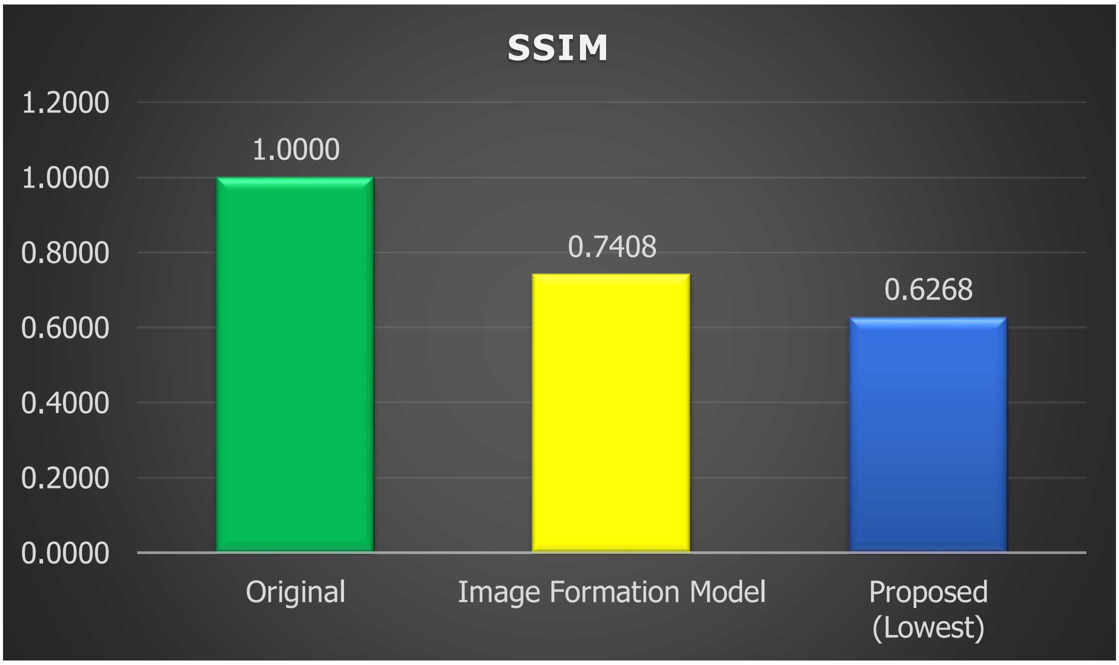


Figure 5.1 Bar Chart Summarizing the SSIM Results of Proposed System for Hazy Images.

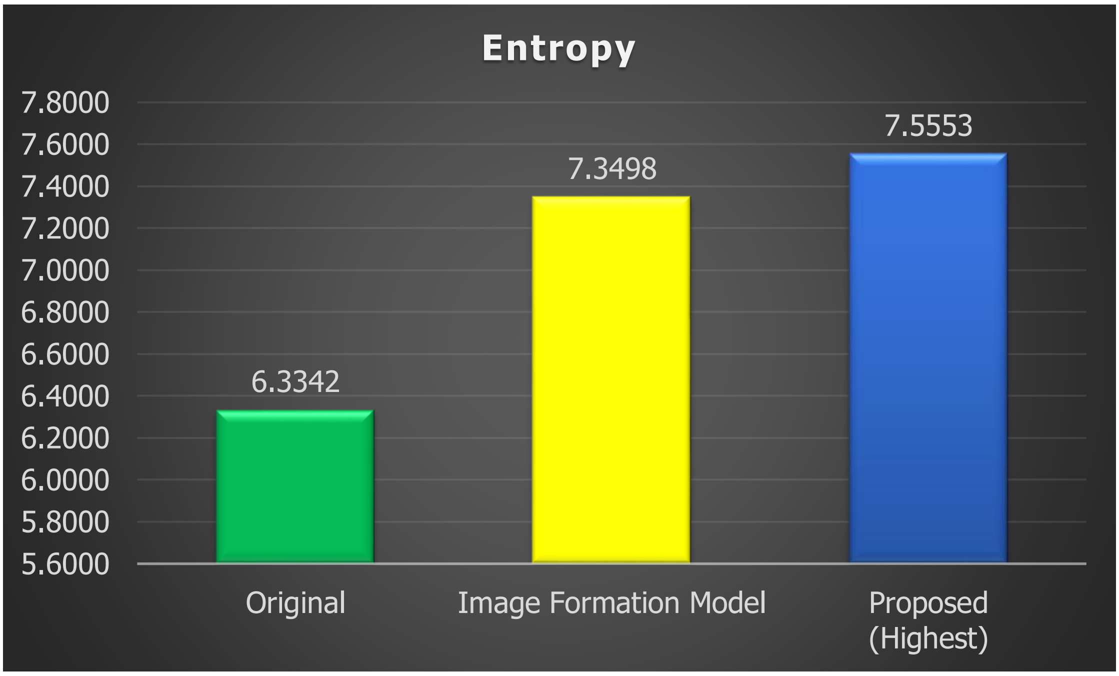


Figure 5.2 Bar Chart Summarizing the Entropy Results of Proposed System for Hazy Images.

The results in this section indicate that the first objective, which aimed to remove the haze-like effect in underwater images, has been achieved.

B. Underwater Image with Low Contrast and Sharpness

The second test was conducted for the group of underwater images with low contrast and sharpness. To assess the effectiveness of the proposed method in improving contrast and sharpness, the output from the image formation model, white balance model and original image were used for comparison. Table 5.7 presents the visual quality results of the proposed system.

Table 5.7 Visual Quality Results of Proposed System for Low Contrast and Sharpness Images

No.	Original	Image Formation Model	White Balance Model	Combined
1.				
2.				
3.				
4.				
5.				
6.				
7.				
8.				

9.				
10.				

Table 5.7 displays all the outputs, with the leftmost column containing the original image, the output from the image formation model, the output from the white balance model, and the output from the proposed method. From the results, it is evident that both the image formation model and proposed method effectively enhance the contrast and sharpness of the underwater images. However, the white balance model performed slightly worse, as some images remained dark, and the edges were not clear enough. Comparing the outputs of the image formation model and the proposed method, the latter appears clearer, brighter, and with higher contrast, making the edges more discernible. Since, there is only a slight difference between the two outputs, further evaluation using quantitative metrics is necessary to validate the quality of the results.

PCQI and UIQM have been chosen to evaluate contrast and sharpness of underwater images. PCQI measures the quality of contrast changes in the image while UIQM measures the quality of sharpness, contrast and colourfulness of underwater image. Higher values of both metrics indicate better overall quality of the underwater image, particularly in terms of contrast and sharpness. Table 5.8 presents the quantitative metric results of the proposed system by using PCQI and UIQM.

Table 5.8 Quantitative Metric Results of Proposed System for Low Contrast and Sharpness Images

No.	**Methods**	**Metric**	
		PCQI	**UIQM**
1.	Original	1.286	1.120
	Image Formation Model	1.352	1.350
	White Balance Model	1.324	1.230
	Combined (Proposed)	**1.407**	**1.447**

2.	Original	1.137	0.675
	Image Formation Model	1.505	0.925
	White Balance Model	1.200	0.779
	Combined (Proposed)	**1.543**	**0.975**
3.	Original	1.006	1.004
	Image Formation Model	1.186	1.314
	White Balance Model	1.083	1.134
	Combined (Proposed)	**1.249**	**1.366**
4.	Original	0.423	0.188
	Image Formation Model	0.766	0.935
	White Balance Model	0.692	0.948
	Combined (Proposed)	**0.844**	**0.997**
5.	Original	1.477	0.812
	Image Formation Model	1.676	1.279
	White Balance Model	1.542	0.903
	Combined (Proposed)	**1.722**	**1.418**
6.	Original	1.016	0.364
	Image Formation Model	1.306	1.051
	White Balance Model	1.284	0.858
	Combined (Proposed)	**1.335**	**1.067**
7.	Original	0.675	0.195
	Image Formation Model	0.931	0.699
	White Balance Model	0.753	0.483
	Combined (Proposed)	**0.983**	**0.715**
8.	Original	1.380	0.626
	Image Formation Model	1.683	1.109
	White Balance Model	1.634	1.003
	Combined (Proposed)	**1.824**	**1.167**
9.	Original	1.406	1.280
	Image Formation Model	1.503	1.542
	White Balance Model	1.454	1.394
	Combined (Proposed)	**1.571**	**1.568**

10.	Original	0.737	0.546
	Image Formation Model	1.184	1.102
	White Balance Model	0.892	0.551
	Combined (Proposed)	**1.225**	**1.176**

Based on test values obtained and tabulated in Table 5.8, it is noticeable that proposed method achieved more highest score (bold text) for both PCQI and UIQM in each output. This indicates that the proposed method is more successful in enhancing underwater images by improving contrast and sharpness, resulting in high-quality outcomes compared to the image formation model and white balance model. To further imply that which method outperforms the underwater image enhancement, the test values of each output are summarised by their mean value to compare overall performance.

Table 5.9 Summary of Quantitative Metric Results of Proposed System for Low Contrast and Sharpness Images

Metrics	PCQI	UIQM
Original	1.0543	0.6810
Image Formation Model	1.3092	1.1306
White Balance Model	1.1858	0.9283
Combined (Proposed)	**1.3703**	**1.1896**

Based on mean values of PCQI and UIQM tabulated in table 5.9, it is evident that the proposed method achieves highest PCQI value of 1.3703 and highest UIQM value of 1.1896. In summary, the proposed method demonstrates an enhancement in improving contrast and sharpness of the image by approximately 4.67% in PCQI and 5.22 % in UIQM as compared to image formation model. This further proves that the output from the proposed method exhibits the highest quality in terms of contrast, sharpness, and colourfulness among all outputs, including the original image. To visualize these quantitative results effectively, bar charts are employed to compare the quantitative results of the original image with those from the image formation model, white balance model, and proposed method. Figure 5.3 and 5.4 displays the bar chart summarizing PCQI and UIQM results of proposed system for low contrast and sharpness images.

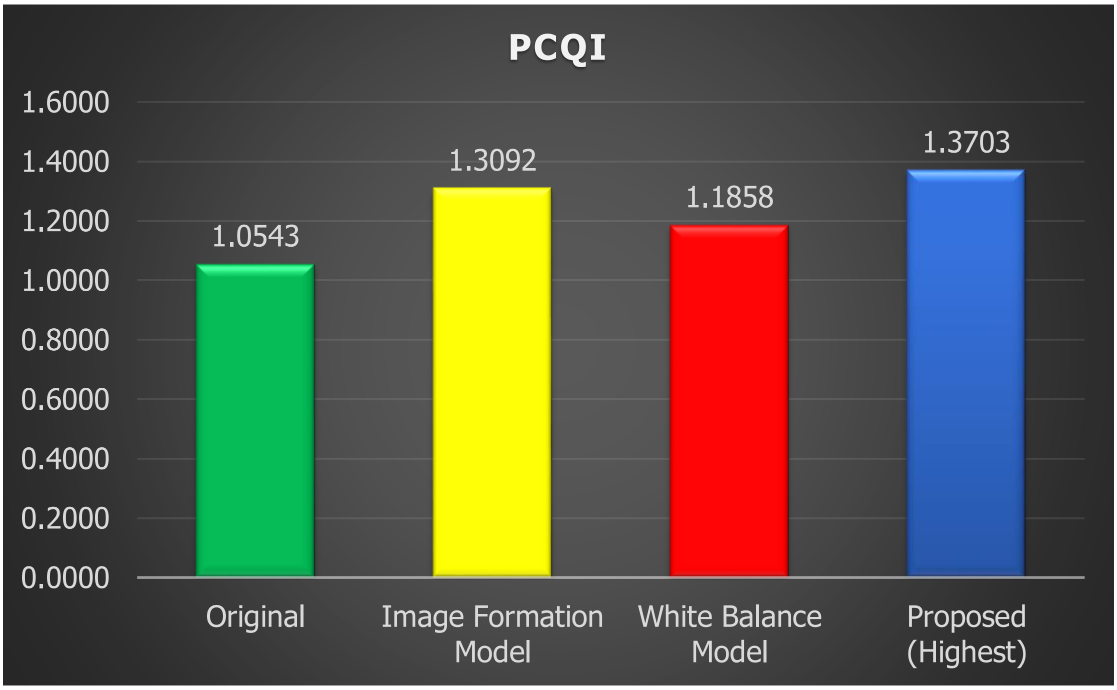


Figure 5.3 Bar Chart Summarizing the PCQI Results of Proposed System for Low Contrast and Sharpness Images

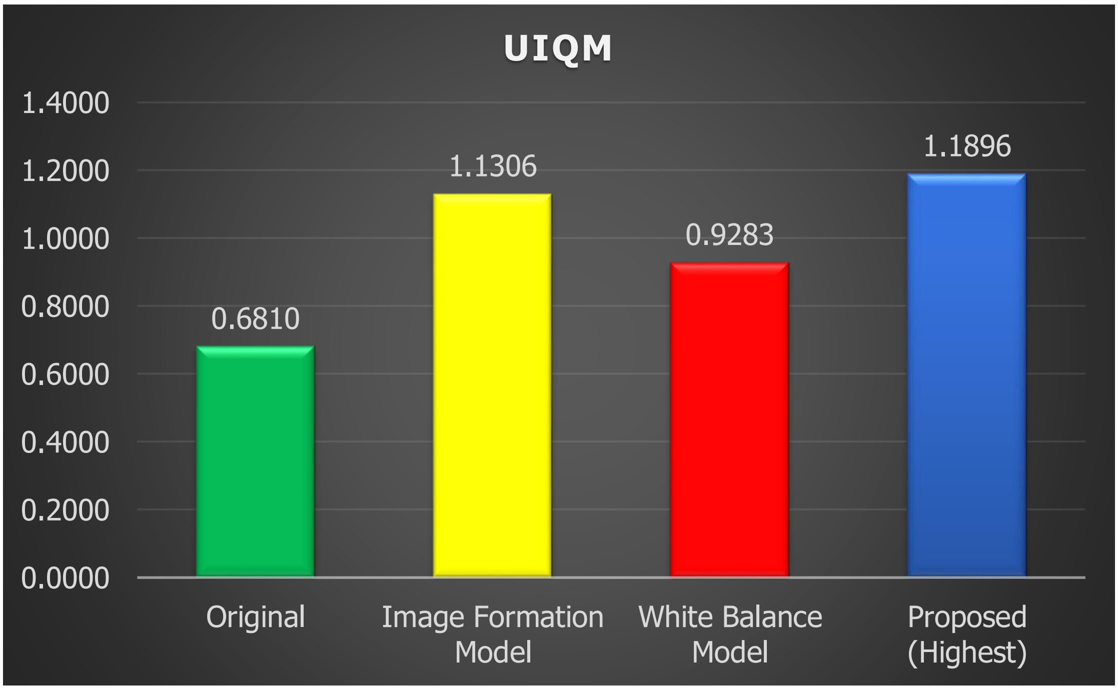


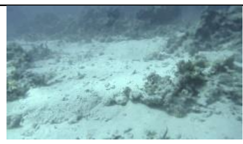
Figure 5.4 Bar Chart Summarizing the UIQM Results of Proposed System for Low Contrast and Sharpness Images

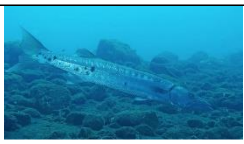
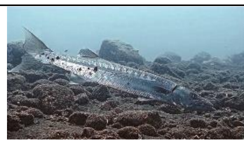
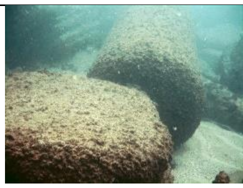
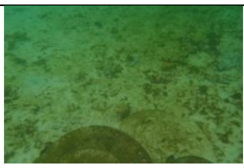
The results in this section indicate that the second objective, which aimed to increase sharpness, contrast and colour balance in the degraded underwater colour images, has partially been achieved.

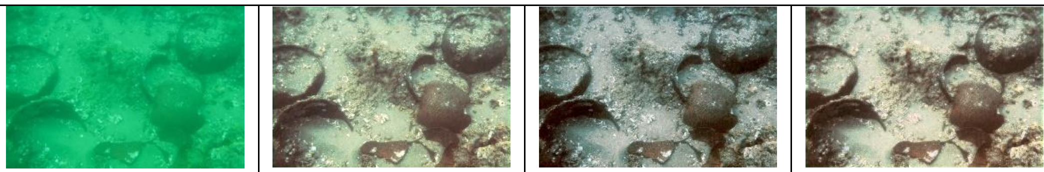
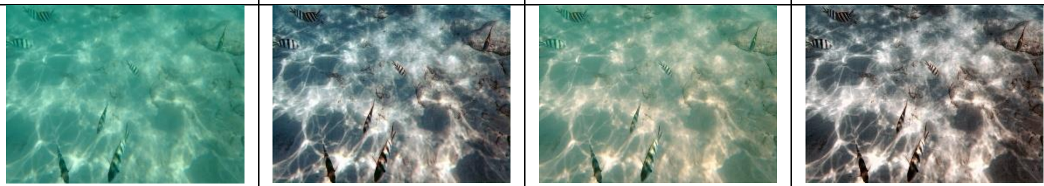
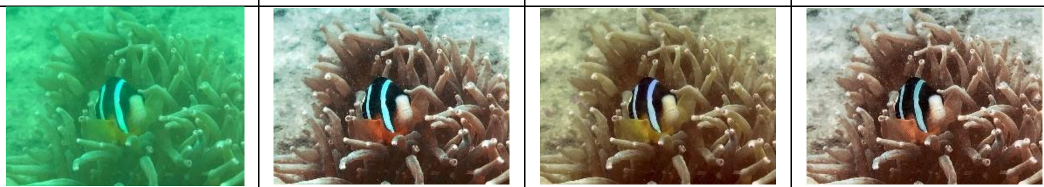
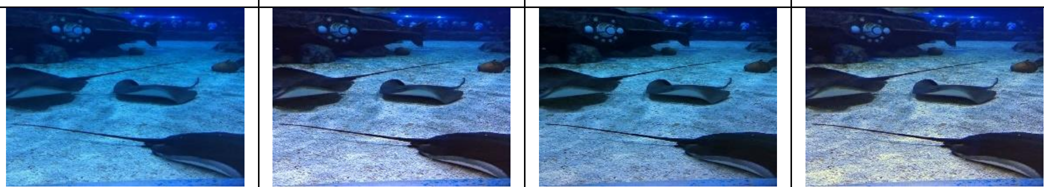
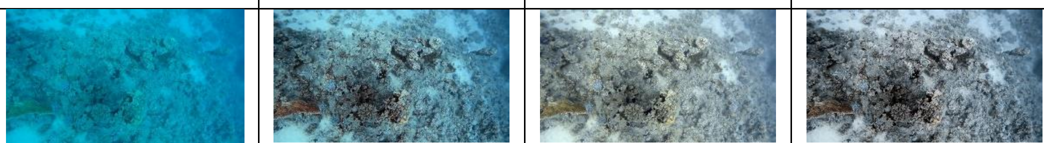
C. Underwater Greenish and Bluish Image

The third test was conducted for the group of underwater greenish and bluish images. To assess the effectiveness of the proposed method in improving contrast and sharpness, the output from the image formation model, white balance model and original image were used for comparison. Table 5.10 presents the visual quality results of the proposed system.

Table 5.10 Visual Quality Results of Proposed System for Greenish and Bluish Images

No.	Original	Image Formation Model	White Balance Model	Combined
1.				
2.				
3.				
4.				
5.				

6.				
7.				
8.				
9.				
10.				
11.				
12.				
13.				
14.				
15.				

16.	
17.	
18.	
19.	
20.	
21.	
22.	
23.	
24.	

25.				

Table 5.10 displays all the outputs, with the leftmost column containing the original image, the output from the image formation model, the output from the white balance model, and the output from the proposed method. The results indicate that both the image formation model and proposed method effectively remove the greenish and bluish effects in underwater images. However, the white balance model performed slightly less effectively, leaving some images with a small amount of greenish and bluish tone. Comparing the outputs of the image formation model and the proposed method, the latter shows superior performance in colour balancing, with most of the images appearing clear and restored in colour. Since, there is only a slight difference between the two outputs, further evaluation using quantitative metrics is necessary to validate the quality of the results.

UCIQE has been selected to evaluate the enhancement of underwater greenish and bluish images. UCIQE assesses the colour degradation and distortion in water caused by light attenuation, scattering, and colour cast in underwater colour images. A higher UCIQE value indicates better quality in terms of colour balance and colour restoration for underwater colour images. Table 5.11 presents the quantitative metric results of the proposed system by using UCIQE.

Table 5.11 Quantitative Metric Results of Proposed System for Greenish and Bluish Images

No.	Methods	Metric
		UCIQE
1.	Original	42.405
	Image Formation Model	59.033
	White Balance Model	58.937
	Combined (Proposed)	**62.329**
2.	Original	47.771
	Image Formation Model	57.809

	White Balance Model	52.747
	Combined (Proposed)	**62.888**
3.	Original	59.782
	Image Formation Model	69.466
	White Balance Model	61.601
	Combined (Proposed)	**69.822**
4.	Original	28.861
	Image Formation Model	48.171
	White Balance Model	48.210
	Combined (Proposed)	**57.248**
5.	Original	48.912
	Image Formation Model	**61.465**
	White Balance Model	47.762
	Combined (Proposed)	54.405
6.	Original	44.223
	Image Formation Model	65.586
	White Balance Model	64.214
	Combined (Proposed)	**65.935**
7.	Original	53.009
	Image Formation Model	67.488
	White Balance Model	**67.826**
	Combined (Proposed)	62.640
8.	Original	42.814
	Image Formation Model	64.745
	White Balance Model	55.616
	Combined (Proposed)	**70.424**
9.	Original	34.506
	Image Formation Model	55.739
	White Balance Model	53.265
	Combined (Proposed)	**67.134**
10.	Original	54.237
	Image Formation Model	**69.415**

	White Balance Model	64.521
	Combined (Proposed)	63.230
11.	Original	49.511
	Image Formation Model	**68.019**
	White Balance Model	66.804
	Combined (Proposed)	67.912
12.	Original	34.605
	Image Formation Model	54.587
	White Balance Model	51.345
	Combined (Proposed)	**61.215**
13.	Original	38.417
	Image Formation Model	62.726
	White Balance Model	64.098
	Combined (Proposed)	**68.691**
14.	Original	23.344
	Image Formation Model	42.581
	White Balance Model	46.048
	Combined (Proposed)	**50.794**
15.	Original	34.300
	Image Formation Model	55.094
	White Balance Model	54.658
	Combined (Proposed)	**60.796**
16.	Original	28.017
	Image Formation Model	56.832
	White Balance Model	**67.338**
	Combined (Proposed)	59.966
17.	Original	40.764
	Image Formation Model	67.690
	White Balance Model	46.660
	Combined (Proposed)	**69.068**
18.	Original	40.248
	Image Formation Model	**64.734**

	White Balance Model	58.840
	Combined (Proposed)	64.599
19.	Original	39.139
	Image Formation Model	**63.854**
	White Balance Model	61.915
	Combined (Proposed)	58.350
20	Original	40.075
	Image Formation Model	65.201
	White Balance Model	63.183
	Combined (Proposed)	**66.157**
21.	Original	56.476
	Image Formation Model	**67.009**
	White Balance Model	58.883
	Combined (Proposed)	64.019
22.	Original	67.310
	Image Formation Model	**69.822**
	White Balance Model	66.652
	Combined (Proposed)	65.670
23.	Original	35.377
	Image Formation Model	58.589
	White Balance Model	58.279
	Combined (Proposed)	**64.551**
24.	Original	48.391
	Image Formation Model	61.753
	White Balance Model	**63.282**
	Combined (Proposed)	61.611
25.	Original	60.001
	Image Formation Model	**67.175**
	White Balance Model	57.364
	Combined (Proposed)	66.699

Based on test values obtained and tabulated in Table 5.11, it is noticeable that proposed method achieved more highest score (bold text) for UCIQE in each output. This suggests that the proposed method is more effective in removing the greenish and bluish effects, as well as in colour balance and colour restoration, resulting in high-quality underwater colour images compared to the image formation model and white balance model. To further establish which method outperforms others in underwater image enhancement, the test values of each output are summarized by their mean value to compare overall performance.

Table 5.12 Summary of Quantitative Metric Results of Proposed System for Greenish and Bluish Images

Metrics	**UCIQE**
Original	54.6248
Image Formation Model	77.2292
White Balance Model	73.0024
Combined (Proposed)	**79.3077**

Based on mean values of UCIQE tabulated in table 5.9, it is evident that the proposed method achieves highest value of 79.3077. In summary, the proposed method demonstrates an enhancement in restoring and balancing the colour in the image by approximately 2.69% in UCIQE as compared to image formation model. This further proves that the output from the proposed method exhibits the highest quality of underwater colour image among all outputs, including the original image by removing the colour bias, balance and restore the colour in the image. To visualize this quantitative result effectively, bar chart is employed to compare the quantitative results of the original image with those from the image formation model, white balance model, and proposed method. Figure 5.5 illustrates the bar chart summarizing UCIQE results of proposed system for greenish and bluish images.

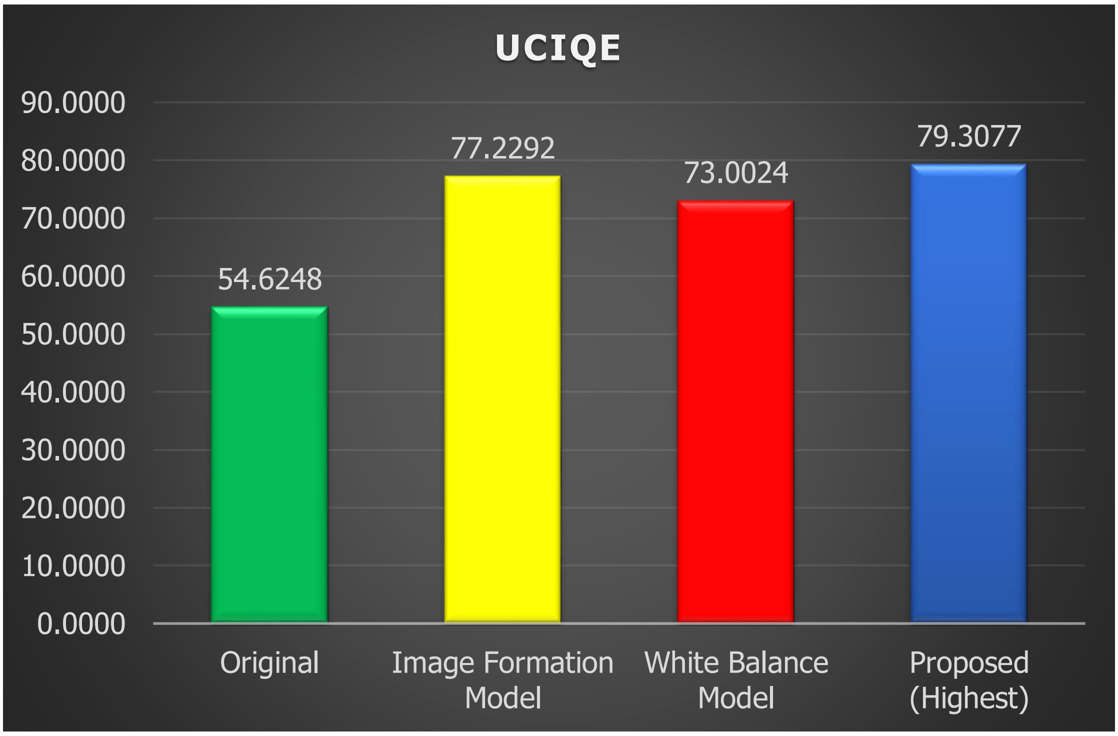


Figure 5.4 Bar Chart Summarizing the UCIQE Results of Proposed System for Greenish and Bluish Images

The results in last section and this section indicate that the second objective, which aimed to increase sharpness, contrast and colour balance in the degraded underwater colour images, has fully been achieved.

5.4 Summary

In this chapter, the results of proposed technique were observed, evaluated and discussed. Total 55 input images used were acquired from UIEB dataset which consists of challenging set and a set of raw image and corresponding ground truth. The inputs images consist of 20 underwater hazy images, 10 underwater images with low contrast and sharpness and 25 underwater images with greenish and bluish effect to test on the proposed system. Three tests were conducted on each group of images, with five metric evaluations used to assess the effectiveness of combining dual CNN models and comparing them with each model separately. It was observed that the proposed technique performed well in SSIM, Entropy, PCQI, UIQM, and UCIQE. The results suggest that the proposed system is capable of producing enhanced underwater images with removed haze, high contrast and sharpness, as well as balanced colour and restoration. Overall, it can be concluded that the aim and objective of this project have been achieved, demonstrating that the performance of deep learning-based underwater image enhancement can be enhanced by combining image formation and white balance models.

CHAPTER 6

CONCLUSION

6.1 Summary

In this project, a deep learning-based underwater image enhancement system was proposed and implemented. The primary objective was to achieve high-quality underwater images, driven by the fact that the existing underwater image enhancement is still in a developmental stage and lacks comprehensiveness. Hence, the proposed system leverages a convolutional neural network in both the image formation and white balance model, combing them sequentially.

The image formation model is a mathematical representation, simulates the effects of underwater conditions and considers the optical imaging properties of the underwater environment in image formation. Meanwhile, the white balance model plays a crucial role in removing colour casts caused by scene illumination. Through the integration of these models and the power of deep learning, the system learns to enhance underwater images to high quality.

A comprehensive review of existing literature revealed a significant weakness in current underwater image enhancement methods. These methods often excel in enhancing specific categories of underwater images， requiring manual parameter adjustments for different types. For instance, some methods focus solely on haze removal, while other prioritize colour restoration. This highlighting the complexity of optical imaging underwater. Researches must consider numerous factors, demanding substantial resources and time to develop a comprehensive underwater enhancement technique.

However, our project addresses these challenged by implementing deep learning in underwater image enhancement, enable the computer to learn the complexities from the provided dataset. This approach overcomes the intricate nature of optical imaging underwater, eliminating the need for manual adjustments. To achieve this, a deep kerning model is essential for the training and testing processes. As a result, the image formation model and white balance model are designed and trained to produce high quality underwater images.

Benchmarking of SSIM, Entropy, PCQI, UIQM and UCIQE are conducted upon the result of deep learning-based underwater image enhancement system to validate the proposed method. These benchmarking are corresponded to similarities, information and variability, quality of contrast, sharpness and quality of colour balance. The evaluation has been carried out on three different group of test images. It was observed that the proposed technique performed well and achieved an improvement of 15.38%, 2.80%, 4.67%, 5.22% and 2.69% in SSIM, Entropy, PCQI, UIQM, and UCIQE respectively as compared to image formation model. The results suggest that the proposed system is capable of producing enhanced underwater images with removed haze, high contrast and sharpness, as well as balanced colour and restoration. To conclude the combination of both models, it has a significant improvement compared to single model. Hence, it can be deduced that the proposed method has effectively improved the quality of underwater colour image in which the objectives are validated.

The strength of the proposed system lies in the combination of the image formation model and white balance model and the implementation of deep learning algorithm. In fact, it is difficult to obtain enhancement result as good as indoor and outdoor image due the unique environment of underwater. However, more efforts are required in designing deep learning and training models, provide more datasets in every category and more training time for model so that the model can learn more on the complexity of underwater image.

6.2 Contribution

There are two main contributions that this project has effectively done in the field of computer vision. The first contribution is removing haze-like effect in underwater image by using image formation model. The second contribution is to increase contrast, sharpness, and colour balance in degraded underwater colour image by develop a fusion of image formation model and white balance model.

6.2.1 Remove Haze-Like Effect in Underwater Image by using Image Formation Model

The Image Formation Model is a CNN model comprising a backward scatter estimation module and a direct transmission estimation module. It has been designed to estimate the backward scatter and direct transmission components in an image and perform a reformulation calculation to compute the image with the true intensity values of the scene. This approach is capable of removing the haze-like effect in underwater images, as this effect is caused by the attenuation of light underwater.

6.2.2 Increase Contrast, Sharpness, and Colour Balance in Degraded Underwater Colour Image by Develop a Fusion of Image Formation Model and White Balance Model

The sequential integration of the image formation model and white balance model serves to enhance the quality of underwater images by increase contrast, sharpness and colour balance. The image formation model is tasked with reconstructing the image within the scene with true intensity values considering backward scatter estimation and direct transmission estimation. Once the image is reconstructed, colour restoration and balancing are necessary to enhance the overall colour quality. To maintain consistent runtime performance, the image is down sampled and colour mapping is applied to the original resolution image to produce final output. In essence, the collaborative function of these models significantly contributes to the continued improvement of underwater image quality.

6.3 Future Work

Although the proposed system has been evaluated and proved to be effective in achieving a higher quality result but it is not without its flaws. Thus, this subtopic is dedicated to future researchers to further enhance the proposed method in order to yield a supreme result.

6.3.1 Integrated and Unified Framework

As it stands, the combination of image formation model and white balance model is very effective in improving the overall quality of underwater image compared to a single model. However, they are two separate models with its own function which will consume more time to train the models and lack of adaptability. By integrating these models into a singular and unified framework is envisioned to enhance computational efficiency, reduce redundancy and improve adaptability. The integrated model streamlines image enhancement processes, fostering seamless collaboration between image formation and white balance functionalities. By consolidating these functions, we aim to simplify parameter tuning, mitigate challenges associated with managing multiple models and showcase the potential for a more effective approach to underwater image enhancement.

6.3.2 Inclusive Training Dataset

As for now, the underwater dataset is very limited for researches to carry out study on underwater image enhancement. Same goes to deep learning technique, it required mode dataset to learn in order to form a comprehensive model to achieve high accuracy and quality result. The objective is to investigate the profound impact of incorporating diverse datasets, encompassing various categories of underwater scene environments, in the training process. By exposing the model to a broader range of scenarios, including different lighting conditions, water clarity levels, and underwater terrains, we aim to empower the model with more comprehensive understanding of the complexities inherent in the underwater environment and optical imaging. The envisioned outcome is a heightened ability of the model to navigate diverse underwater conditions, resulting in a more nuanced and effective enhancement process. This exploration underscores the significance of dataset diversity in refining and broadening the capabilities of the model for robust underwater image enhancement.

REFERENCES

Afifi, M., & Brown, M. S. (2020). Deep White-balance editing. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Akkaynak, D., & Treibitz, T. (2018). A revised Underwater image formation model. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Cai, B., Xu, X., Jia, K., Qing, C., & Tao, D. (2016). DehazeNet: An end-to-end system for Single Image Haze Removal. *IEEE Transactions on Image Processing*, *25*(11), 5187–5198.

Chen, X. (2019). Image enhancement effect on the performance of convolutional neural networks.

Chen, X., Zhang, P., Quan, L., Yi, C., & Lu, C. (2021, January 7). *Underwater Image Enhancement based on Deep Learning and Image Formation model*. arXiv.org. https://arxiv.org/abs/2101.00991

Downing, J. (2008). Effects of Light Absorption and Scattering in Water Samples on OBS® Measurements.

Duarte, A. *et al.* (2016) 'A dataset to evaluate underwater image restoration methods', *OCEANS 2016 - Shanghai* [Preprint].

Fu, X., & Cao, X. (2020). Underwater Image Enhancement with global–local networks and compressed-histogram equalization. *Signal Processing: Image Communication*, *86*, 115892.

González, R. C., & Woods, R. E. (2008). *Digital Image Processing*. Prentice Hall.

González-Sabbagh, S. P., & Robles-Kelly, A. (2023). A survey on Underwater Computer Vision. *ACM Computing Surveys*, **55**(13s), 1–39.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014, June 10). *Generative Adversarial Networks*.

Hou, G. *et al.* (2020) 'Benchmarking underwater image enhancement and restoration, and beyond', *IEEE Access*, 8, pp. 122078–122091.

Hu, K., Weng, C., Zhang, Y., Jin, J., & Xia, Q. (2022). An overview of underwater vision enhancement: From traditional methods to recent deep learning. *Journal of Marine Science and Engineering*, *10*(2), 241.

Hu, Y. et al. (2018) Underwater image restoration based on Convolutional Neural Network, PMLR. Available at: https://proceedings.mlr.press/v95/hu18a.html.

Islam, M. J., Xia, Y., & Sattar, J. (2020). Fast underwater image enhancement for improved visual perception.

Katherine, A. S., Zhang, J., Olson, E. A., & Matthew, J.-R. (2019). *UWStereoNet: Unsupervised Learning for Depth Estimation and Color Correction of Underwater Stereo Imagery*. IEEE.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444.

Li, C., Guo, C., Ren, W., Cong, R., Hou, J., Kwong, S., & Tao, D. (2019, November 26). *An underwater image enhancement benchmark dataset and beyond*.

Li, C., Guo, J., & Guo, C. (2018). Emerging from water: Underwater image color correction based on weakly supervised color transfer. *IEEE Signal Processing Letters*, *25*(3), 323–327.

Li, H., Li, J., & Wang, W. (2019, June 30). *A fusion adversarial underwater image enhancement network with a public test dataset*.

Liu, P., Wang, G., Qi, H., Zhang, C., Zheng, H., & Yu, Z. (2019). Underwater image enhancement with a deep residual framework. *IEEE Access*, *7*, 94614–94629.

Lu, H., Li, Y., Uemura, T., Kim, H., & Serikawa, S. (2018). Low illumination underwater light field images reconstruction using deep convolutional neural networks. *Future Generation Computer Systems*, *82*, 142–148.

McAndrew, A. (2015). *A computational introduction to Digital Image Processing*. Chapman & Hall/CRC.

Nababan, B., Ulfah, D., & Panjaitan, J. P. (2021). Light propagation, coefficient attenuation, and the depth of one optical depth in different water types. *IOP Conference Series: Earth and Environmental Science*, **944**(1), 012047.

Narasimhan, S. G., & Nayar, S. K. (2002). Vision and the Atmosphere. In *International Journal of Computer Vision* (pp. 233–254). essay, Kluwer Academic.

Pan, P., Yuan, F., & Cheng, E. (2018). Underwater Image De-Scattering and Enhancing using DehazeNet and HWD. *Journal of Marine Science and Technology*, **26**(4).

Perez, J., Attanasio, A. C., Nechyporenko, N., & Sanz, P. J. (2017). A deep learning approach for underwater image enhancement. *Biomedical Applications Based on Natural and Artificial Computing*, 183–192.

Sahu, P., Gupta, N., & Sharma, N. (2014). A survey on Underwater Image Enhancement Techniques. *International Journal of Computer Applications*, *87*(13), 19–23.

Sara, U., Akter, M., & Uddin, M. S. (2019). Image quality assessment through FSIM, SSIM, MSE and PSNR—a comparative study. *Journal of Computer and Communications*, **07**(03), 8–18.

Schechner, Y. Y., & Karpel, N. (2004). Clear underwater vision. *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*

Subudhi, B. N., Kapoor, M., Rout, D. K., Thangaraj, V., & Jakhetiya, V. (2023). *Passive Visual Underwater Surveillance: A Survey*.

Tao, Y., Dong, L., Xu, L., & Xu, W. (2021). Effective solution for underwater image enhancement. *Optics Express*, **29**(20), 32412.

Teuwen, J., & Moriakov, N. (2020). Convolutional Neural Networks. *Handbook of Medical Image Computing and Computer Assisted Intervention*, 481–501.

Tsai, D.-Y., Lee, Y. and Matsuyama, E. (2007) 'Information entropy measure for evaluation of Image Quality', *Journal of Digital Imaging*, **21**(3), pp. 338–347.

Uplavikar, P., Wu, Z., & Wang, Z. (2019, May 30). *All-in-one underwater image enhancement using domain-adversarial learning*.

Wang, S., Ma, K., Yeganeh, H., Wang, Z., & Lin, W. (2015). A patch-structure representation method for quality assessment of contrast changed images. *IEEE Signal Processing Letters*, **22**(12), 2387–2390.

Wang, Y., Zhang, J., Cao, Y., & Wang, Z. (2017). A deep CNN method for underwater image enhancement. *2017 IEEE International Conference on Image Processing (ICIP)*.

Yang, D., & Sun, J. (2018). Proximal dehaze-net: A prior learning-based deep network for single image Dehazing. *Computer Vision – ECCV 2018*, 729–746.

Yang, M., Hu, J., Li, C., Rohde, G., Du, Y., & Hu, K. (2019). An in-depth survey of underwater image enhancement and restoration. *IEEE Access*, *7*, 123638–123657.

Yang, M., & Sowmya, A. (2015). An underwater color image quality evaluation metric. *IEEE Transactions on Image Processing*, **24**(12), 6062–6071.

Yu, X., Qu, Y., & Hong, M. (2018). Underwater-gan: Underwater image restoration via conditional generative Adversarial Network. *Pattern Recognition and Information Forensics*, 66–75.

APPENDIX

Sample coding

FI_Model.py > ...

```python
from model_blocks import *

import torch
import numpy as np
import torch.nn as nn


class FINet(nn.Module):
    def __init__(self): #checked
        super(FINet,self).__init__()
        #self.in_channels=3
        self.BsNet = BsConvBlock(3,3)
        self.DtNet = DtConvBlock(3,8)

    def forward(self,x): #checked
        BSE = self.BsNet(x)
        DTE = self.DtNet(torch.cat((x*0+BSE,x),1))
        out = ((x-BSE)*DTE + BSE)
        return torch.clamp(out,0.,1.)
```

WB_Model.py > ...

```python
from model_blocks import *

import torch
import numpy as np
import torch.nn as nn


class WBNet(nn.Module):
    def __init__(self): #Checked
        super(WBNet,self).__init__()
        self.encoder_inc = DoubleConvBlock(3, 24)
        self.encoder_down1 = DownBlock(24, 48)
        self.encoder_down2 = DownBlock(48, 96)
        self.encoder_down3 = DownBlock(96, 192)
        self.encoder_bridge_down = BridgeDown(192, 384)
        self.awb_decoder_bridge_up = BridgeUP(384, 192)
        self.awb_decoder_up1 = UpBlock(192, 96)
        self.awb_decoder_up2 = UpBlock(96, 48)
        self.awb_decoder_up3 = UpBlock(48, 24)
        self.awb_decoder_out = OutputBlock(24, 3)

    def forward(self,x): #Checked
        x1 = self.encoder_inc(x)
        x2 = self.encoder_down1(x1)
        x3 = self.encoder_down2(x2)
        x4 = self.encoder_down3(x3)
        x5 = self.encoder_bridge_down(x4)
        x_awb = self.awb_decoder_bridge_up(x5)
        x_awb = self.awb_decoder_up1(x_awb, x4)
        x_awb = self.awb_decoder_up2(x_awb, x3)
        x_awb = self.awb_decoder_up3(x_awb, x2)
        awb = self.awb_decoder_out(x_awb, x1)
        return awb
```