WAV-DUC: A Dynamic Image Enhancement Algorithm for Underwater Environment

by

Washikur Rahman
21301029
Tausif Mahmud
21301345
Khondker Shahed Arnob
21301028
Sadnima Amin Chowdhury
21301260

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
October 2024

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material that has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Washikur Rahman 21301029	Tausif Mahmud 21301345
Khondker Shahed Arnob 21301028	Sadnima Amin Chowdhury 21301260

Approval

The thesis/project titled "WAV-DUC: A Dynamic Image Enhancement Algorithm for Underwater Environment" submitted by

1. Washikur Rahman (21301029)
2. Tausif Mahmud (21301345)
3. Khondker Shahed Arnob (21301028)
4. Sadnima Amin Chowdhury (21301260)

Of Summer, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on October 20, 2024.

Examining Committee:

Supervisor:
(Member)

Md. Ashraful Alam, PhD
Associate Professor
Department of Computer Science and Engineering
BRAC University

Program Coordinator:
(Member)

Md. Golam Rabiul Alam, PhD
Professor
Department of Computer Science and Engineering
BRAC University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Associate Professor
Department of Computer Science and Engineering
Brac University

Acknowledgement

Firstly, all praise to the Great Allah for whom our thesis has been completed without any major interruption.
Secondly, to our Supervisor Dr. Ashraful Alam sir for his kind support and advice in our work. He helped us whenever we needed help.

Abstract

Object detection underwater is particularly difficult because of issues such as low illumination, illumination changes, and the scarcity of labeled underwater images. To overcome these issues, this paper introduces a new type of hybrid image enhancement method (Wav-DUC). The method incorporates the strengths of CLAHE, DCP, DWT, Unsharp Masking, and Bilateral filtering. While adopting the advantages of these techniques, the proposed method improves image quality under the various conditions of marine environments. The enhancement is done in such a way that the enhancement methods are selectively used above a predetermined threshold so as to enhance the image, resulting in an optimum result. testing the model with benchmark dataset [2], we achieved a PSNR score of 21.78 and an SSIM score of 0.8535. Also, the method has an average execution time of 64.78ms on Raspberry Pi 3. This proves the robustness and faster response time of the algorithm, making it suitable for edge devices.

Keywords: Underwater, CLAHE, DWT, DCP, YOLO, Raspberry pi, Unsharp masking, Transfer Learning

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

BN Batch Normalization

CNN Convolutional Neural Network

$COCO$ Common Objects in Context dataset

$CycleGAN$ Cycle Generative Adversarial Network

MAP Mean Average Precision

mAP Mean Average Precision

$PSNR$ Peak Signal to Noise Ratio

ROV Remotely Operated Vehicle

$SFA-FPN$ Scale-Frequency Attention Feature Pyramid Network

SSD Single Shot Multibox Detector

$SSIM$ Structural Similarity Index Measure

$TranSDet$ Transfer Learning for Small Object Detection

$TT100K$ Traffic Time 100K dataset

$UResnet$ Underwater Resnet

$VDSR$ Very-Deep Super-Resolution Reconstruction

Chapter 1

Introduction

1.1 Motivation

Underwater image enhancement and exploration has numerous aspects including the scientific, environmental, economic, and security domains. The vast and complex ecosystems found in the aquatic environment are crucial in determining the general state of our planet. Comprehending and observing submerged entities enhances our capacity to oversee and protect this vital domain.

1.1.1 Environmental Monitoring and Conservation

Monitoring and protecting marine ecosystems depend heavily on the ability to inspect objects submerged in water. Effective conservation efforts are facilitated by the identification of marine objects and the evaluation of their populations and movements. Monitoring submerged objects can significantly expand the domain of underwater for further research.

1.1.2 Resource Management and Fisheries

Fish populations can be identified and monitored to assist fisheries authorities in enforcing ethical fishing methods, preventing overfishing, and maintaining the long-term sustainability of marine resources. This in turn helps ensure the food security of the world and sustains the fishing-dependent communities' means of subsistence.

1.1.3 Infrastructure Development and Maintenance

Planning, building, and maintaining underwater infrastructure has a significant dependence on underwater surveillance. Exact monitoring of submerged objects is essential to prevent collisions, disruptions, and potential environmental hazards when laying submarine cables, erecting offshore platforms, or maintaining underwater pipelines. This guarantees the dependability and security of vital infrastructure supporting numerous sectors.

1.1.4 Search and Rescue Operations

Search and rescue operations depend heavily on the ability to identify and locate underwater objects in emergencies, such as maritime mishaps or natural disasters.

Quickly locating submerged objects, wreckage, and individuals presumed missing can improve the efficiency of response teams, possibly saving lives and lessening the effects of disasters on coastal communities.

1.2 Research Conundrum

The distinct features of the aquatic environment present a wide range of challenges for underwater object detection, many of which have been extensively documented in recent literature. These difficulties include, but are not limited to, poor visibility, distorted images, and dynamic nature of underwater lighting.

1.2.1 Low Visibility

The impact of light absorption and scattering in water, which results in decreased contrast and visibility [19]. The decrease in visibility presents a major challenge to object detection methodologies as well as complicates the visual interpretation of underwater scenes [19].

1.2.2 Distorted Imagery

Conventional computer vision algorithms are challenged by the changing geometric and chromatic characteristics of underwater images since these algorithms operate under stable conditions. In underwater scenarios, addressing these distortions becomes essential for accurate and dependable object detection [18].

1.2.3 Dynamic Nature of Underwater Lighting

The light-absorbing and light-spreading properties of water particles create constantly varying illumination conditions, which pose a significant challenge to conventional computer vision models that are not designed to handle such dynamics [10].

1.3 Research Objectives

The study focuses on the use of a hybrid image enhancement algorithm to solve the multi-faceted problem of underwater image enhancement. The research goals of our investigation include the analysis of image enhancement techniques, including contrast enhancement, dehazing, and sharpening as well as adaptive filters and assessment of their impact on the identification of images underwater. Our strategy involves developing a traditional algorithm to work on different scenarios and environments associated with the underwater environment that will improve image quality in optimal time and with fewer resources. Ultimately, this research aims to develop a flexible, reliable, and efficient underwater image enhancement algorithm that can be used to create a pipeline for underwater research.

Chapter 2

Related Work

Computer vision in the domain of underwater imaging has specific difficulties where novel strategies are very much needed to solve problems like low contrast, texture distortion, and also small training datasets.
The challenge of underwater object detection was discussed by Zhu, Liu and Weibo [39]. This task is marred by the low contrast, distorted texture, and also the limited number of datasets available. They developed a new approach based on the YOLOv3-tiny network, which was pre-trained on the Pascal VOC2012 dataset and later trained and tested on CHINAMM2019. In their approach, pre-trained deep learning models were adapted to the specific needs of the underwater environment thus improving the detection performance. The results demonstrated a significant performance enhancement, with an 8.79% increase in Mean Average Precision (mAP) relative to the SSD method and a roughly halved runtime as compared to the original YOLOv3 algorithm. Nevertheless, the study identified some limitations, including the dependence on the availability and quality of training data and further optimization for real-time detection in a wide range of underwater conditions [39].
The article "Underwater Image Enhancement With a Deep Residual Framework" [20] focused on the problem of insufficient contrast, blurred details, and also color distortion in raw underwater images that resulted in the severe loss of information for visual problems such as segmentation and tracking tasks. They offered a deep residual solution that included the use of CycleGAN to generate synthetic underwater images as training data for CNN models and the inclusion of VDSR in underwater resolution applications, leading to Underwater Resnet. The outcome revealed that their approaches worked very well in underwater image restoration, especially in color correction and also detail enhancement, with the UResnet-P- A model performing the best. After 2000 epochs, the VDSR-P-A model that was tested on dataset B yielded PSNR and SSIM scores of 43.9115 and 0.8689, respectively. Further, the VDSR models and UResnet models with and without BN layers on the testing dataset A presented visible edge evaluation scores reflecting that the model was able to improve the edge details and contrast.
In the work "Enhancing Unpaired Underwater Images With Cycle Consistent Network" [30], the author discusses the problem of improving the imaging quality underwater – often characterized by low contrast, haziness, and poor visibility. The approach employs a Cycle Consistent Network, an algorithmic technique that differs from the methods of restoring the images through optical models and physical pa-

rameters. The essence of this approach is improving the visual clarity without any resource intensity and also processing time associated with other methods. Significant enhancements in the image visibility and detail for underwater imaging, which is a lot more suited to tasks such as the detection of objects and navigation. The classical image processing method reached a 9-15 PSNR and SSIM of 50% to 70%. However, the unsupervised approach suggested in the paper yielded a PSNR of 18-25 and SSIM values from 83% to 89%, which shows much better results regarding texture enhancement, color grading, color correction, contrast enhancement, and also overall image quality improvement. Nevertheless, the research recognizes limitations like possible dependence on the diversity of the dataset used and the necessity for additional optimization to be suitable across underwater conditions.
The critical factors of image processing and enhancements for underwater applications have been a focus of recent studies that pointed out the difficulties associated with a lack of data below water compared to the land, posing a major challenge in developing effective underwater image recognition systems [1]. This paper aims to analyze the difficulties of using deep learning methods in underwater environments, presenting the YOLO algorithm, describing image reading methodology, explaining implementation procedure, and reporting findings as well as comparisons. The methodology used an ROV with a camera for fish detection in the tank, data training and testing was carried out in two steps; the first stage without fine-tuning the pre-trained original dataset while at second phase transfer learning and tuning of three layers of the Yolov3 algorithm were incorporated. The fine-tuned dataset, which was augmented with 200 additional training data, demonstrated an improvement of 4% in the mean Average Precision (mAP) score which proves that transfer learning and also adding more training data is beneficial. On the other hand, the paper details limitations in the research, mainly due to a suboptimal test environment.
Underwater object detection poses a serious trial, given impairments such as color reduction, uneven lighting, and amblyopia. Such trials are overcome by utilizing performance enhancements to make the best out of sparse datasets [13]. Object localization is done through Yolov2 along with Resnet-50 for feature extraction. Moreover, instance segmentation is done using bounding boxes along with non-max suppression to exclude boxes with low object probability and bounding boxes with the highest common area. The data has been trained with variable learning rates and the best model yielded an accuracy of 95.9% .
The introduced Scale-Frequency Attention Feature Pyramid Network (SFA-FPN) and anchor relation module significantly improve small-object detection on TT100K, BUUISE-MO-Lite, and COCO datasets. Notable improvements include 8.0% for Faster R-CNN and 22.7% for RetinaNet on TT100K-Lite, 32.2% for RetinaNet, and 12.8% for YOLOv3 on BUUISE-MO-MO-Lite [35].
A real-time object detection algorithm [18], YOLOV3-Pruning(transfer), to accelerate object detection in real-time image processing. This method, leveraging transfer learning, demonstrates notable advancements. Compared to unused YOLOv3-Pruning, the model incorporating transfer learning exhibits a substantial 20.29% increase in Mean Average Precision at 50 intersections over union (MAP-50) and a significant 16.08% improvement in MAP-75. The proposed noise-immune deep detection framework, SWIPENET+CMA, addresses challenges in object detection in underwater scenes with heterogeneous noisy data and small objects.
In the paper 'Object detection in noisy underwater images' [10], the proposed noise-

immune deep detection framework, SWIPENET+CMA, addresses challenges in object detection in underwater scenes with heterogeneous noisy data and small objects. SWIPENET employs high-resolution and semantic-rich Hyper Feature Maps for improved small object detection.
The framework introduces a novel sample-weighted detection loss to control the influence of training samples based on their weights. The study utilizes VGG16 as the backbone for the SWIPENET+CMA framework. In Faster RCNN, four backbones—VGG16, ResNet50, ResNet101, and FPN—are employed. YOLOv3 uses its original DarkNet53 network. RetinaNet, FCOS, FreeAnchor, and GHM utilize ResNet50 as the backbone. In Faster RCNN, four backbones—VGG16, ResNet50, ResNet101, and FPN—are employed. YOLOv3 uses its original DarkNet53 network. RetinaNet, FCOS, FreeAnchor, and GHM utilize ResNet50 as the backbone.

Chapter 3

Methodology

3.1 Data Collection

BRACKISH Dataset

The dataset encompasses images of marine organisms categorized into six distinct classes: shrimp (548 images), starfish (7,912 images), crabs (12,348 images), jellyfish (637 images), small fish (10,768 images), and fish (3,352 images). Meticulously curated, the dataset is stratified into 11,739 training images, 1,467 validation images, and 1,468 test images, each meticulously selected to encapsulate the intricacies of brackish underwater environments. This stratification accounts for diverse factors including variations in ambient conditions, object dimensions, and illumination dynamics. Furthermore, among the dataset's 14,674 images, 12,444 are annotated with bounding boxes delineating marine species such as fish, crabs, small fish, shrimp, and jellyfish. The dataset allocation adheres to the standard practice, with 80% designated for training, 10% for testing, and 10% for validation purposes [25].
In **Figure 3.1** and **Figure 3.2** we can see the class distribution and and dataset distribution of the Brackish Dataset. **Figure 3.3** samples shows samples from the Brackish Dataset [25].

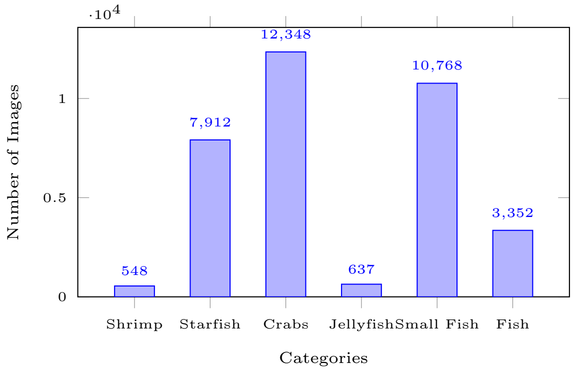


Figure 3.1: Distribution of Images Across Categories

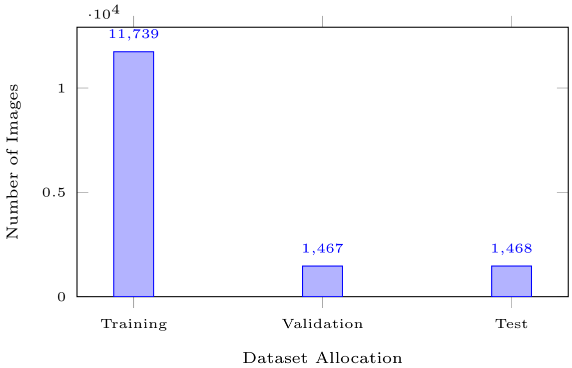


Figure 3.2: Dataset Allocation

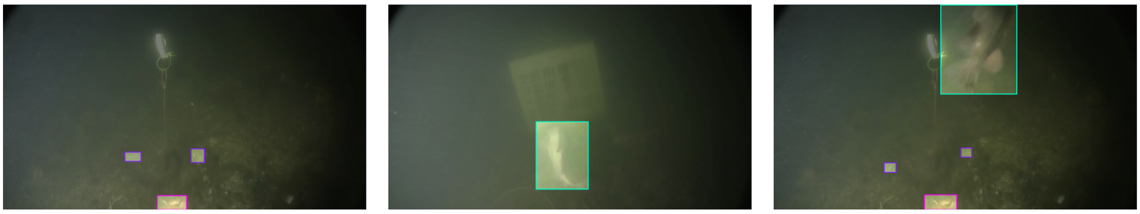


Figure 3.3: Bracksih Dataset Samples

DeepFish Dataset

The DeepFish dataset contains video streams of 20 different types of underwater environments allowing the capture of various situations. Different habitats like seagrass bed, low complexity bed, complex reef, large boulder etc. The dataset contains 4505 annotated images and 15463 bounding box images [23]. **Figure 3.4** shows samples images from the dataset.
The **Table 3.1** shows the distribution of images according to the habitats.

Habitat	**Annotated Images**	**Bounding Boxes**
Rocky mangrove prop roots	407	2995
Sparse algal bed	462	4146
Upper-Mangrove medium Rhizophora	140	140
Sandy mangrove prop roots	250	555
Complex reef	283	611
Low algal bed	450	2031
Seagrass bed	502	1716
Low complexity reef	300	651
Boulders	149	454
Mixed substratum mangrove prop roots	100	198
Reef trench	207	382
Upper mangrove tall Rhizophora	99	99
Large boulder	102	108
Muddy mangrove pneumatophores and trunk	351	499
Muddy mangrove pneumatophores	106	106
Bare substratum	131	155
Mangrove - mixed pneumatophore prop root	152	183
Rocky mangrove - large boulder and trunk	79	79
Rock shelf	139	235
Large boulder and pneumatophores	96	117
Total	**4505**	**15463**

Table 3.1: DeepFish Dataset Overview: Number of Annotated Images and Bounding Boxes per Habitat

Figure 3.4: DeepFish dataset samples

3.2 Proposed Enhancement Algorithm

3.2.1 Background and Motivation

The adverse conditions that can severely degrade image quality make underwater image enhancement a particularly challenging area of research. Low visibility, high noise levels, reduced contrast and lack of sharpness between underwater scenes make accurate visualization and interpretation difficult.

Deep learning approaches are explored as one means of overcoming the complexity of underwater image enhancement, though with their challenges. Multiple problems have been addressed simultaneously with the help of deep learning models, like Convolutional Neural Networks (CNNs) and Generative Adversary Networks (GANs). Unfortunately, these models are computationally expensive and resource-intensive. For this reason, they are also typically not used in situations where faster response with constrained computational power is required, particularly for edge devices constrained by computing power like underwater drones, autonomous vehicles, or portable devices used in marine research.

To overcome these challenges, we propose a new method called Wav-DUC (Wavelet-based Advanced Vision - Dehazing and Unsharp Masking). Wav-DUC is specially devised to improve underwater images under a wide range of conditions. It integrates five powerful techniques to improve underwater images comprehensively. To reduce the hazy appearance that is common in underwater environments, we use the Dark Channel Prior (DCP) to effectively reduce the hazy appearance, using wavelet transforms to extract better local features. A natural blurriness evident in scattering and absorption of light underwater can be countered with Unsharp Masking, which applies to remove it from images to improve image sharpness and highlight edges. CLAHE (Contrast Limited Adaptive Histogram Equalization) is used to locally increase the contrast of the images, allowing the recovery of details lost in the low-contrast regions of underwater scenes.

The adaptive nature of Wav-DUC is one important aspect that distinguishes it from any existing method. Our approach is based on an iterative threshold-based mechanism that recognizes that each underwater image is unique in terms of quality and the amount of degradation. This mechanism allows the algorithm to adaptively apply the enhancement operations between input images. For instance, if images are very hazy, DCP can be made stronger for application, but images with low contrast but good visibility would benefit from CLAHE.

To summarize, the methodology proposed herein – Wav-DUC – represents a substantial improvement for underwater image enhancement, yielding a complete technique for solving these specific image problems while maintaining the computational efficiency necessary for applications.

3.2.2 Methods

The proposed algorithm comprises five core techniques: CLAHE, Dark Channel Prior (DCP), Wavelet Transformation, Unsharp masking, and Biliteral filter. The methods presented above are integrated in a specific order to perform optimally.

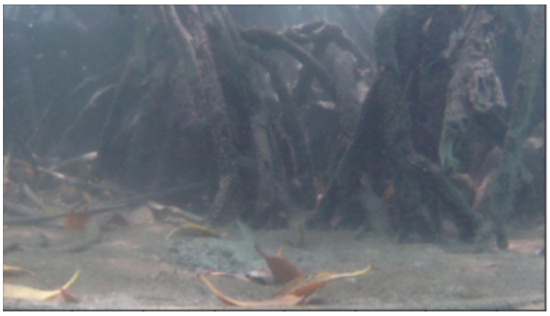
CLAHE

CLAHE (Contrast Limited Adaptive Histogram Equalization) is used to bring increased contrast to an image and specially in regions with poor contrast. Unlike standard histogram equalization, CLAHE operates on smaller regions or tiles, so that localized enhancement can be achieved without over-amplifying noise. Since the parts lack any coloring themselves, they can be particularly effective for undersea images where different lighting conditions can cause one part of the image to be overexposed while another is underexposed [36].
For CLAHE, we used a grid size of 8x8 in our implementation. In other words, the image is broken into 8x8 tiles, and histogram equalization is performed separately within each tile. With localization, CLAHE can adapt to different regions of the image, increasing contrast where it is needed without affecting already well-illuminated regions. The tiles are then equalized and then merged back together by bilinear interpolation to maintain smooth transitions between neighboring regions. In addition, we used a clip limit of 2.0 in our approach to avoid over-amplification of contrast. The clip limit is a cutoff in the histogram bin height that limits how high the histogram bins can reach preventing the noise from becoming super-enhanced. By limiting the contrast enhancement on this value CLAHE performs a trade-off between emphasizing the detail and not causing artificial artifacts or excessive noise amplification. The grid size and clip limit used in this choice are such that contrast enhancement in underwater images is effective, while preserving the overall visual integrity of the scene, by making subtle features more discernible.
As shown in **Figure 3.9**, the enhanced image has a much broader histogram compared to the original image, indicating a wider distribution of pixel values.

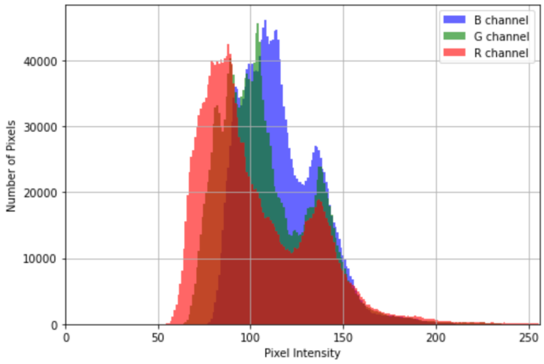
$$\text{CLAHE}(x, y) = (\text{CDF}_{\text{clip}}(x, y) - \text{CDF}_{\text{clip}}(x_0, y_0)) \times \text{Range of pixel values} \quad (3.1)$$

Where:

- $\text{CDF}_{\text{clip}}(x, y)$ is the cumulative distribution function for the pixel at (x, y) after applying the clip limit.
- $\text{CDF}_{\text{clip}}(x_0, y_0)$ is the minimum value of the CDF in the tile.
- The *Area of Tile* and the *Range of pixel values* are used to normalize the result.

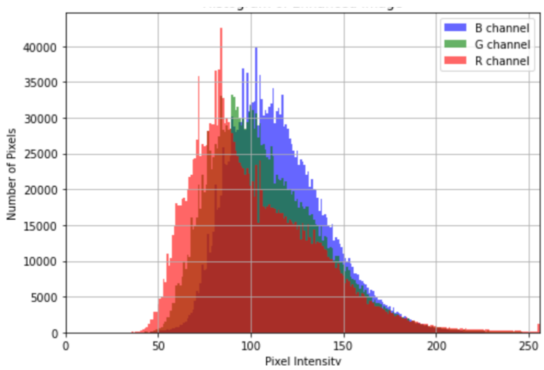


(a) Original Image



(b) Original Image Histogram

(c) Contrast Enhanced Image



(d) Contrast Enhanced Image Histogram

Figure 3.5: Histogram Comparison

Descrete Wavelet Transform

The Discrete Wavelet Transform (DWT), one of the most common forms of wavelet transformations, decomposes images into a set of approximation and detail coefficients in a redundant hierarchical manner [29]. Approximation coefficients are low-frequency components of the image, and they represent the overall structure of the image in a smoothed version. On the other hand, detail coefficients represent high frequencies that distinguish our image from fine details and textures.
The wavelet-based sharpness measure can be defined as follows:

$$S = \frac{\sum_{d \in \text{detail\_coeffs}} \sum_{(i,j) \in d} |d_{i,j}|}{\sum_{(i,j) \in A} |A_{i,j}|} \tag{3.2}$$

Where:

- S is the sharpness measure.
- d represents each detail coefficient matrix from the Discrete Wavelet Transform (DWT).
- A is the approximation coefficient matrix from the DWT.

To quantify the sharpness of an image, we focus specifically on the energy associated with the detail coefficients, deliberately excluding the approximation coefficients from this assessment. Energy is defined as the sum of the absolute values of the

detail coefficients. This approach is grounded in the understanding that sharpness is inherently related to the presence of high-frequency information within an image [29]. By computing the wavelet energy, we gain insight into the extent of detail present, which directly correlates with the perceived sharpness of the image.
A wavelet energy to the total energy in the approximation coefficient ratio is calculated as the sharpness score. This ratio gives us a fair and standardized measure of sharpness and is used for comparing different images or different processing techniques. A sharper score indicates a more defined presence of fine details compared with the overall structure and brings the image as detailed and sharp.

DCP with Guided Filters

The basic idea of DCP is that in most natural images, at least one color channel (R, G, or B) is low in intensity in regions with non-sky content [3]. The atmospheric light and transmission map of the image are estimated by computing the dark channel as the minimum intensity across the RGB channels.
For an image I with pixel values $I(x, y)$ in three channels (R, G, B), the dark channel $D(x, y)$ is defined as:

$$D(x,y) = \min_{c \in \{R,G,B\}} (I_c(x,y)) \tag{3.3}$$

To enhance the dark channel, a morphological operation (erosion) is typically applied using a rectangular kernel of size size:

$$D_{dark}(x,y) = \text{erode}(D(x,y), \text{kernel}) \tag{3.4}$$

The transmission map represents the scene's transparency based on the dark channel. It is calculated using the formula:

$$T(x,y) = 1 - \omega \cdot \frac{D_{dark}(x,y)}{\max(D_{dark})} \tag{3.5}$$

Here, ω is a parameter (usually between 0 and 1) that controls the transmission.
The atmospheric light is estimated by finding the brightest pixels in the dark channel. The atmospheric light A can be computed as:

$$A = \text{mean}(I(x,y)) \text{ for the brightest pixels in } D_{dark} \tag{3.6}$$

To refine the transmission map, a guided filter is used, which employs the input image I as guidance to compute an output q:

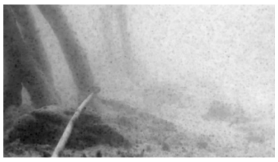
$$q = \alpha \cdot I + b \tag{3.7}$$

Where α and b are derived from the statistics of the transmission map and the guidance image.
To improve the quality of the transmission map, a guided filter reduces noise and preserves edges. The scene is then reconstructed using the refined transmission and atmospheric light:

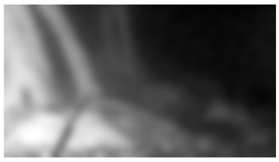
$$J(x,y) = \frac{I(x,y) - A}{T(x,y)} + A \tag{3.8}$$

This formula recovers the dehazed image J by adjusting the pixel values based on the estimated transmission.
These steps are all combined and produce the final dehazed of the image. This approach improves visibility and detail in hazy underwater images by leveraging both dark channel information and guided filtering for refining the transmission map, and is a powerful technique in image processing. **Figure 3.6** shows the original image (a), the image after the dark channel is computed (b), the image after the refined transmission is applied (c), and the final dehazed image (d).
Figure 3.7 shows comparison of histogram between the original image(a) and dehazed image(b) and as we can see the histogram is well spread in the dehazed image.

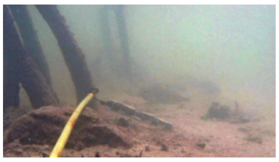
(a) Original Image



(b) Dark Channel

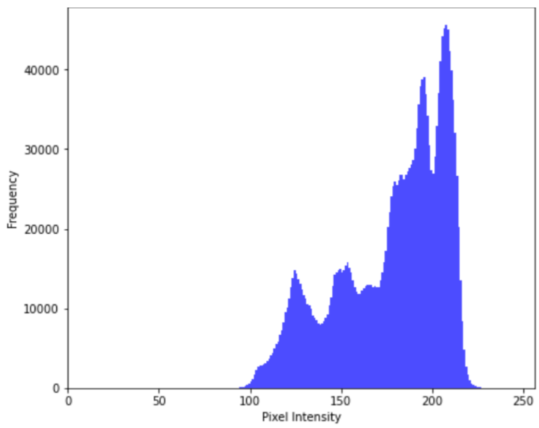


(c) Refined transmission

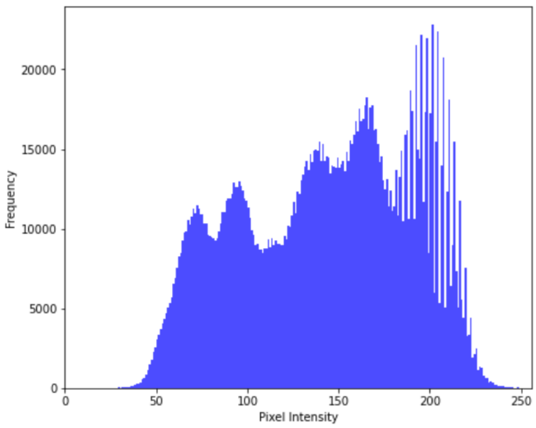


(d) Dehazed Image

Figure 3.6: DCP with Guided Filter



(a) Original Image Histogram



(b) Dehazed Image Histogram

Figure 3.7: Histogram Comparison After Dehazing

Unsharp Mask

To sharpen an image, an unsharp masking technique is used for the manipulation of frequency components of an image. It starts with a blurred version of the original image generated using a Gaussian filter. This blurred image can be mathematically represented as B, where the original image I is convolved with a Gaussian kernel G_σ of standard deviation σ:

$$B(x, y) = G_\sigma * I(x, y) \tag{3.9}$$

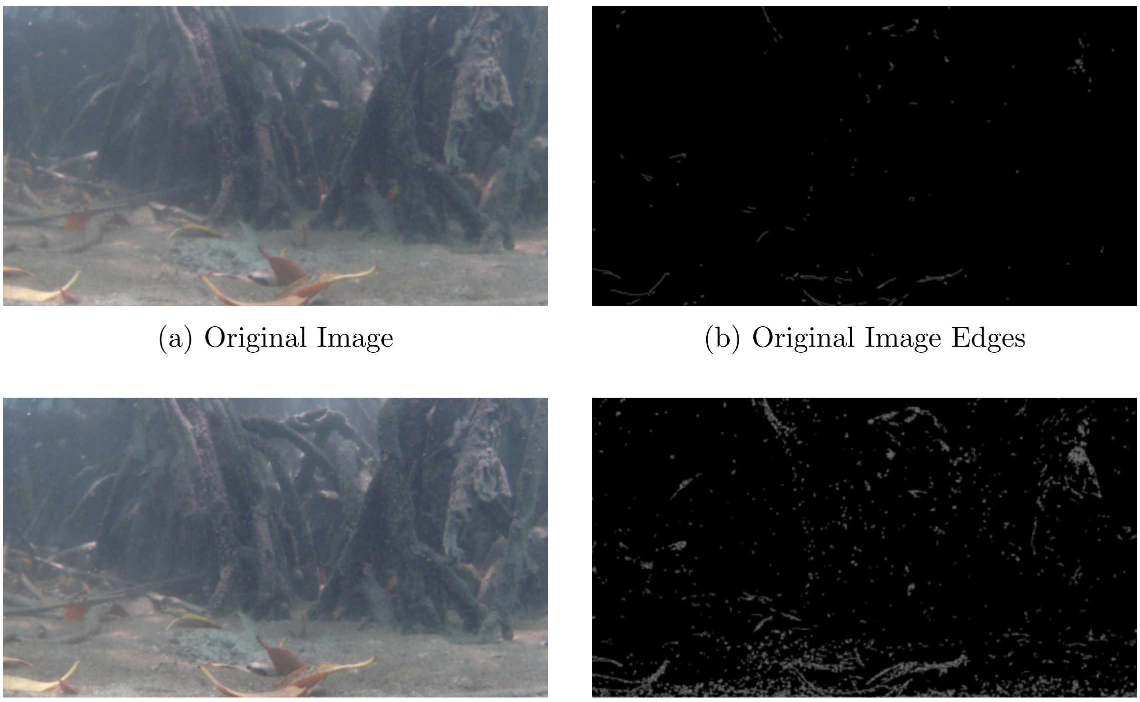
In this equation, $*$ denotes the convolution operation, and the parameter σ controls the degree of blurring, with larger values resulting in more significant smoothing of the image.
An unsharp mask M is computed by subtracting the blurred image from the original, which highlights the edges and fine details that have been diminished due to the blurring:

$$M(x, y) = I(x, y) - B(x, y) \tag{3.10}$$

The final step involves creating the sharpened image S by adding a scaled version of this unsharp mask back to the original image. This is mathematically expressed as:

$$S(x, y) = I(x, y) + \lambda \cdot M(x, y) \tag{3.11}$$

In this formula, λ represents the strength of the sharpening effect.
We can see in the **Figure 3.9** that the edges of the unsharp mask are far more prominent than the original image, making it sharper.



(a) Original Image

(b) Original Image Edges

(c) Sharpened Image

(d) Sharpened Image Edges

Figure 3.8: Unshap Masking

Bilateral Filter

By applying the non-linear, edge-preserving, and noise-reducing smoothing filter to images, to perform tasks such as edge reduction while preserving critical image details, the bilateral filter can be used. The filter takes into account the spatial proximity of pixels together with a similarity in the color values of such pixels [32]. The filter can be defined as follows:

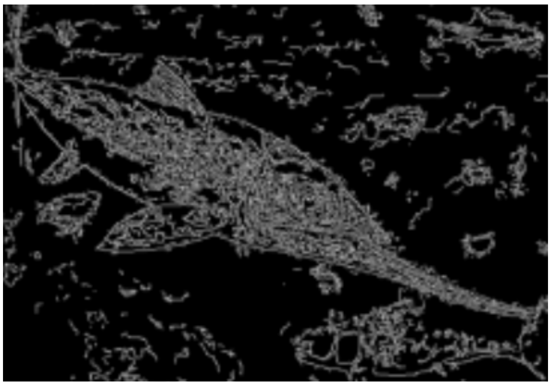
$$I_{\text{filtered}}(x) = \frac{1}{W_p} \sum_{y \in \Omega} I(y) \cdot \exp\left(-\frac{\|x - y\|^2}{2\sigma_s^2}\right) \cdot \exp\left(-\frac{\|I(x) - I(y)\|^2}{2\sigma_r^2}\right) \tag{3.12}$$

where:

- $I_{\text{filtered}}(x)$: the filtered intensity value at pixel location x.
- $I(y)$: the intensity of pixel y.
- Ω: the neighborhood of pixel x.
- W_p: a normalization factor to ensure that the weights sum to 1:
$$W_p = \sum_{y \in \Omega} \exp\left(-\frac{\|x - y\|^2}{2\sigma_s^2}\right) \cdot \exp\left(-\frac{\|I(x) - I(y)\|^2}{2\sigma_r^2}\right)$$
- $\|x - y\|$: the spatial distance between pixels x and y.
- $\|I(x) - I(y)\|$: the intensity difference between pixels x and y.
- σ_s: controls the spatial extent of the filter (how far the influence of nearby pixels extends).
- σ_r: controls the influence of intensity differences (how similar in color values the neighboring pixels must be).

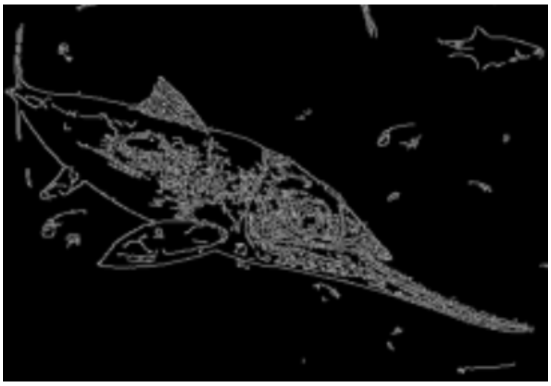
The spatial proximity and the similar intensity of the pixels are considered in applying weights in the case of the bilateral filter. This decreases the contribution of distant, or dissimilar pixels in the case of exponential terms. W_p ensures that the output value is normalized; otherwise, the filter will affect the overall brightness of the image. **Figure 3.9** show the edge difference between an overly sharpened image and an image after applying the filter.

(a) Original Image



(b) Original Image Edges

(c) Filtered Image



(d) Filtered Image Edges

Figure 3.9: Bilateral Filtering Edges

3.2.3 WAV-DUC Algorithm

WAV-DUC (Wavelet-based Advanced Vision - Dehazing and Unsharp Masking) is a sophisticated image enhancement algorithm developed to adaptively improve input images through several target processing steps. The algorithm starts with an initial contrast assessment as low contrast levels may severely degrade the effectiveness of following image enhancement techniques, especially the Dark Channel Prior (DCP) method. The algorithm prioritizes contrast enhancement so that subsequent processing stages are set up for the best performance.

Once the contrast is restored, the algorithm then checks if the image contains haze. It is crucial for such evaluation as haze can blur image details and cause visual degradation. The algorithm uses a Guided Dehazing method to dehaze an image, and thus effectively mitigates its effects to recover the clarity and depth of the image. After dehazing, the enhanced image is assessed using a discrete wavelet transform (DWT) based sharpness score. Sharpness is a critical partial factor in overall image quality. If the sharpness is low then the algorithm applies unsharp masking on to enhance the image clarity and image sharpness. It is an upgrade to the visible quality so that important elements become more discernible.

The WAV-DUC algorithm is recursive so that it can continually improve the image at each step of processing. For each enhancement, the algorithm reevaluates the image and guarantees that the criteria of contrast, haze, and sharpness meet all specified. This iterative process results in an adaptive enhancement workflow in which the image goes through repeated processing until all the quality metrics hit the thresholds specified.

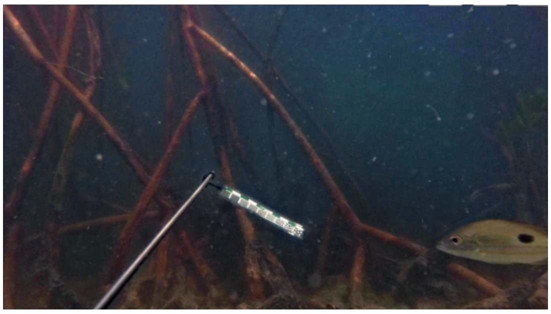
WAV-DUC further augments the algorithm's performance by parallel thread processing, allowing the execution of multiple tasks in parallel. The parallelization greatly reduces execution time, and the algorithm becomes more efficient and suitable for edge devices. Iterative refinement and parallel processing are combined to position WAV-DUC as a robust solution.

Threshold Selection

We adopted an empirical approach to threshold selection in our study, as no formula or literature exists for threshold based image enhancement. We determined this threshold through a trial and error process. We tested the selected threshold on two distinct datasets: We used the Brackish Underwater Dataset (14,674 images) [25] and the DeepFish Dataset (4,605 images) [23] as we can see in the sample in **Figure 3.11** and **Figure 3.10**. The depth levels, underwater scenarios, lighting conditions, and haziness levels of these datasets are significantly different, which enables us to evaluate the robustness of our method under different conditions.

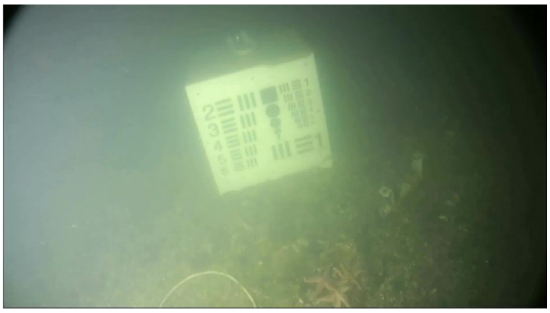
The results were optimal on both datasets and will be presented in the result analysis section with further metric analysis. Furthermore, when the selected thresholds were applied to high quality images as in **Figure 3.12**, the algorithm did not enhance them, preserving the actual image. This shows that our algorithm is able to distinguish between images that require enhancement and those that do not.

(a) Original Image



(b) Enhanced Image

Figure 3.10: WAV-DUC applied on DeepFish dataset sample



(a) Original Image

(b) Enhanced Image

Figure 3.11: WAV-DUC applied on Brackish Underwater dataset sample

(a) Original Image

(b) Enhanced Image

Figure 3.12: WAV-DUC applied on an optimal image

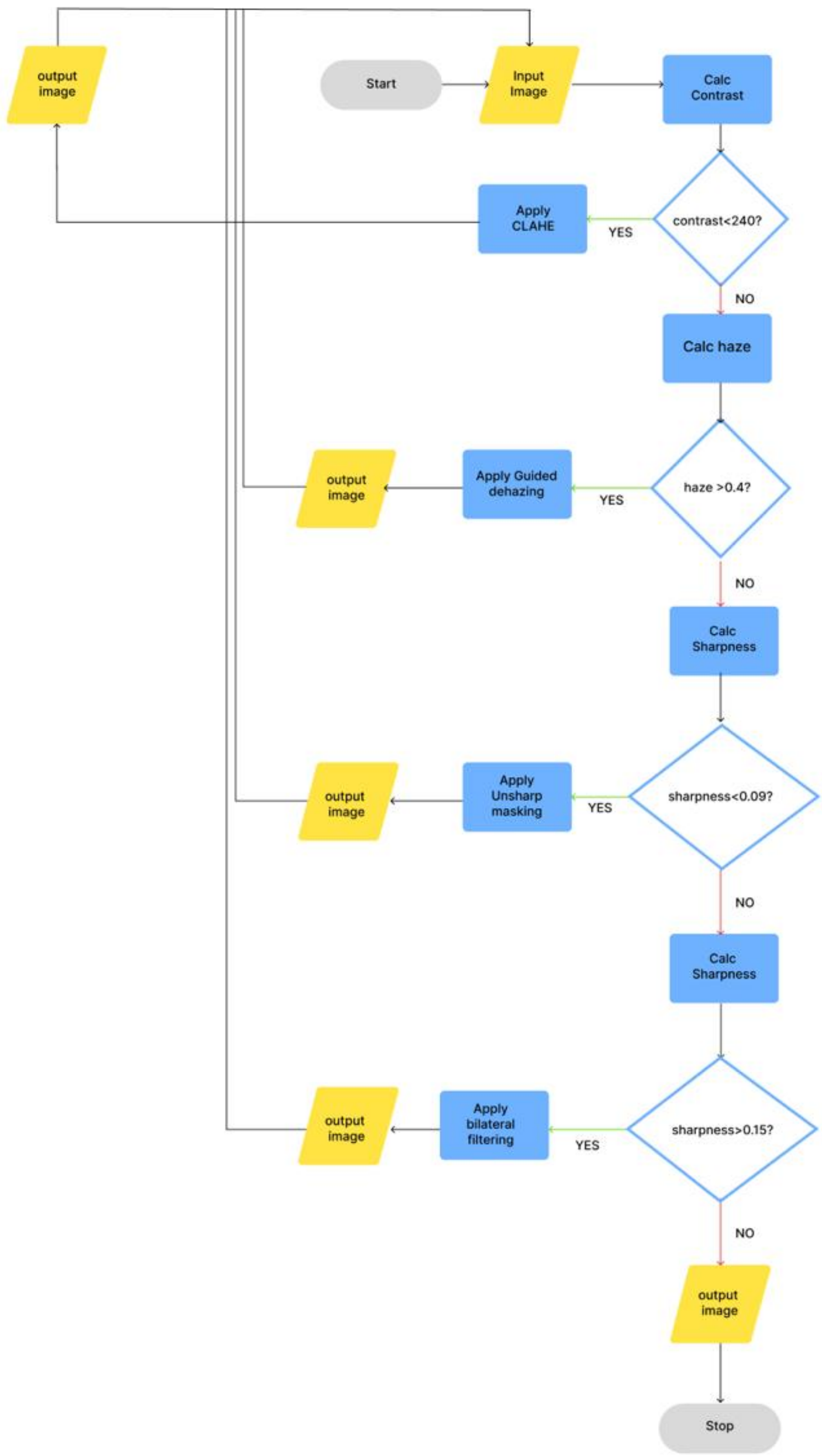


Figure 3.13: Algorithm Flowchart

Complexity Analysis

The algorithm WAV-DUC iteratively improves underwater images by enhancing contrast, reducing haze, and sharpening details. The image quality is improved at each step in a way that is done iteratively so that steps contribute to the progressive improvement in image quality.
Contrast Score is calculated with a time complexity of $O(m \times n)$. Local contrast enhancement is applied if needed, also at $O(n \times m)$, using CLAHE.
Subsequently, the algorithm computes a haze score ($O(n \times m)$), and if needed, it applies dehazing, whose cost is $O(n \times m)$ or higher. The wavelet-based methods are then used to evaluate sharpness with a complexity of $O(n \times m \log(n \times m))$. If necessary, adjustments, including bilateral filtering (i.e., $O(n \times m \times s^2)$), are applied. Since the algorithm iteratively refines the image, the overall time complexity is:

$$O(k \cdot n \times m \cdot \max(s^2, \log(n \times m))) \tag{3.13}$$

This assumes that we have k iterations and maximum number of iterations has been locked to 10 in order to tackle over enhancement and infinite loop.

3.3 Model Explanation

3.3.1 YOLOv5

Yolov5 plays the role of feature extraction from the image input. It employs the CSPDarknet53 that alleviates the limitations of conventional darknet-53, a convolutional neural network. It is meant for expressing spatial relationships and is able to deal with various features such as the edge, texture, and shape at various levels of abstraction. The effectiveness of the CSP layers to advance the gradient flow and lessen the computational load of the model at the same time ensures high accuracy. This is because YOLOv5 optimizes the scales in detecting objects more effectively than before. This enables it to have a Path Aggregation Network (PAN) to allow for an integration of feature fusion from the different levels in the backbone network. YOLOv5 also employs anchors for object detection, this means that YOLOv5 predicts the offsets of the bounding box to default or anchor boxes. Some of these predictions include; the position of the object, size of the object, and class confidence [14].

3.3.2 YOLOv8

The Yolov8 parses low-level to high level features from an image input that we have fed to the model. It presents a new CSPDarknet structure that enhances feature extraction compared to the prior method of searching for features with Cross Stage Partial (CSP) connections. CSP connections enhance the learning capability by dividing the feature map into two parts and passing through a series of residual blocks and then joining the divided part. There is also the Spatial Pyramid Pooling (SPP) module to achieve better descriptions of objects over different scales, which is particularly effective when detecting regularly scattered objects of different sizes. The detection in YOLOv8 eliminates predefined anchor boxes, thus making it an anchor-free method as compared to previous versions. YOLOv8, on the other hand, directly gives the center of the object, width, height, and the class of the object which can easily adjust the ways through which smaller objects will be captured by it [31].

3.3.3 YOLOv10

The proposed YOLOv10 model is introduced as a hybrid backbone with CNNs and Vision Transformers, addressing the weak-feature-extraction and weak-global-context difficulties in detecting small and complex objects. It increases feature fusion in Feature Pyramid Networks (FPN) or Path Aggregation Networks (PAN); it also adds improved multi-scale representation. YOLOv10 therefore uses an anchor-free detection mechanism which was found to be more flexible than the anchor boxes that define the network in other models. Originally, its lightweight design empowers the model across mobiles and edge devices via pruning, quantization, and knowledge distillation. The architecture also opens a door to self-supervised learning to use the unlabeled data, which in turns makes the architecture more suitable for Medical, Satellite, and underwater imagery domains. With superior real-time performance and multi-tasking features, YOLOv10 enhances its features to object detection, tracking, segmentation, and pose estimation capabilities; further enhances the data augmentation method to accommodate environmental constraints [37].

3.3.4 YOLOv11

The YOLOv11 architecture brings new features intended to enhance object detection accuracy and speed at the same time. It uses an enriched backbone which consists of CNN and transformer to capture both local and global features that will aid detection in complex environments. A larger detection head enables object detection at different scales thus improving the performance of the model at different aspect ratios. The YOLOv11 also improves the detection of objects by eliminating anchor boxes: again, it has an anchor-free detection system, which makes it easier to detect the shapes and sizes of objects. One of the novelties of the presented model is the use of self-supervised learning; this way, the model can learn from the illustration without a text, which increases its generalization with a small amount of annotated data. Furthermore, YOLOv11 also enables Multi-Tasking for tasks such as segmentation, tracking, and 3D detection. It is best suited for live streaming of videos, which makes it applicable in areas such as self-driving vehicles and/or security. In addition, YOLOv11 is lightweight and can operate on mobile and edge devices so that fast inference without compromising the performance is achieved [33].

3.4 Workflow

We initiated our research by gathering data from two distinct sources: There are two datasets which include the BRACKISH Underwater Dataset[25] and the DeepFish Dataset[23]. After data collection, we developed a preprocessing phase that was extensive enough in order to guarantee the quality and appropriateness of the data for the next stages of analysis. After preprocessing the data, Wav-DUC was applied on the data to created a new enhanced dataset.
In order to test the proposed enhancement approach, four different YOLO models were trained on the presented enhanced datasets.
We additionally performed a validation of the entire pipeline across a range of devices, particularly edge devices like Raspberry Pi. To carry out this goal, we devised this evaluation to compare the inference times and test the viability of integrating our models in real-world settings which are normally characterized by limited resources. By following this systematic approach, we seek to demonstrate reliability and strength regarding the proposed algorithm in improving underwater image quality for object detection purposes.

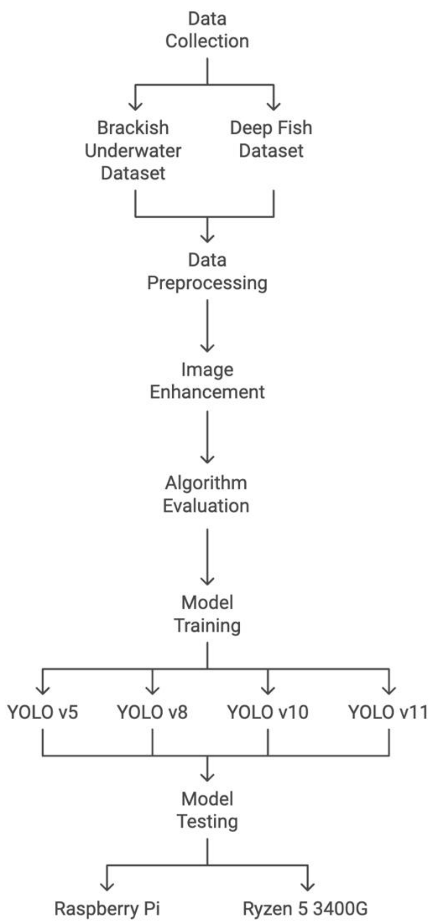


Figure 3.14: Workflow Diagram

Chapter 4

Experiment and Result Analysis

4.1 Algorithm Evaluation and Comparison

The performance image enhancement algorithms are commonly evaluated by Structural Similarity Index (SSIM), Peak Signal-to-Noise Ratio (PSNR), and Underwater Image Quality Measure (UIQM).
The SSIM measures the similarity between two images subjecting to variations in structural information, luminance, and contrast [34]. The reason behind its use is its suitability for image quality assessment as it reveals how the human visual system perceives image quality. The formula for SSIM is as follows:

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{4.1}$$

where μ_x and μ_y are the means of images x and y, σ_x^2 and σ_y^2 are the variances, σ_{xy} is the covariance, and C_1 and C_2 are constants to stabilize the division.
PSNR is a measure of image quality in comparison to its reference and quantifies how close the signal power (image) is to the power of the corrupting noise. It's measured in decibels (dB) based on the Mean Squared Error (MSE) between the reference and the enhanced image [12]. The formula is as follows:

$$\mathrm{PSNR} = 10 \cdot \log_{10}\left(\frac{L^2}{\mathrm{MSE}}\right) \tag{4.2}$$

where L is the maximum pixel value (usually 255 for an 8-bit image) and MSE is calculated as:

$$\mathrm{MSE} = \frac{1}{MN}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}[x(i,j) - y(i,j)]^2 \tag{4.3}$$

for an image of dimensions $M \times N$ where x and y represent the reference and enhanced images, respectively.
The objective of UIQM is to evaluate the visual quality of underwater images taking into account colorfulness, sharpness, and contrast. It is a weighted combination of three sub-components: Underwater Image Colorfulness Measure (UICM), Underwater Image Sharpness Measure (UISM), and Underwater [24]. It is generally expressed as:

$$\text{UIQM} = \alpha \cdot \text{UICM} + \beta \cdot \text{UISM} + \gamma \cdot \text{UIConM} \tag{4.4}$$

where α, β, and γ are weighting coefficients.

4.1.1 Quantitative Comparison

Referential

First, we conduct a full-reference evaluation which includes the most common metrics, including SSIM and PSNR. These metrics are chosen because they hold the advantage of easy comparison and highly accurate benchmarking. For example, SSIM is used to compare perceived image quality while PSNR is used to compare the degree of pixel distortion. Altogether, they offer a solid set of parameters for precise benchmarking and, more specifically, for comparing one image enhancement model to another. With regards to their correlation, a higher PSNR value means the raw image is closer to the reference image in terms of pixel difference while a higher SSIM values mean structural similarity between images. Based on the benchmark dataset[2] and UIEB dataset[17], a quantitative comparison against the existing models is made in **Table 4.2** and **Table 4.1** to contrast our proposed Wav-Duc model. From the comparison in table **Table 4.2**, it is clear that incorporating our proposed Wave-Duc model yields the highest SSIM score and follows it closely in PSNR. Consequently, other algorithms cannot offer reasonable results as compared to both presented metrics. Also, one more interesting fact is that the performance indicators in our proposed model and other algorithms contradict each other. Despite a lower PSNR of 18.1, a higher SSIM of 0.86 has been obtained illustrating the better perceptual quality and structural similarity with the reference image.

On the other hand, other models that indicate higher PSNR values also result in higher scores on the SSIM scale. This discrepancy may have been caused by the failure to incorporate color correction algorithms in our model. Hence, in our model we retain the overall construction and the way it looks over the pixel density of the image.

As a result, although PSNR can work as an effective pixel accuracy measure, it might not be sufficient enough to describe a model's ability to produce images that look good to the human eye, as has been demonstrated in our case. We further compare our results against other models based on PSNR and SSIM for the UIEB dataset. From the comparison in **Table 4.1**, it is clear that incorporating our proposed model yields the second-highest PSNR score compared to other Deep learning models. Moreover, Wav-Duc comes up in second in terms of PSNR having a PSNR difference of 0.3 against SCNet[38] and an SSIM difference of 0.0267 against Espinosa [7]. Moreover, it is imperative to note that the Wave-Duc algorithm is classified as a classic model due to not using any deep learning architectures. On the contrary, SCNET[38] and Espinosa[7] use computationally complex deep learning models that yield higher PSNR and SSIM at the cost of computational power. However, Wav-Duc manages to yield close PSNR and SSIM scores to SCNET[38] and Espinosa[7] while maintaining significantly less resources. Therefore, it is evident from PSNR and SSIM scores that our classic Wave-Duc model manages to obtain structurally close images to SCNET[38] and Espinosa[7]. Furthermore, **Figure 4.1** illustrates the raw input and reference image against, IBLA[28], Fusion[6], GLCHE[9], DWG[22], Water-Net[2], Deep WaveNet[21], Espinosa[7], Espinosa + db[7], Wave-DUC.

Also, **Figure 4.2** show the comparison of different algorithms, Fusion-based[4], retinex-based[8], histogram prior[15], blurriness based[27], water CycleGan[16], DenseGAN[11], Water-net[2] and ours (Wav-DUC).

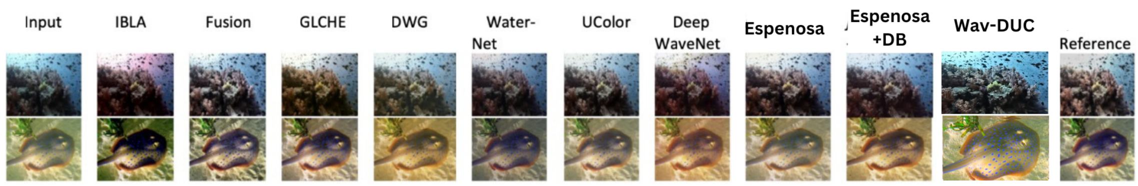


Figure 4.1: Different Algorithms on UIEB Dataset [17]

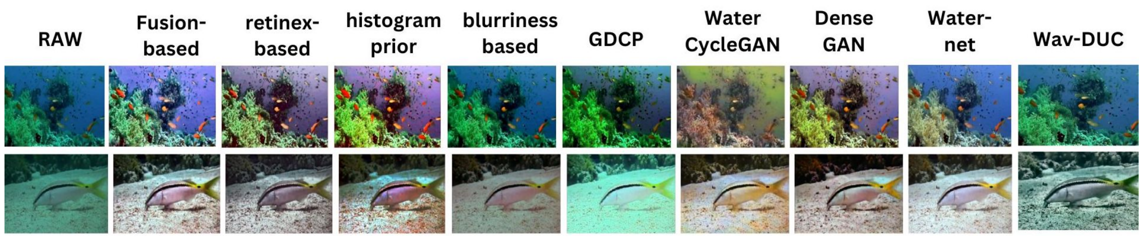


Figure 4.2: Different Algorithms on Benchmark Challenge Dataset [2]

Non-Referential

We use the UIQM metric for non-reference evaluation. The metric is selected so that the image can be judged based on its contrast, color, and sharpness. Furthermore,

Method	PSNR↑	SSIM↑	Type
IBLA [28]	14.3856	0.4299	Classic
Fusion [6]	21.1849	0.8222	Classic
GLCHE [9]	21.027	0.8487	Mix
DWG [22]	19.6727	0.8614	Deep
Water-Net [2]	19.3134	0.8303	Deep
Ucolor [5]	20.63	0.77	Deep
SCNet [38]	22.08	0.8625	Deep
Deep WaveNet [21]	21.57	0.8	Deep
Espinosa et el.[7]	20.8694	0.8703	Deep
Espinosa et el. +DB [7]	20.9226	0.8802	Deep
Wav-DUC (Ours)	**21.78**	**0.8535**	Classic

Table 4.1: Comparison of underwater image enhancement methods based on PSNR and SSIM on UIEB dataset [17]

Method	PSNR↑	SSIM↑	Type
Wav-DUC (Ours)	**18.1**	**0.86**	Classic
Fusion-based [4]	17.61	0.61	Classic
GDCP [26]	12.10	0.51	Classic
Histogram Prior [15]	15.82	0.54	Classic
Blurriness-based [27]	15.32	0.60	Classic
Water CycleGAN [16]	15.75	0.52	Deep
Dense GAN [11]	17.28	0.44	Deep
Water-Net [2]	19.11	0.80	Deep

Table 4.2: Comparison of different methods based on PSNR and SSIM values.

due to the absence of ground-truth images, it is hard to assess the quality only based on reference images provided in the benchmark dataset. Using the UIQM metric, a quantitative comparison is established between our Wav-DUC model and other pre-existing image enhancement models based on their benchmark dataset [2], illustrated in **Table 4.3**. Although our model provides better UIQM scores it is to be noted that there are discrepancies in the UIQM metric as images having higher UIQM do not always translate into higher perceptibility as illustrated in Insert broken dive+ and histogram prior image. Therefore, although UIQM is a valid image assessment metric, it is limited in its capability when assessing underwater images. [2]

Method	UIQM
Histogram Prior[15]	1.07
dive+ [2]	0.92
Wav-DUC	1.29

Table 4.3: Comparison of UIQM scores for different methods.

4.1.2 Qualitative Analysis

We select one image that activates all the steps of the iterative algorithm, allowing us to visualize and judge each step qualitatively. Furthermore, each step is activated based on a qualitative threshold set on the clarity scores. The effects of each of the iterative enhancements are illustrated along **Figure 4.3**. Note that this image does not account for all images in the benchmark dataset.

Contrast Enhanced: Subjective comparison on raw image (a) and contrast-enhanced image (b) having a contrast score difference of 8 units from left to right.

Haze Removal: Subjective comparison on raw image (a) and haze adjusted image (c) having reduced haze by 0.31 units from left to right.

Sharpness Adjustment: Subjective comparison on raw image (a) and sharpness adjusted image (d) having increased sharpness by 0.16 units.

(a)

(b)

(c)

(d)

Figure 4.3: Qulitative Comparison

4.2 Model Evaluation and Analysis

In Tab 4.3, the results of YOLO series models on the brackish dataset are presented in terms of mAP@50, mAP@50-95, precision, recall, and inference time. The mAP@50 also emphasizes that the YOLOv5 model provides the highest score of 0.985 to identify objects in the dataset. All variants of YOLO including YOLOv11 and YOLOv8 perform very well in different detection cases with 0.983 and 0.981 mAP respectively followed by YOLOv10 with 0.980 mAP. With mAP@50-95 that measures detection accuracy based on intersection over union (IoU) thresholds, YOLOv10 is still dominant with a score of 0.821 for all tiers of difficulties, while the scores of YOLOv5, YOLOv8, and YOLOv11 are 0.817, 0.815 and 0.818 correspondingly. YOLOv11 is the most precise model that possesses the highest score, 0.981 – with the lowest rate of false positives, best suited for situations where misidentification is crucial. Recall, on the other hand, reveals that YOLOv5 (0.971) outperforms the other models in true positive detection.

In Table 4.4, we can see the result of the deepfish dataset. mAP@50 proves that YOLOv8 has a high accuracy of 0.969, accompanied by YOLOv5 at 0.966. YOLOv10 has the smallest mAP@50 of 0.956 and can be considered relatively precise but less accurate in comparison to the other models. Similar to mAP@50-95, YOLOv8 is also the best performing at 0.695 for all the IoU thresholds. YOLOv5 has the best precision of 0.953, therefore indicating that it detects fewer wrong objects while YOLOv8 is the best at recall of 0.935 meaning that it will detect all the relevant objects. YOLOv10 has comparatively less accuracy and much less recall implying that one detects objects less at times or generates more false alarms.

Model	**mAP@50**	**mAP@50-95**	**Precision (P)**	**Recall (R)**
YOLOv5	0.966	0.690	0.953	0.928
YOLOv8	0.969	0.695	0.952	0.935
YOLOv10	0.956	0.681	0.929	0.919
YOLOv11	0.962	0.680	0.944	0.926

Table 4.4: Performance comparison of YOLO models on mAP, precision, and recall on Brackish underwater Dataset

Model	**mAP@50**	**mAP@50-95**	**Precision (P)**	**Recall (R)**
YOLOv5	0.985	0.817	0.979	0.971
YOLOv8	0.981	0.815	0.976	0.963
YOLOv10	0.980	0.821	0.966	0.963
YOLOv11	0.983	0.818	0.981	0.963

Table 4.5: Performance comparison of YOLO models on mAP, precision, and recall on DeepFish dataset

The **Figure 4.4** and **Figure 4.5** show the training and validation loss trends for YOLO v5, v8, v10, and v11 models on the Brackish Underwater datasets and DeepFish dataset for 80 epochs. Losses for both datasets were seen to consistently fall, showing improved learning and a better model performance.
In conclusion, YOLOv8 is convenient to apply in real-time object detection more than the other models because of the good obviousness, but slight delay time, however, it depends on the potential outcome whether the application could accept false negatives and provide a higher recall rate or not.

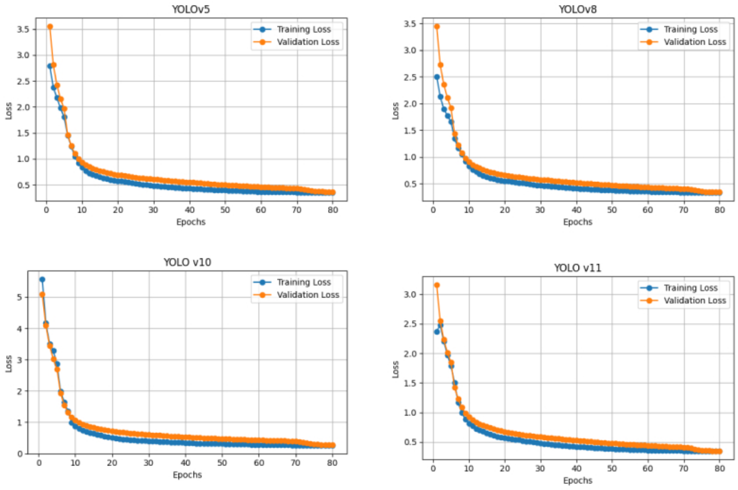


Figure 4.4: Training and Validation loss on BRACKISH dataset

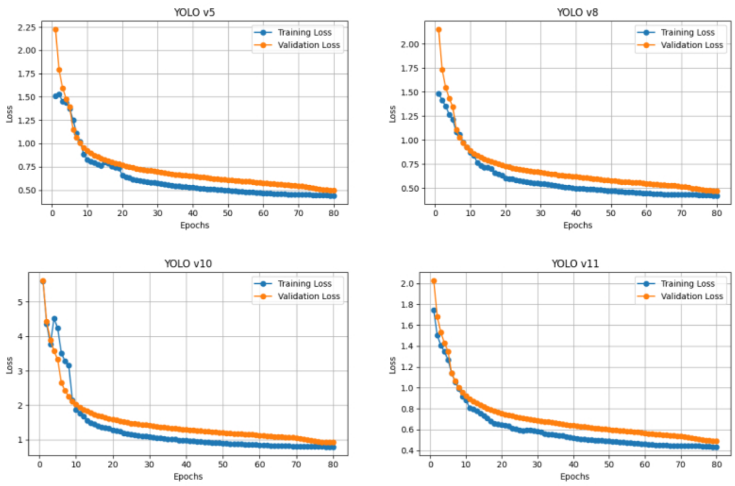


Figure 4.5: Training and Validation loss on DeepFish dataset

4.3 Experimental Analysis

Training Setup

Training and evaluating the model was done on an AMD Ryzen 5 3400G CPU at 4×8 with a base frequency of 3.7 GHz with 16 GB DDR4 RAM and an NVIDIA GTX 1660 SUPER graphics card with 6 GB VRAM. With CUDA 12.1 & cuDNN 8.9.0 for effective deep learning computation, the setup used Pytorch as the backend, and the GPU supported both training and inference to aid in computations.

Testing Setup

We have used a Raspberry Pi 3 for testing. The Raspberry Pi 3 has a 64-bit 2.4 GHz quad-core ARM Cortex-A76.

4.3.1 Experimental Result Analysis

We have performed an experiment where images from the dataset are enhanced and then fed to YOLO models for object detection. Predicted bounding boxes are stored and logs are generated, simulating underwater data collection. However, improved image clarity obtained via enhancement may not affect the detection results when using raw and enhanced images, as the former is intended for consistency in detection results between them, and the latter is created to better analyze and visualize the images. This simulation gives us valuable insights towards our ultimate goal of processing real time data on hardware such as the Jetson platform.
The timing results show that inference times on the Raspberry Pi 3 are much longer than on the Ryzen 5 3400G, but the Wav-DUC algorithm's run time is still impressive on the Pi. The algorithm's median execution time is 64.78 ms ms on the Raspberry Pi 3 and 21 ms on the Ryzen as can be seen on **Figure 4.6**. The execution time of this algorithm is sub 70 ms, which demonstrates that the algorithm is lightweight enough to run effectively on resource-constrained devices. On the other hand, YOLO inference times on the Pi are 978.20 ms to 1232.27 ms, much higher than the 76-86 ms on the Ryzen as can be seen in **Figure 4.7**. To better represent the data, we use the median score, which minimizes the impact of outliers and provides a more accurate representation of typical performance, especially on devices like the Raspberry Pi.
In conclusion, the Raspberry Pi 3 shows good performance for the Wav-DUC algorithm, with a median execution time of 64ms, which is highly satisfactory for an edge device. However, the significantly higher inference times for object detection indicate that real-time processing will require more powerful hardware, such as the Jetson platform. The results of this experiment validate the effectiveness of the Wav-DUC algorithm and YOLO models for underwater data collection, and future improvements and hardware optimizations are expected to improve real-time performance.

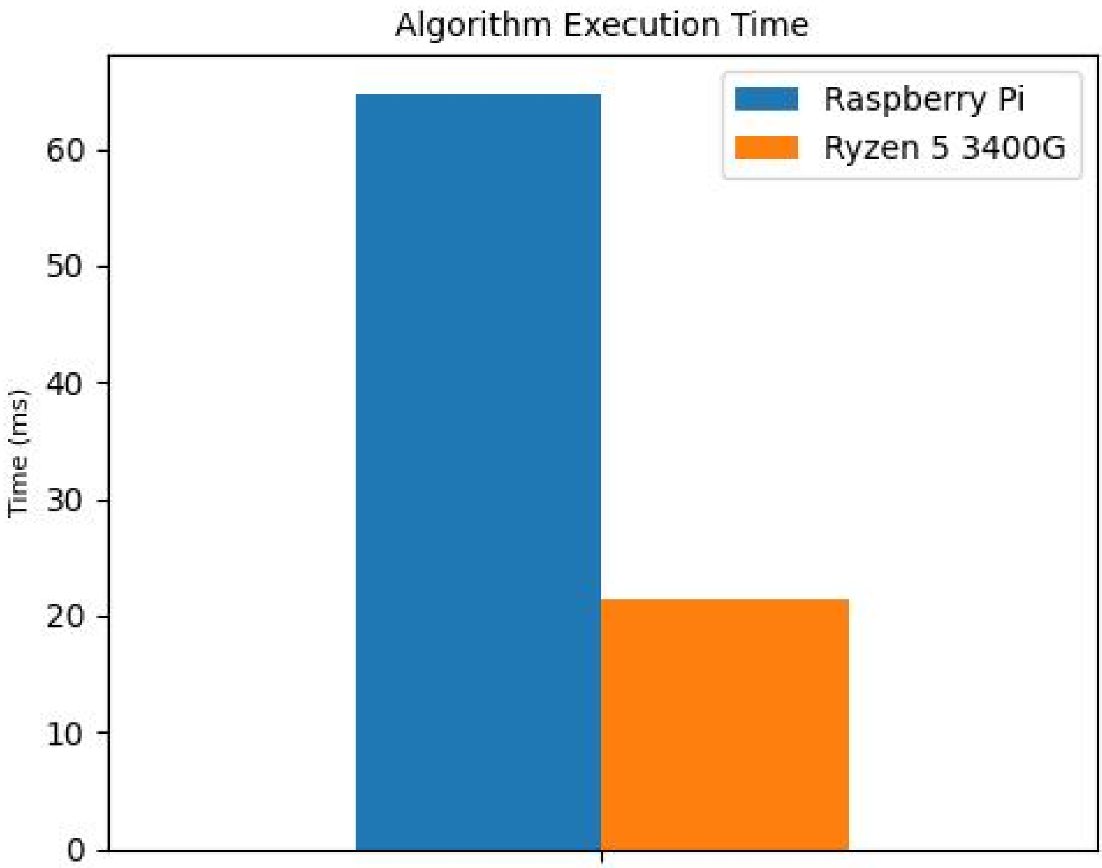


Figure 4.6: Average Execution Time Comparison

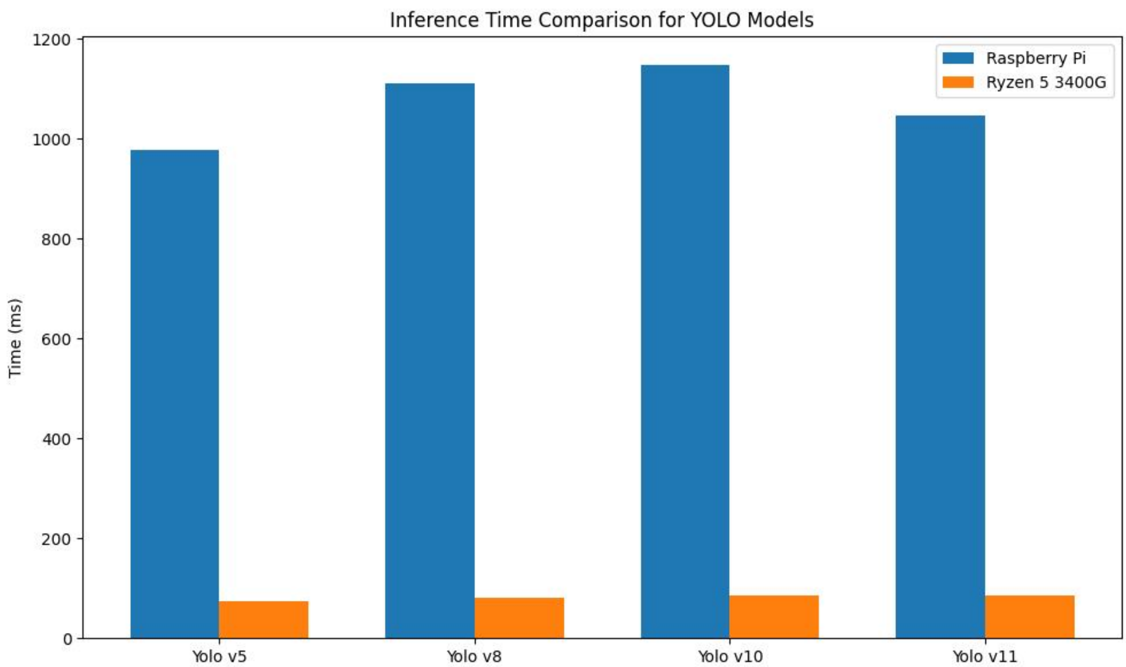


Figure 4.7: Average Inference Time Comparison

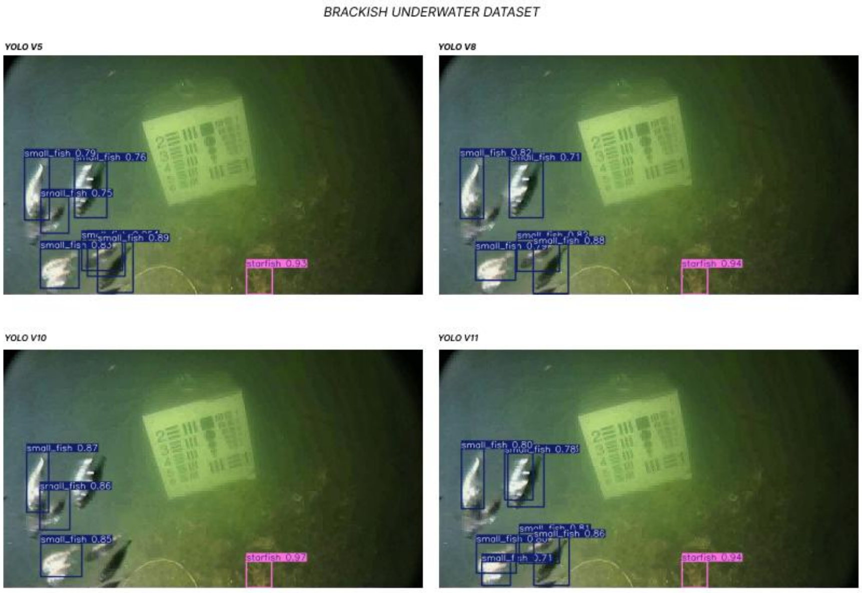


Figure 4.8: Prediction on Brackish Underwater Dataset

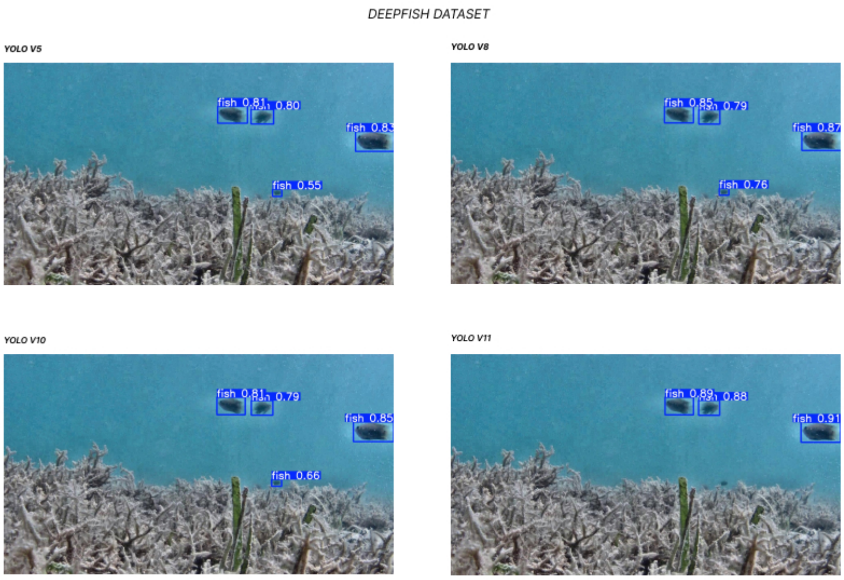


Figure 4.9: Prediction on Brackish DeepFish Dataset

Chapter 5

Future Work

In the future, we will enhance our underwater image enhancement algorithm by integrating more advanced color correction techniques to better counteract the color cast caused by water absorption and scattering, and thus restore the natural colors of underwater scenes. We also wish to further optimize the computational efficiency of the algorithm to reduce the time it takes. The image-processing pipeline will be streamlined and key image-processing steps will be implemented more efficiently. We will further reduce the computational load by quantization and model pruning, which will make the algorithm more suitable for deployment on edge devices such as Raspberry Pi. Throughout this optimization, we will work to preserve image quality, to keep the metrics such as Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), and Underwater Image Quality Measure (UIQM) very close to their original values. With these improvements, our improved algorithm can provide real-time, high-quality image enhancement in various underwater environments, making it more robust and adaptable for real-world use.

Chapter 6

Conclusion

In this work, we introduced a new Underwater Image Enhancement Algorithm to solve the difficult problems of poor visibility conditions and fluctuating environmental factors. When comparing the results of the proposed methods with the previous techniques, the proposed approach enhances the performance of underwater image enhancement and produces more clear and distinct images. Although this algorithm failed to produce better images than some neural network-based enhancement models, it is still promising as the metrics were very competitive The enhancement ability of the algorithm is dynamic for a variety of underwater scenes so that the performance can be optimal under any conditions.

The advantage of the proposed technique was confirmed by the experimental, acquiring an average Peak Signal-to-Noise Ratio (PSNR) of 21.78; Structural Similarity Index Measure (SSIM), 0.8535 and Underwater Image Quality Measure (UIQM), 1.29. Also, our proposed algorithm can reduce the average response time to 64,78ms, which makes the algorithm efficient enough for real-time applications. We also validated the whole pipeline using a Raspberry Pi – an edge device – revealing the algorithms's feasibility in low-resource settings.

Bibliography

[1] Object detection using transfer learning for underwater robot. In *IEEE Conference Publication*, August 2020.

[2] An underwater image enhancement benchmark dataset and beyond. *IEEE Journals & Magazine*, 2020.

[3] Chapter 14 - a wavelet-based low frequency prior for single-image dehazing. In Siddhartha Bhattacharyya, Paramartha Dutta, Debabrata Samanta, Anirban Mukherjee, and Indrajit Pan, editors, *Recent Trends in Computational Intelligence Enabled Research*, pages 245–262. Academic Press, 2021.

[4] Cosmin Ancuti, Codruta Orniana Ancuti, Tom Haber, and Philippe Bekaert. Enhancing underwater images and videos by fusion. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 81–88, 2012.

[5] Junhui Hou Runmin Cong Chunle Guo Chongyi Li, Saeed Anwar and Wenqi Ren. Underwater image enhancement via medium transmission-guided multi-color space embedding. *IEEE Transactions on Image Processing*, 30:4985–5000, 2021.

[6] Tom Haber Cosmin Ancuti, Codruta Orniana Ancuti and Philippe Bekaert. Enhancing underwater images and videos by fusion. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 81–88, 2012.

[7] Alejandro Rico Espinosa, Declan McIntosh, and Alexandra Branzan Albu. An efficient approach for underwater image improvement: Deblurring, dehazing, and color correction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*, pages 206–215, January 2023.

[8] X. Fu, P. Zhang, Y. Huang, and et al. A retinex-based enhancing approach for single underwater image. In *Proc. of IEEE Int. Conf. Image Process. (ICIP)*, pages 4572–4576, 2014.

[9] Xueyang Fu and Xiangyong Cao. Underwater image enhancement with global–local networks and compressed-histogram equalization. *Signal Processing: Image Communication*, 86:115892, 2020.

[10] G. Gallego et al. Underwater image enhancement and color correction for improved object detection. *Journal of Visual Communication and Image Representation*, 72:102036, 2020.

[11] Yecai Guo, Hanyu Li, and Peixian Zhuang. Underwater image enhancement using a multiscale dense generative adversarial network. *IEEE Journal of Oceanic Engineering*, 45(3):862–870, 2020.

[12] Alain Horé and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th International Conference on Pattern Recognition*, pages 2366–2369, 2010.

[13] Ijraset. Object detection in underwater using deep learning techniques.

[14] R. Khanam and M. Hussain. Yolov5: A deep look into the internal features of the popular object detector. *arXiv*, 2024.

[15] C. Li, J. Guo, R. Cong, and et al. Underwater image enhancement by dehazing with minimum information loss and histogram distribution prior. *IEEE Transactions on Image Processing*, 25(12):5664–5677, 2016.

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

[17] Chongyi Li, Chunle Guo, Wenqi Ren, Runmin Cong, Junhui Hou, Sam Kwong, and Dacheng Tao. An underwater image enhancement benchmark dataset and beyond, January 2019.

[18] J. Li et al. A novel underwater image segmentation algorithm using saliency detection and adaptive threshold. *Sensors*, 18(12):4297, 2018.

[19] W. Lin et al. Image quality assessment for underwater imagery. *Journal of Visual Communication and Image Representation*, 65:102671, 2019.

[20] P. Liu, G. Wang, H. Qi, C. Zhang, H. Zheng, and Z. Yu. Underwater image enhancement with a deep residual framework. *IEEE Access*, 2019.

[21] Youya Xia Md Jahidul Islam and Junaed Sattar. Fast underwater image enhancement for improved visual perception. *IEEE Robotics and Automation Letters (RA-L)*, 5(2):3227–3234, 2020.

[22] Yankun Yu Jun Chen Minghan Fu, Huan Liu and Keyan Wang. Dw-gan: A discrete wavelet transform gan for non-homogeneous dehazing. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 203–212, 2021.

[23] Abdullah Al Muksit, Fakhrul Hasan, Md. Fahad Hasan Bhuiyan Emon, Md Rakibul Haque, Arif Reza Anwary, and Swakkhar Shatabda. Yolo-fish: A robust fish detection model to detect fish in realistic underwater environment. *Ecological Informatics*, 72:101847, 2022.

[24] Karen Panetta, Chen Gao, and Sos Agaian. Human-visual-system-inspired underwater image quality measures. *IEEE Journal of Oceanic Engineering*, 41(3):541–551, 2016.

[25] Malte Pedersen, Joakim Bruslund Haurum, Rikke Gade, Thomas B. Moeslund, and Niels Madsen. Detection of marine animals in a new underwater dataset with varying visibility. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[26] Y. Peng, T. Cao, and P. Cosman. Generalization of the dark channel prior for single image restoration. *IEEE Transactions on Image Processing*, 27(6):2856–2868, 2018.

[27] Y. Peng and P. Cosman. Underwater image restoration based on image blurriness and light absorption. *IEEE Trans. Image Process.*, 26(4):1579–1594, 2017.

[28] Yan-Tsung Peng and Pamela Cosman. Underwater image restoration based on image blurriness and light absorption. *IEEE Transactions on Image Processing*, pages 1–1, 2017.

[29] Athanassios Skodras. Discrete wavelet transform: An introduction, 01 2003.

[30] M. V. Srigowri. Enhancing unpaired underwater images with cycle consistent network. *IEEE Access*, 2022.

[31] J. R. Terven and D. M. Cordova-Esparza. A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. *arXiv*, 2024.

[32] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pages 839–846, 1998.

[33] Ultralytics. Yolov11: Overview. Retrieved from https://docs.ultralytics.com/models/yolo11/overview, n.d.

[34] Zhou Wang, Alan Bovik, and Hamid Sheikh. Structural similarity based image quality assessment. *Digital Video Image Quality and Perceptual Coding, Ser. Series in Signal Processing and Communications*, 11 2005.

[35] X. Xu, H. Zhang, Y. Ma, K. Liu, H. Bao, and Q. Xu. Transdet: Toward effective transfer learning for small-object detection. *Remote Sensing*, 15(14):3525, 2023.

[36] Garima Yadav, Saurabh Maheshwari, and Anjali Agarwal. Contrast limited adaptive histogram equalization based enhancement for real time video system. In *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 2392–2397, 2014.

[37] T. Zhang, X. Li, Z. Xu, and B. Li. Yolov11: Hybrid backbone for improved object detection, tracking, and segmentation. *arXiv*, 2024.

[38] Wu Wang Yue Huang Zhenqi Fu, Xiaopeng Lin and Xinghao Ding. Underwater image enhancement via learning water type desensitized representations. arXiv preprint, 2021.

[39] K. Zhu, X. Liu, and S. Weibo. Underwater object detection using transfer learning with deep learning. In *Proceedings of the 2020 International Conference on Computers, Information Processing and Advanced Education*, 2020.