Underwater Image Enhancement using Bayesian Retinex

An MSc. Project Report Submitted
in Partial Fulfillment of the Requirements
for the Degree of

Master of Science in Mathematics & Computing

by

Rahim Khan
(222123040)

under the guidance of

Prof. Rajen Kumar Sinha

to the

DEPARTMENT OF MATHEMATICS

INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI
GUWAHATI - 781039, ASSAM

CERTIFICATE

*This is to certify that the work contained in this thesis entitled "**Underwater Image Enhancement using Bayesian Retinex**" is a bonafide work of **Rahim Khan (Roll No. 222123040)**, carried out in the Department of Mathematics, Indian Institute of Technology Guwahati under my supervision and that it has not been submitted elsewhere for a degree.*

Supervisor: **Prof. Rajen Kumar Sinha**

May, 2024 Department of Mathematics,

Guwahati. Indian Institute of Technology Guwahati Assam.

Abstract

The paper proposes a new Bayesian retinex algorithm for enhancing single underwater images. A simple color correction approach is first applied to remove color casts and recover natural colors. A maximum a posteriori formulation imposes multi-order gradient priors on the reflectance and illumination components of the color-corrected image. The l1 norm models piecewise/piecewise linear priors on the reflectance, while the l2 norm enforces spatial smoothness priors on the illumination.This formulation breaks down the enhancement problem into two denoising subproblems that can be efficiently optimized. The method operates pixelwise without requiring prior knowledge of underwater conditions. Experimental results show the proposed approach outperforms traditional and leading methods for color correction, naturalness preservation, detail enhancement, and artifact/noise suppression on qualitative and quantitative assessments. The method shows promising applications for challenging underwater image enhancement scenarios. In summary, it introduces a novel Bayesian retinex model with effective priors and a decomposition approach for high-quality single underwater image enhancement.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Underwater imaging is a crucial research field, given the abundant resources present in oceans, rivers, and lakes. However, underwater image processing poses unique challenges due to the complex physical properties of the underwater environment. Factors such as color distortion and contrast degradation arise from light absorption and scattering in water.

The underwater optical imaging model is explained, illustrating that captured light consists of three main components: direct, forward scattering, and backward scattering. The forward scattering component causes blurred structures in underwater images, while the backward scattering component obscures image edges and details. Additionally, color distortion results from the varying absorption rates of different wavelengths of light in water.

To address this problem the given research paper proposes a retinex-based algorithm using bayes theorem and l_1 prior to Reflectance and l_2 prior to illumination. This algorithm enhances a single underwater image by correction of color distortion and contrast degradation, ultimately striving to produce high-quality underwater images for further processing.

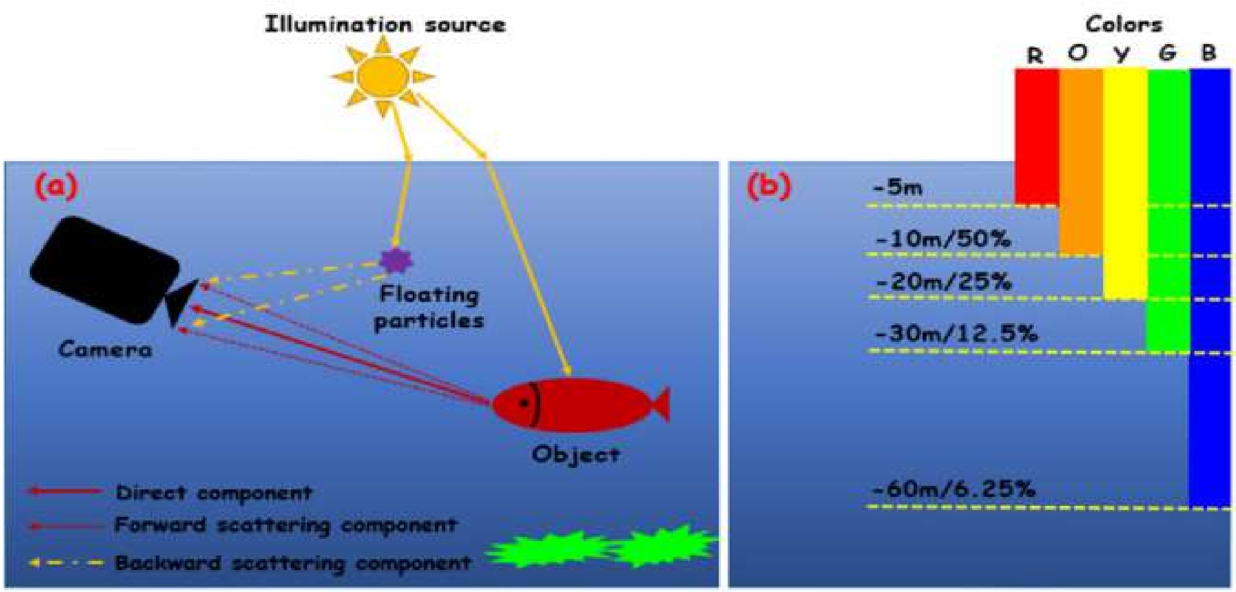

Fig. 1.1: Underwater Image formation

Chapter 2

RGB and HSV Color spaces

RGB Color Space[Bur09]

The RGB color model is a method for encoding colors by blending three primary colors: red (R), green (G), and blue (B). In the RGB model, color creation is achieved through an additive process starting with black and adding primary colors. It can be imagined as we are projecting red, green, and blue lights onto a white area in the dark room, and we can adjusting each beam's intensity independently, by adjustment we can form various colors. The color obtained by this process is the result of specific intensities of each primary color . Gray and white colors result from combining these three primary color beams at equal intensity. The RGB color space can be imagined as a three-dimensional cube of unit length, with the primary colors as R,G,B forming the coordinate axes. The RGB values are positive and fall within the range $[0, C_{max}]$;$C_{max} = 255$ for most digital images. Each possible color C_i can be visualized as a point in the RGB color cube of the form $C_i = (R_i, G_i, B_i)$, where $0 \leq R_i, G_i, B_i \leq C_{max}$. RGB values are normalized to the interval [0, 1] so that the resulting color space forms a unit cube. The point W = (1, 1, 1) corresponds to the color white, S = (0, 0, 0) corresponds to the color black, and all the points lying on the diagonal between S and W are shades of gray created from equal color components R = G = B.

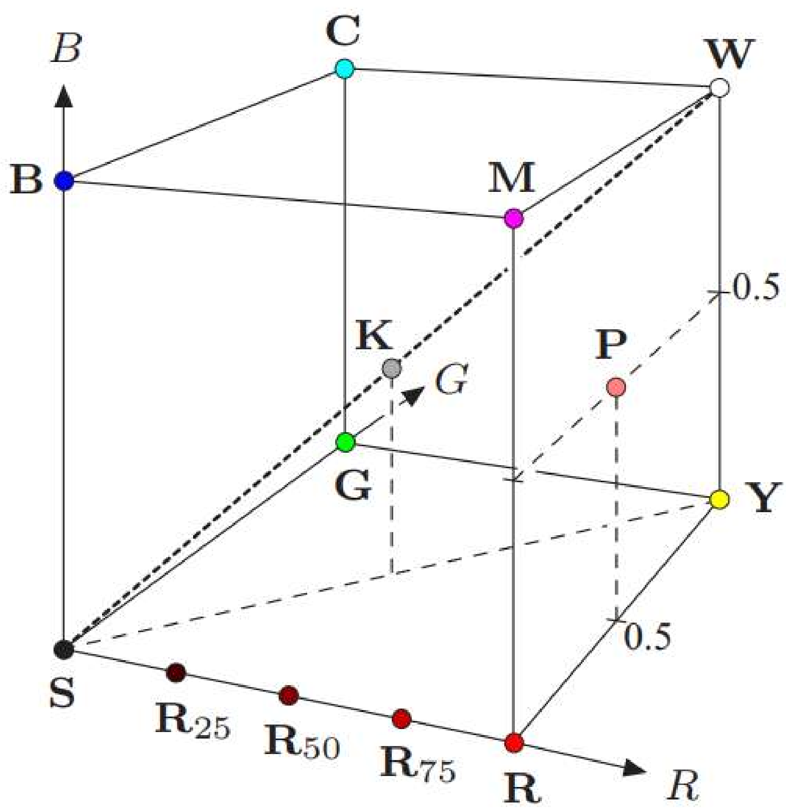

Fig. 2.1: RGB color space as a three-dimensional unit cube. The primary colors red (R), green (G), and blue (B) form the coordinate system. color (R), green (G), blue (B), cyan (C), magenta (M), and yellow (Y) gray (K) black S and white W.

HSV Color Space

We can imagine HSV color space as the six-sided inverted pyramid. where the top point is the point that represents the black color. White color is the center point of base. Three primary colors and their pairwise combination makes the base of pyramid. It can also be visualized as a cylinder. Brightness is the verticle axis. Saturation S is the distance from the center and H is the angle from red color line in the anticlockwise direction. In HSV color space H is the angle from Red color

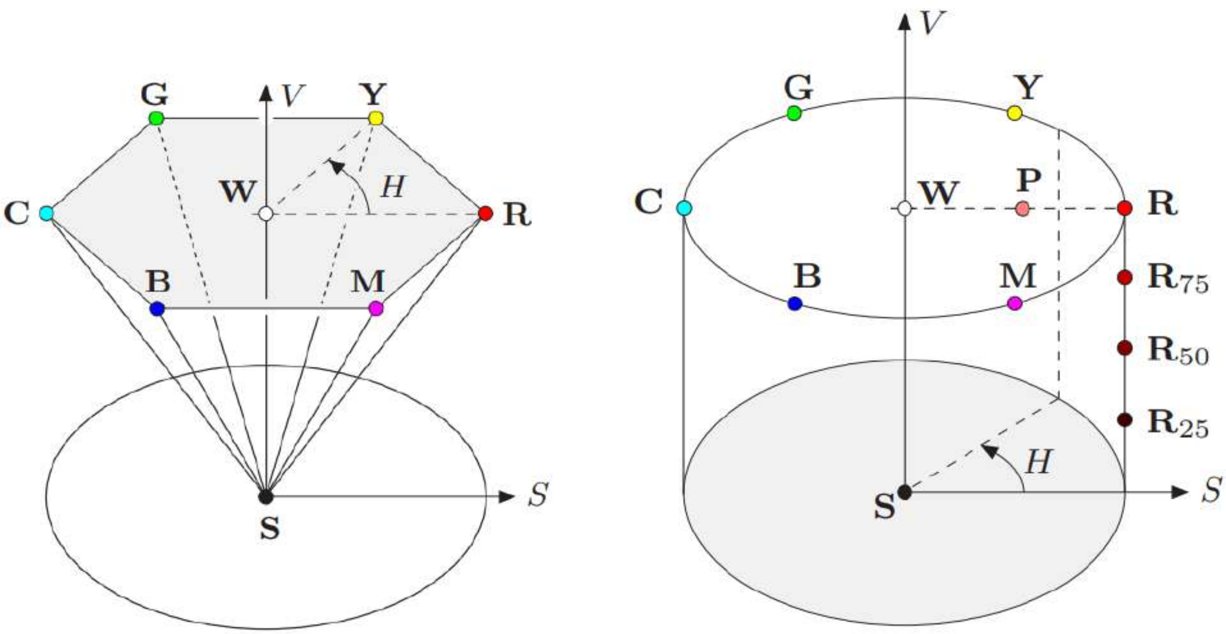

Fig. 2.2: HSV Color Space

point in anticlockwise direction, S is the distance of the point from the central axis and V is the distance of the point from the bottom.

The conversion process is defined as follows →

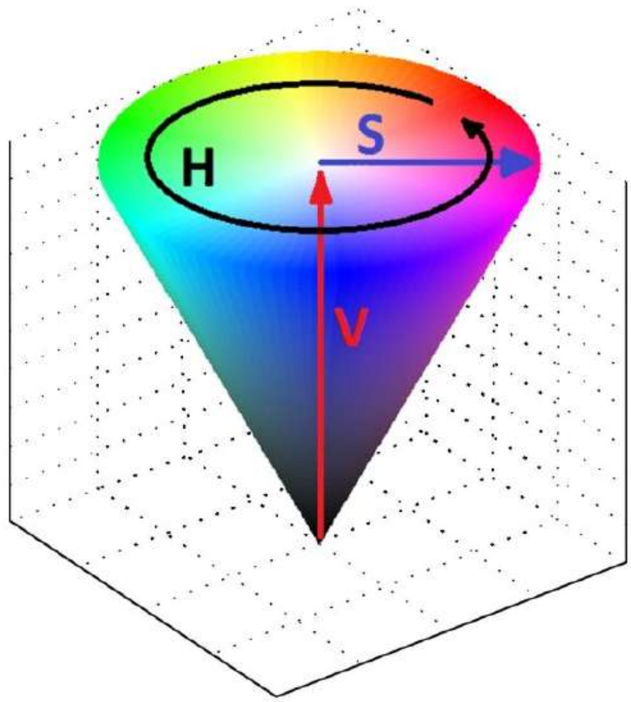

Fig. 2.3: HSV Color Space

for any given color pixel having RGB values P(R,G,B)

define $C_{high} = max(R, G, B)$, $C_{low} = min(R, G, B)$ and $C_{rng} = C_{high} - C_{low}$

$$S_{HSV} = \begin{cases} \frac{C_{rng}}{C_{high}}, & for\ C_{high} > 0 \\ 0, & otherwise \end{cases}$$

$$V_{HSV} = \frac{C_{high}}{C_{max}}$$

$$R' = \frac{C_{high} - R}{C_{rng}}, B' = \frac{C_{high} - B}{C_{rng}}, G' = \frac{C_{high} - G}{C_{rng}}$$

$$H' = \begin{cases} B' - G', & if\ R = C_{high} \\ R' - B', & if\ G = C_{high} \\ G' - R', & if B = C_{high} \end{cases} \qquad H_{HSV} = 60^\circ \times \begin{cases} H' + 6, & if\ H' < 0 \\ H', & otherwise \end{cases}$$

Color Correction

Underwater environments often appear green or blue due to the way green light, with its shorter wavelength, can travel farthest through water. To correct this color cast in underwater images, a statistical method is employed. Let's denote the observed underwater image as S. The color correction process involves the following steps: Initially, compute the mean value and mean squared error separately for the Red (R), Green (G), and Blue (B) channels of S. Then we determine the maximum and minimum pixel values for each RGB channel by applying the given formula $\rightarrow$

$$S^c_{max} = S^c_{mean} + \mu S^c_{var}$$
$$S^c_{min} = S^c_{mean} - \mu S^c_{var}$$

where c $\in$ {R,G,B} and S^c_{min} and S^c_{max} are the minimum and maximum of the c channel. S^c_{var} and S^c_{mean} are mean value and mean square error of the channel c. μ is the parameter to control the image dynamicity; Finally, the color corrected image is obtained by

$$S^c_{CR} = \frac{S^c - S^c_{min}}{S^c_{max} - S^c_{min}} \times 255$$

where S^c_{CR} is the color corrected image. On further simplification $\rightarrow$

$$\begin{aligned}
S^c_{CR} &= \frac{S^c - S^c_{min}}{S^c_{max} - S^c_{min}} \times 255 \\
&= \frac{S^c - S^c_{mean} + \mu S^c_{var}}{S^c_{mean} + \mu S^c_{var} - S^c_{mean} + \mu S^c_{var}} \times 255 \\
&= \frac{S^c - S^c_{mean} + \mu S^c_{var}}{2\mu S^c_{var}} \times 255 \\
&= \frac{255}{2} \times \left(1 + \frac{S^c - S^c_{mean}}{\mu S^c_{var}}\right).
\end{aligned}$$

$$\boxed{S^c_{CR} = \frac{255}{2} \times \left(1 + \frac{S^c - S^c_{mean}}{\mu S^c_{var}}\right)}$$

now this formula can be used for color correction of the underwater images.

* I have used standard deviation instead of variance in the color correction for better results.

Ref-[Din14]

Chapter 3

ADMM Algorithm

The ADMM (Alternating Direction Method of Multipliers) algorithm is employed for solving convex optimization problems. Its goal is to minimize an objective function under certain constraints by breaking down the problem into smaller sub-problems that can be addressed iteratively. The algorithm operates by updating the primal variables, dual variables, and a Lagrange multiplier in an alternating fashion. Consider the following optimization problem

$$min_x \mathcal{F}(x) + \mathcal{G}(z)$$

$$subject\ to\ \mathcal{A}x + \mathcal{B}z = c$$

where x $\in \mathbb{R}^n$, z $\in \mathbb{R}^m$,c $\in \mathbb{R}^p$, $\mathcal{B} \in \mathbb{R}^{p\times m}$, $\mathcal{A} \in \mathbb{R}^{p\times n}$. Here we are assuming that $\mathcal{F}$ and $\mathcal{G}$ are convex and differentiable functions. Now define the augmented langrangian

$$\mathcal{L}_\rho(x,z,y) = \mathcal{F}(x) + \mathcal{G}(z) + y^T(\mathcal{A}x + \mathcal{B}z - c) + \frac{\rho}{2}\|\mathcal{A}x + \mathcal{B}z - c\|_2^2$$

ADMM consists of the iterations

$$\begin{aligned}
x^{k+1} &= \arg\min_x \mathcal{L}_\rho(x, z^k, y^k) \\
z^{k+1} &= \arg\min_z \mathcal{L}_\rho(x^{k+1}, z, y^k) \\
y^{k+1} &= y^k + \rho(\mathcal{A}x^{k+1} + \mathcal{B}z^{k+1} - c)
\end{aligned}$$

where $\rho > 0$

Ref- [PE10]

Chapter 4

Metrics

4.1 Underwater Image Quality Metric (UIQM)

The Underwater Image Quality Measure (UIQM) is computed as a linear combination of three distinct underwater image attribute measures: Underwater Image Sharpness Measure (UISM), Underwater Image Contrast Measure (UIConM), and Underwater Image Colorfulness Measure (UICM). This metric evaluates three crucial aspects of underwater image quality: colorfulness, sharpness, and contrast. The recommended weighted parameters for UISM, UIConM, and UICM are empirically determined as 0.2953, 3.5753, and 0.0282, respectively. Higher UIQM values indicate a superior balance among colorfulness, sharpness, and contrast within the image.

UICM

Underwater images frequently encounter color-casting problems caused by light absorption in water, resulting in color distortion. As water depth increases, colors diminish in a predictable manner, starting with red due to its shorter wavelength, often giving underwater images a bluish or greenish appearance. Moreover, inadequate lighting conditions can worsen color desaturation. Effective enhancement algorithms for underwater images should prioritize accurate color representation. Studies suggest that natural scene colorfulness can be accurately captured using statistical image values. Users have the flexibility to choose color spaces, statistical order, fusion methods, and weighting factors for enhancement. Recognizing that the human visual system perceives colors in opposing planes, techniques like utilizing RG and YB opponent color components are employed in underwater image enhancement algorithms, known as the Underwater Image Color Model (UICM).

$$RG = R - G$$
$$YB = \frac{R + G}{2} - B$$

where R,G,B has there usual meaning.Underwater images usually have heavy noise. Therefore, we use asymmetric alpha-trimmed statistical values for measuring the colourfulness of underwater image instead of using the regular statistical values.

If we take an image of M$\times$N pixel size then let K=M$\times$N and all the pixels in the image are sorted in increasing order. Let $T_{\alpha_L} = \lceil \alpha_L K \rceil$ and $T_{\alpha_R} = \lfloor \alpha_R K \rfloor$ be the number of the smallest and greatest pixel values to be trimmed or discarded from the sorted sequence The asymmetric alpha-trimmed mean is defined by

$$\mu_{\alpha,RG} = \frac{1}{K - T_{\alpha_L} - T_{\alpha_R}} \sum_{i=T_{\alpha_L}+1}^{K-T_{\alpha_R}} Intensity_{RG,i}$$

$$\sigma^2_{\alpha,RG} = \frac{1}{N} \sum_{p=1}^{N} (Intensity_{RG,p} - \mu_{\alpha,RG})^2$$

Now similarly define $\mu_{\alpha,YB}, \sigma^2_{\alpha,YB}$. Then underwater image colorfulness measure (UICM) is calculated by the given formula $\rightarrow$

$$UICM = -0.0268\sqrt{\mu^2_{\alpha,RG} + \mu^2_{\alpha,YB}} + 0.1586\sqrt{\sigma^2_{\alpha,RG} + \sigma^2_{\alpha,YB}}$$

UISM

In underwater photography, achieving sharpness is particularly difficult because of considerable blurring caused by forward scattering. This blurring impairs the retention of fine details and edges in underwater images. To evaluate sharpness, the Sobel edge detector is applied to each RGB color component, producing an edge map. This map is then combined with the original image to generate a grayscale edge map, highlighting only the pixels associated with edges. The Enhancement Measure Estimation (EME) technique, suitable for images featuring uniform backgrounds and non-repetitive patterns, is used to assess edge sharpness in underwater images. The Underwater Image Sharpness Measure (UISM) is defined as

$$EME = \frac{2}{k_1 k_2} \sum_{l=1}^{k_1} \sum_{k=1}^{k_2} log\left(\frac{I_{max,k,l}}{I_{min,k,l}}\right)$$

$$UISM = \sum_{c=1}^{3} \lambda_c EME(graysclae\ edge_c)$$

where $\lambda_R = 0.299, \lambda_G = 0.587, \lambda_B = 0.114$

UIConM

Contrast has been identified as a key factor affecting underwater visual performance, such as stereoscopic acuity. In underwater imagery, contrast degradation often results from backward scattering. This paper proposes measuring contrast using the logAMEE (logarithmic Angular Measure of Edge Enhancement) applied to the intensity image. The logAMEE is defined as:

$$\text{logAMEE} = \frac{1}{k_1 k_2} \sum_{l=1}^{k_1} \sum_{k=1}^{k_2} \frac{I_{\max,k,l} \ominus I_{\min,k,l}}{I_{\max,k,l} \oplus I_{\min,k,l}} \times \log\left(\frac{I_{\max,k,l} \ominus I_{\min,k,l}}{I_{\max,k,l} \oplus I_{\min,k,l}}\right)$$

where the image is partitioned into blocks of size $k_1 \times k_2$, and $\ominus, \oplus, \otimes$ are the PLIP operations, The PLIP operations introduce an entropy-like operation to the traditional Agaian measure of enhancement by entropy (AMEE), which calculates the average Michelson contrast in local regions of an image. These operations, designed to provide nonlinear representations aligned with human visual perception, are also incorporated into the formulation of logAMEE. Given the typically poor lighting conditions underwater, logAMEE is preferred because its log and PLIP operations prioritize areas with low luminance. Underwater images can be represented as a linear combination of absorbed and scattered components. The effects of absorption and scattering contribute to degradation in color, sharpness, and contrast. Therefore, it is reasonable to adopt a linear superposition model to generate an overall measure of underwater image quality. The overall underwater image quality measure can be expressed as

$$UIQM = 0.0282 \times UICM + 0.2953 \times UISM + 3.5753 \times UIConM$$

Ref-[Aga16]

4.2 Underwater Color Image Quality Evaluation Metric(UCIQE)

Underwater color images, particularly those used in pipeline monitoring or engineering surveys. These images often suffer from issues like blurriness, low contrast, and strong color casts. Selecting an appropriate metric for evaluating image quality involves considering aspects such as correlation with subjective test data, computational cost, and limitations of the experimental setup. One consideration is whether to emphasize the blue-yellow axis in the CIELab color space, as suggested by Hasler and Suesstrunk. However, since CIELab is designed to be a uniform color space, this emphasis may not be reasonable. Additionally, it's suggested to avoid using the deviation of saturation (σ_s) as a metric because it can overly highlight dark areas, which are common in underwater images due to limited lighting. The underwater color image quality evaluation metric, UCIQE, operates in the CIELab color space. The pixel values of an image in this space are represented as Ip=[lp, ap, bp], where p ranges from 1 to N, with N being the total number of pixels in the image. Chroma (CI) is also a factor in this metric.

$$UCIQE = c_1 \times \sigma_c + c_2 \times con_l + c_3 \times \mu_s$$

for calculating UCIQE first we convert the image into CIELab color space.

let $I_{M \times N}$ is the image represented in CIELab color space consists of L,a,b channels.

$$\begin{aligned}
&chroma_{ij} = \sqrt{a_{ij}^2 + b_{ij}^2} \\
&\mu_c = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} chroma_{ij} \\
&\sigma_c = \sqrt{\frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left(chroma_{ij}^2 - \mu_c^2\right)} \\
&saturation_{ij} = \frac{chroma_{ij}}{l_{ij}} \\
&\mu_s = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} saturation_{ij} \\
&con_l = max(l) - min(l)
\end{aligned}$$

here I have used c_1=0.4680, c_2=0.2745 ans c_3=0.2576 as used in official implementation.

Ref-[Sow15]

4.3 Colorfulness,Contrast and Fog-density (CCF)

Colorfulness Index

Underwater images have serious color distortion because the indtensity of the light affects a lot . Fu and Panetta proposed that we can represent the colourfulness of an image using combination of image statistics. While calculating the colourfulness index of image ,the three channels R(i,j), G(i,j), and B(i,j) of a RGB underwater image are converted into logarithmic-scale. The conversion formulas are shown as follows,

$$B(i,j) = logB(i,j) - \mu_B$$

$$G(i,j) = logG(i,j) - \mu_G$$

$$R(i,j) = logR(i,j) - \mu_R$$

where μ_B, μ_G, μ_R are the mean values of logB(i,j),logG(i,j),logR(i,j)respectively. Now define

$$colorfulness = \frac{\sqrt{\sigma_\alpha^2 + \sigma_\beta^2} + 0.3\sqrt{\mu_\alpha^2 + \mu_\beta^2}}{85.59}$$

$$\beta = 0.5 \times (R + G) - B$$

$$\alpha = R - G$$

where $\sigma_\alpha^2, \sigma_\beta^2, \mu_\alpha, \mu_\beta$ are the variance and mean of α and β channels.

Contrast Index

Underwater color images suffer degradation from blurring caused by the scattering effect of water, particularly forward scattering, which makes assessing blurring crucial for evaluating image quality. The proposed method involves partitioning the underwater color image into 64x64 blocks and employing edge detection with the Sobel operator to differentiate between edge and flat blocks. Edge blocks are identified based on whether the number of edge pixels exceeds 0.2% of the total pixels in a block. The blurring index of an underwater color image is then computed as the sum of the RMS (Root Mean Square) contrast values of all identified edge blocks. The formula for calculating the RMS contrast index is given below

$$contrast = \sum_{i=1}^{T} \sqrt{\frac{1}{MN} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (I_{ij} - \bar{I})^2}$$

where I_{ij} is the intensity of i,j th pixel in the image and $\bar{I}$ is the average intensity of the pixel.

Fog Density Index

In predicting the degree of fog in a scene from a natural image, this model assesses deviations from statistical patterns observed in both fog-free and foggy images. Specifically, twelve features extracted from the Natural Scene Statistics (NSS) model, including offset brightness and entropy loss, are utilized to estimate the fog density of an image. The mean vector ν and the covariance matrix Σ are derived by fitting all the statistical features extracted from the test image into a multivariate Gaussian (MVG) model, which is calculated as follows,

$$MVG(f) = \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma|^{\frac{d}{2}}} exp\left(-\frac{1}{2}(f-v)^t\Sigma^{-1}(f-v)\right)$$

Let f be a d-dimensional feature vector representing statistical features, with t indicating transposition. The foggy level D_f is computed by measuring the Mahalanobis distance between the Multivariate Gaussian (MVG) model of a test image and the MVG model of 500 natural fog-free images. Similarly, the fog-free level D_{ff} is calculated by measuring the Mahalanobis distance between the MVG model of the test image and the MVG model of 500 foggy images. For instance, the computation of D_f is given by:

$$D_f(v_1, v_2, \Sigma_1, \Sigma_2) = \sqrt{(v_1 - v_2)^t \left(\frac{\Sigma_1 + \Sigma_2}{2}\right)^{-1} (v_1 - v_2)}$$

Here, v_1, v_2, Σ_1, and Σ_2 represent the mean vector and covariance matrices of the two MVG models, respectively.

Finally, the fog density index D of an image is computed as:

$$D = \frac{D_f}{1 + D_{ff}}$$

Now the CCF can be calculated as the weighted sum of these three quantities as follows

$$\boxed{CCF = \omega_1 \times Fogdensity + \omega_2 \times Contrast + \omega_3 \times Colorfulness}$$

here I have used $\omega_1 = 0.33988, \omega_2 = 0.61759, \omega_3 = 0.17593$ as suggested by the original paper. Ref-[Men18]

4.4 Entropy

To define entropy in the context of a discrete information source modeled as a Markov process, we aim to quantify the uncertainty or information content associated with the possible events or outcomes. Given a set of possible events with known probabilities $p_1, p_2, \ldots, p_n$, where $\sum_{i=1}^{n} p_i = 1$, we seek a measure $H(p_1, p_2, \ldots, p_n)$ that captures the degree of uncertainty or "choice" involved in selecting one of these events.

The properties we desire for H are as follows:

1. **Continuity in** p_i: The entropy H should be a continuous function of the probabilities p_i.

2. **Monotonic with** n: If all probabilities p_i are equal ($p_i = \frac{1}{n}$ for all i), then H should increase monotonically with n. This reflects that with more equally likely events, there is increased uncertainty or choice.

3.**Additivity under successive choices**: If a choice can be broken down into two successive choices, the overall entropy H should be the weighted sum of the individual entropies of the choices.

The function that satisfies these properties is the Shannon entropy, given by:

$$H = -\sum_{i=1}^{n} p_i \log p_i$$

where log denotes the logarithm (typically base 2) and $p_i \log p_i$ is defined as 0 when $p_i = 0$.

To calculate the entropy of an image, we follow these steps:

1. **Histogram Calculation**: For each channel (e.g., red, green, blue) of the image, compute a histogram that counts the frequency of each pixel intensity value.

2. **Probability Calculation**: Divide the frequencies in each histogram by the total number of pixels in the image, transforming them into probabilities $p_1, p_2, \ldots, p_n$ where n is the number of different intensity values.

3. **Entropy Computation**: Apply the Shannon entropy formula to each set of probabilities $p_1, p_2, \ldots, p_n$ corresponding to each channel of the image. Sum up these entropies to obtain the overall entropy of the image.

Therefore, the entropy of an image provides a measure of the uncertainty or amount of information content present in the pixel intensity distribution across its channels, reflecting how predictable or varied the pixel values are within the image.

Ref-[SHA48]

Chapter 5

Underwater Image Enhancement using Bayesian Retinex

The proposed algorithm in the paper [Wu21] utilizes both Retinex theory and Bayes theory of probabilities to enhance the underwater images. It first apply a color correction on the image as described in the chapter 3 . After the color correction it converts the image from other formats to HSV format as described in chapter 2. we construct the model as given below.

5.1 Model Construction

Step 1 : Color Correction :-

for each channel in RGB image do the color correction according to the given formula :-

$$U^c = \frac{255}{2}\left(1 + \frac{S^c - M^c}{\mu V^c}\right)$$

where S^c = Color channel having values in [0,1].

M^c = Mean of the channel .

V^c = Varience of the channel.

Step 2 : RGB to HSV conversion :-

Digital color images has 3 color channels R,G,B . Every value of RGB channels lies between 0 to 255 including both 0 and 255. RGB stands for RED,GREEN,BLUE and HSV stands for HUE, SATURATION, VALUE

$$B' = B/255, G' = G/255, R' = R/255$$

$$cmin = min(B', G', R')$$

$$cmax = max(B', G', R'),$$

$$\Delta = cmax - cmin$$

$$H = \begin{cases} 60 \times \left(\frac{G'-B'}{\Delta} \mod 6\right) & \text{if } cmax = R' \\ 60 \times \left(\frac{B'-R'}{\Delta} + 2\right) & \text{if } cmax = G' \\ 60 \times \left(\frac{R'-G'}{\Delta} + 4\right) & \text{if } cmax = B' \end{cases}$$

$$S = \begin{cases} 0 & \text{if } cmax = 0 \\ \frac{\Delta}{cmax} & \text{if } cmax \neq 0 \end{cases}$$

$$V = cmax$$

Step 3 : Model Description :-

Now select the V channel and let L=V , $L = L \times 255$. By Retinex theory, $L = I \circ R$ where R is Reflectance and I is Illumination. $\circ$ is the element wise multiplication.

$$I_{ij} \in [0, 255]$$

$$R_{ij} \in [0, 1]$$

Now consider L,I,R as Random Variables then :-

$$p(I, R|L) \propto p(L|R, I)p(I)p(R)$$

where $p(L|R, I)$ is the likelihood of L given R and I, $p(I)$ is the prior of I and $p(R)$ is the prior of R. Now consider that $p(L|R, I)$ follows the standard multivariate Normal distritribution for given errors are zero. Priors for R and I are considered as product of two priors. Prior1 of R follows the laplacian distribution of first order gradient of R and Prior 2 follows the 2nd order gradient of the R. Similarly we assume that the Prior of I is the product of two Priors which are prior1 of first order gradient of I which follows standard multivariate Normal distritribution and prior2 of the 2nd order gradient of I which follows standard multivariate Normal distritribution :-

$$e = L - I \circ R$$

$$p(L|I,R) \sim \mathcal{N}(e|0, \sigma^2 I)$$

$$p_1(R) \sim \mathcal{L}(\nabla R|0, s_1 I)$$

$$p_2(R) \sim \mathcal{L}(\triangle R|0, s_2 I)$$

$$p(R) = p_1(R)p_2(R)$$

$$p_3(I) \sim \mathcal{N}(\nabla I|0, \sigma_1^2 I)$$

$$p_4(I) \sim \mathcal{N}(\triangle I|0, \sigma_2^2 I)$$

$$p(I) = p_3(I)p_4(I)$$

where $\mathcal{N}$ is the Gaussian distribution and $\mathcal{L}$ is the Laplacian distribution.

$$\nabla_h = [-1, 1], \nabla_v = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

$$\triangle = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

$$\mathcal{L}(x|\mu, b) = \frac{1}{2b} exp\left(-\frac{|x-\mu|}{b}\right)$$

$$\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Step 4 : Objective Function :-

Since we want to maximize the probability of $p(I, R|L)$ so we will minimize the negative log likelihood of $p(I, R|L)$.

$$\mathcal{L}(L, I, R) = -\log p(I, R|L)$$

$$\mathcal{L}(L, I, R) = -\log p(L|R, I)p(I)p(R)$$

$$\mathcal{L}(L, I, R) = -\log p(L|R, I) - \log p(I) - \log p(R)$$

$$\mathcal{L}(L, I, R) = -\log p(L|R, I) - \log p(I) - \log p_1(R) - \log p_2(R)$$

$$\mathcal{L}(L, I, R) = -\log p(L|R, I) - \log p_3(I) - \log p_4(I) - \log p_1(R) - \log p_2(R)$$

$$\mathcal{L}(L, I, R) = \|I \circ R - L\|_2^2 + \frac{\sigma^2}{2\sigma^2}\|\nabla I\|_2^2 + \frac{\sigma^2}{2\sigma_1^2}\|\triangle I\|_2^2 + \frac{\sigma^2}{s_1}\|\nabla R\|_1 + \frac{\sigma^2}{s_2}\|\triangle R\|_1 + C$$

$$\mathcal{E}(I, R) = \|I \circ R - L\|_2^2 + \frac{\sigma^2}{2\sigma^2}\|\nabla I\|_2^2 + \frac{\sigma^2}{2\sigma_1^2}\|\triangle I\|_2^2 + \frac{\sigma^2}{s_1}\|\nabla R\|_1 + \frac{\sigma^2}{s_2}\|\triangle R\|_1$$

$$\mathcal{E}(I, R) = \|I \circ R - L\|_2^2 + \nu_1\|\nabla I\|_2^2 + \nu_2\|\triangle I\|_2^2 + \nu_3\|\nabla R\|_1 + \nu_4\|\triangle R\|_1$$

where $\nu_1 = \frac{\sigma^2}{2\sigma^2}$, $\nu_2 = \frac{\sigma^2}{2\sigma_1^2}$, $\nu_3 = \frac{\sigma^2}{s_1}$, $\nu_4 = \frac{\sigma^2}{s_2}$ and C is a constant.

Step 5 : Numerical Optimization :- To optimize the objective function . We have to convert l_1 norm to l_2 norm. We introduce two auxiliary variables d, h and two error terms m, n.

$$\mathcal{E}(I, R) = \|I \circ R - L\|_2^2 + \nu_3\|\nabla I\|_2^2 + \nu_4\|\triangle I\|_2^2 + \nu_1\|\nabla R\|_1 + \nu_2\|\triangle R\|_1$$

$$\begin{aligned}\mathcal{E}(I, R) = {} & \|I \circ R - L\|_2^2 + \nu_3\|\nabla I\|_2^2 + \nu_4\|\triangle I\|_2^2 \\ & + \nu_1\left(\|d\|_1 + \lambda_1\|\nabla R - d + m\|_2^2\right) + \nu_2\left(\|h\|_1 + \lambda_2\|\triangle R - h + n\|_2^2\right)\end{aligned}$$

Now we split this into three parts and optimize it using ADMM algorithm $\rightarrow$

P-1

$$d^k = \arg\min_d \left(\|d\|_1 + \lambda_1\|\nabla R^{k-1} - d + m^{k-1}\|_2^2\right)$$

$$h^k = \arg\min_h \left(\|h\|_1 + \lambda_2\|\triangle R^{k-1} - h + n^{k-1}\|_2^2\right)$$

P-2

$$R^k = \arg\min_R \left(\|R - \frac{L}{I^{k-1}}\|_2^2 + \nu_1\lambda_1\|\nabla R - d^k + m^{k-1}\|_2^2 + \nu_2\lambda_2\|\triangle R - h^k + n^{k-1}\|_2^2 \right)$$

$$m^k = m^{k-1} + \nabla R^k - d^k$$

$$n^k = n^{k-1} + \triangle R^k - h^k$$

P-3

$$I^k = \arg\min_I \left(\|I - \frac{L}{R^k}\|_2^2 + \nu_3\|\nabla I\|_2^2 + \nu_4\|\triangle I\|_2^2 \right)$$

Step 6 : Update for P-1 :-

$$d_h^k = \text{shrink}(\nabla_h R^{k-1} + m_h^{k-1}, \frac{1}{2\lambda_1})$$

$$d_v^k = \text{shrink}(\nabla_v R^{k-1} + m_v^{k-1}, \frac{1}{2\lambda_1})$$

$$h^k = \text{shrink}(\triangle R^{k-1} + n^{k-1}, \frac{1}{2\lambda_2})$$

where $\text{shrink}(x, \gamma) = \max(0, |x| - \gamma) \times \frac{x}{|x|}$ and $\frac{x}{|x|} = 0$ if $x = 0$

Step 7 : Update for P-2 :-

$$R^k = \mathcal{F}^{-1}\left(\frac{\mathcal{F}(L/I^{k-1}) + \nu_1\lambda_1\Phi_1 + \nu_2\lambda_2\Phi_2}{1 + \nu_1\Psi_1 + \nu_2\Psi_2}\right)$$

where $\Phi_1 = \mathcal{F}^*(\nabla_h).\mathcal{F}(d_h^k + m_h^{k-1}) + \mathcal{F}^*(\nabla_v).\mathcal{F}(d_v^k + m_v^{k-1})$ and

$\Phi_2 = \mathcal{F}^*(\triangle).\mathcal{F}(h^k + n^{k-1})$

$\Psi_1 = \mathcal{F}^*(\nabla_v).\mathcal{F}(\nabla_v) + \mathcal{F}^*(\nabla_h).\mathcal{F}(\nabla_h)$ and

$\Psi_2 = \mathcal{F}^*(\triangle).\mathcal{F}(\triangle)$ $\mathcal{F}$ is FFT Operator

$m_h^k = m_h^{k-1} + \nabla R_h^k - d_h^k$

$m_v^k = m_v^{k-1} + \nabla R_v^k - d_v^k$

$n^k = n^{k-1} + \triangle R^k - h^k$

Step 9 : Update for P-3 :-

$$I^k = \mathcal{F}^{-1}\left(\frac{\mathcal{F}(\frac{L}{R^k})}{\mathcal{F}(1) + \nu_3\Psi_3 + \nu_4\Psi_4}\right)$$

Algorithm :-

Input : $-$input value channel L, weighting parameters $\boldsymbol{\nu_1, \nu_2, \nu_3, \nu_4}$ **and** $\boldsymbol{\lambda_1, \lambda_2}$ and the number of iterations T

Initialization : $-$ initialize $\mathbf{I^0}$ = Gausian filter of L $\mathbf{R^0} = \mathbf{0}$ and

$d_h^0 = d_v^0 = h_h^0 = h_v^0 = m_h^0 = m_v^0 = n_h^0 = n_v^0 = 0$ and k=1

Iteration on k : − repeat until k=T :

update $d_h^k, d_v^k, h_h^k, h_v^k$ using update for P-1;

update R^k using update for P-2

update m_h^k, m_v^k, n^k using update for P-2;

update I^k using update for P-3

Stopping Criteria : − terminate iteration if k=T otherwise continue iteration k=k+1;

Output : − output the reflectance R and illumination I

Gamma Correction :-

$$I_e = W \times \left(\frac{I}{W}\right)^{\frac{1}{\gamma}}$$

use W = 250 and $\gamma = 2.5$

Image Reconstruction :-

$$L_e = I_e \circ R_e$$

now convert the image from HSV to RGB by using the formula given below .

Conversion from HSV to RGB :-

$$C = V \times S$$
$$X = C \times \left(1 - \left| \left(\frac{H}{60} \mod 2\right) - 1 \right|\right)$$
$$m = V - C$$
$$\begin{cases} (R', G', B') = (C, X, 0) & \text{if } 0 \leq H < 60 \\ (R', G', B') = (X, C, 0) & \text{if } 60 \leq H < 120 \\ (R', G', B') = (0, C, X) & \text{if } 120 \leq H < 180 \\ (R', G', B') = (0, X, C) & \text{if } 180 \leq H < 240 \\ (R', G', B') = (X, 0, C) & \text{if } 240 \leq H < 300 \\ (R', G', B') = (C, 0, X) & \text{if } 300 \leq H < 360 \end{cases}$$
$$(R, G, B) = (R' + m, G' + m, B' + m)$$

In the implementation I have used $\nu_1 = 1, \nu_2 = 10^{-3}, \nu_3 = 10^{-5}, \nu_4 = 10^{-3}, \lambda_1 = 10^{-4}, \lambda_2 = 10^{-3}, K = 4, \mu = 2.5(adjustable)$ as suggested by the author.

Ref-[Wu21]

5.2 Results

Comparison with other algorithms

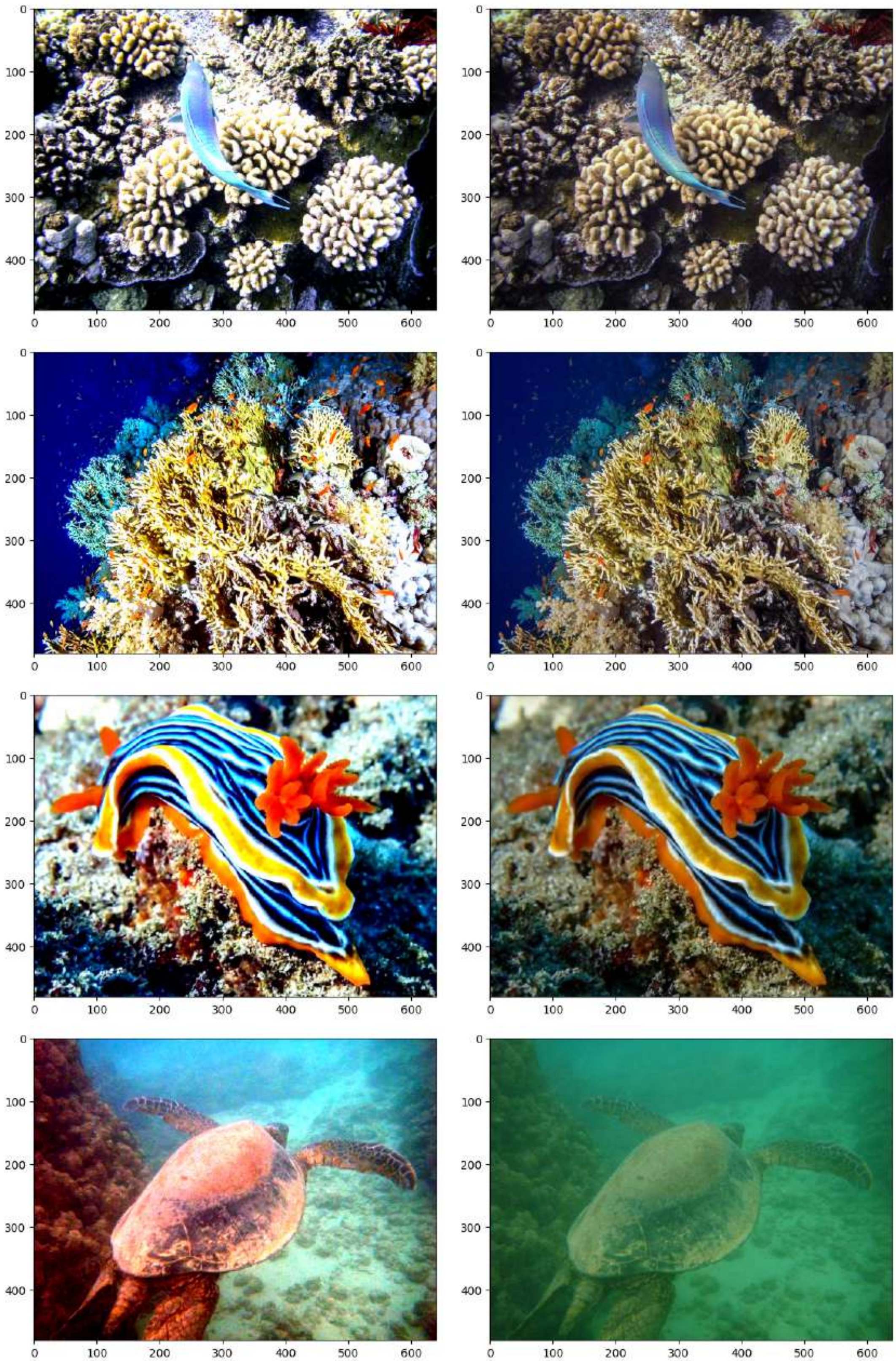

Fig. 5.1: right unprocessed and left processed

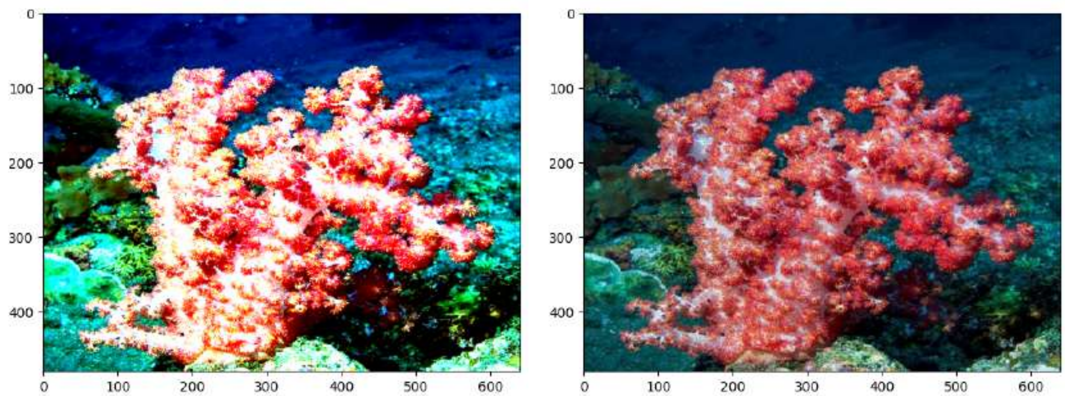

Fig. 5.2: right unprocessed and left processed

In the figure 6.1 and 6.2 we can see the enhanced and raw image. In the figure 6.3 I have shown the raw and processed image further from the figure 6.4 to figure 6.4 we can see the same image as figure 6.3 enhanced by different images. The implementation of the algorithms can be found in the github repo(link given in the conclusion). In the table 6.1 the comparison of the different metrics are given for the different algorithms . All the metrics are the average values of the 31 different images. In the figure 6.8 and figure 6.9 four enhanced images are given. In table 6.2 comparison of number of edges recovered by the different algorithms are given . Number of edges are calculated using Canny Edge detection [Can86] with threshold1=100 and threshold2=200 using OpenCV. In the original paper author have used different method for this purpose. From figure 6.10 to figure 6.13 bar chart comparison is given for single image for each metric namely CCF, UCIQE,UICM,UIConM,UIQM and UISM.

List algorithms for comparison

- Contrast Limited adaptive Histogram Equalization (CLAHE) .

- Grey World .
- Max RGB.
- Multiscale Retinex and Single Scale Retinex (MSR and SSR) .
- Multiscale Retinex with Color Restoration (MSRCR).
- Multiscale Retinex with chromaticity preservation (MSRCP) .
- Wavelength Compensation Image Dehazing (WCID).
- White Balance.

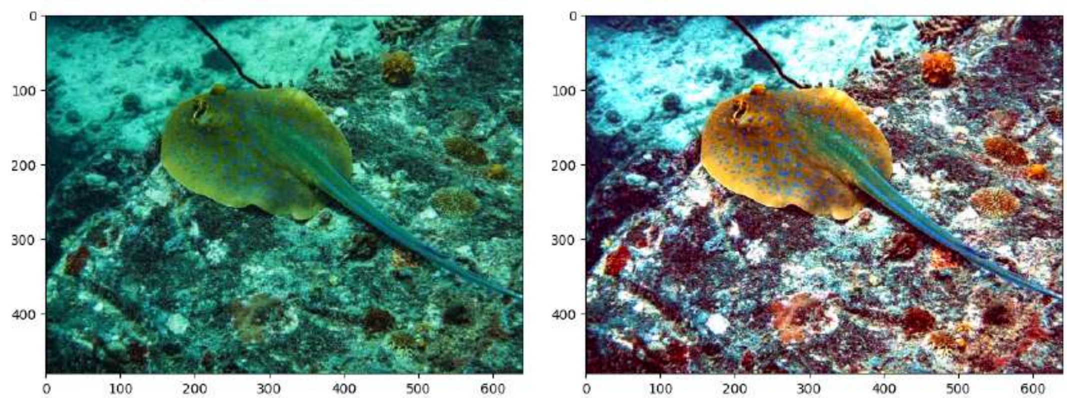

Fig. 5.3: raw and processed image

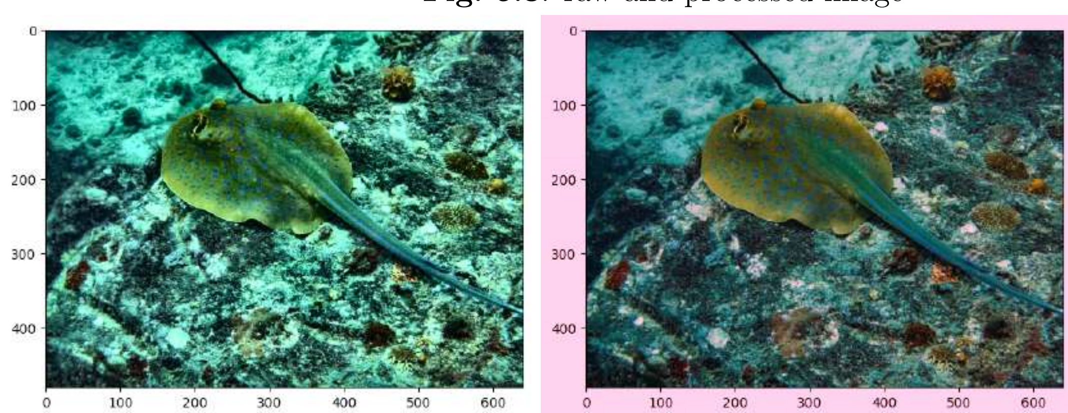

Fig. 5.4: Clahe [Vas13] and Grey World [Buc80]

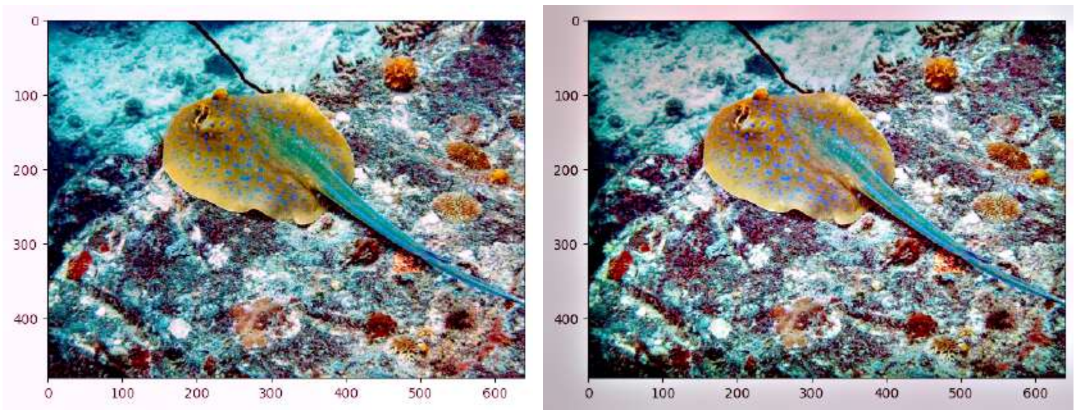

Fig. 5.5: SSR [Job15a] and MSR [Job15b]

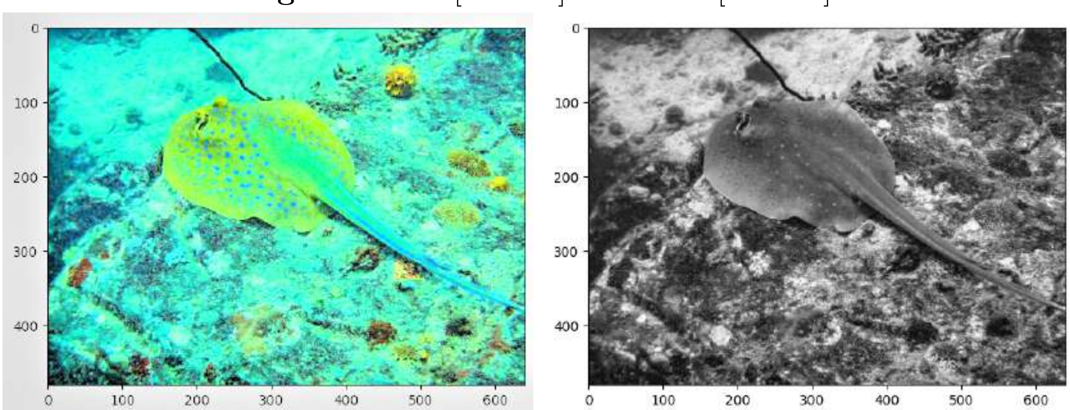

Fig. 5.6: MSRCP [San12] and MAXRGB [Lan77]

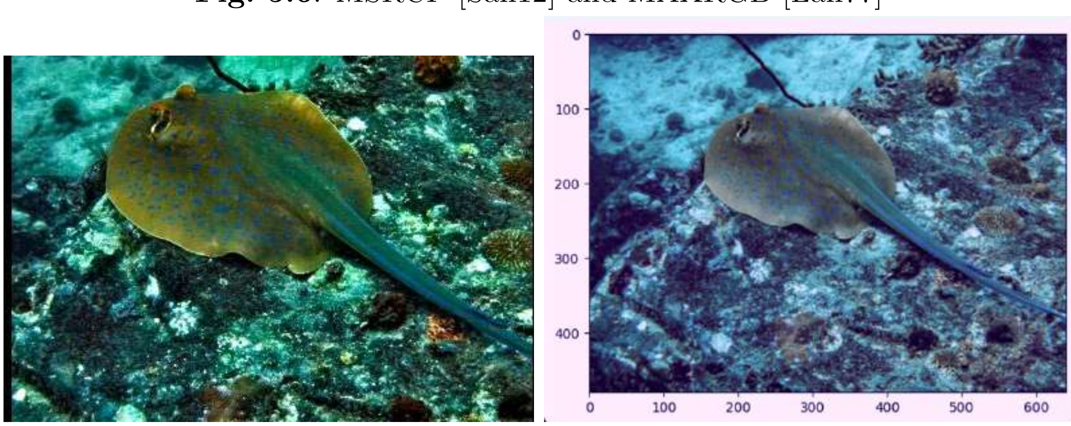

Fig. 5.7: WCID [CC12] and white balance [M07]

Table 5.1: Comparison of Image Processing Methods

Method	UCIQE	UIQM	UIConM	UISM	UICM	Entropy	CCF
Raw	34.5435	0.9638	-0.1489	4.0165	10.9910	6.4360	6.1078
Processed	36.1423	1.3170	-0.1368	4.8742	13.0017	6.7837	12.9806
Clahe	34.2644	1.2024	-0.1560	4.9006	11.1018	6.7788	11.365
Grey World	30.8121	0.9518	-0.1493	3.9647	11.1663	6.4246	6.5771
Max RGB	27.3190	0.9303	-0.1473	4.3682	5.9217	6.5853	2.852
MSR	34.5479	1.3611	-0.1393	5.1399	12.1020	7.7569	17.6559
MSRCP	38.2102	1.0643	-0.1327	4.4564	7.8934	7.0115	5.1098
MSRCR	15.0838	0.7142	-0.1198	3.8844	-0.1653	6.0080	2.3201
SSR	36.0846	1.2733	-0.1338	4.8325	11.5080	7.0979	19.3236
WCID	32.6000	0.7255	-0.0979	2.5337	11.6125	4.3645	12.4484
White Balance	34.9375	0.8975	-0.1495	3.8324	10.6460	6.3755	7.1807

Table 5.2: Recovered Edges by different algorithms

	Image A	Image B	Image C	Image D
Clahe	38020	51369	18999	17576
Grey World	19387	36991	10709	8438
Max RGB	21602	38101	10566	8929
MSR	36897	55918	22716	18669
MSRCP	13458	41676	10444	6818

MSRCR	5300	4512	4402	4159
Processed	38291	43419	17687	15331
Raw	21127	37759	9440	8196
SSR	37285	48639	16494	14904
WCID	10238	23330	7293	5832
White Balance	20123	37840	11066	9094

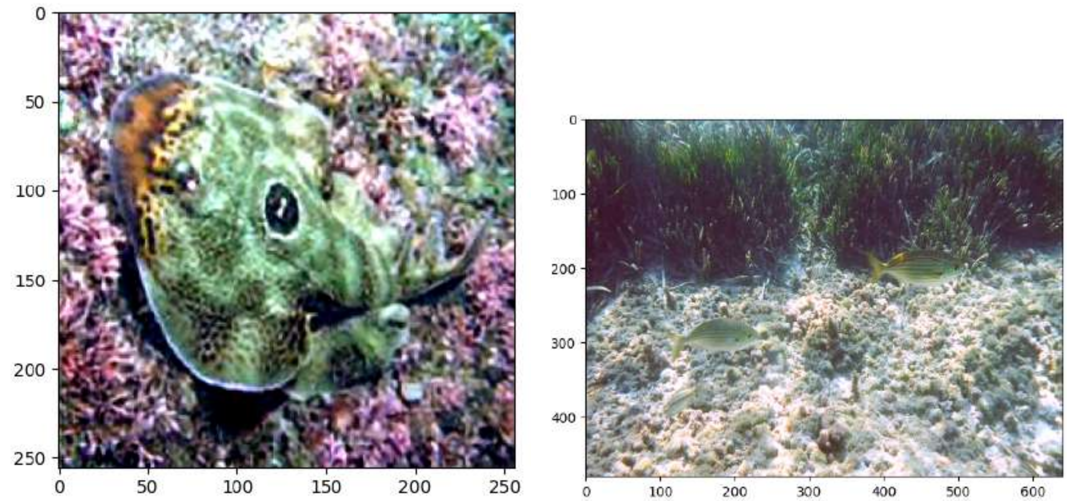

Fig. 5.8: (A) & (B)

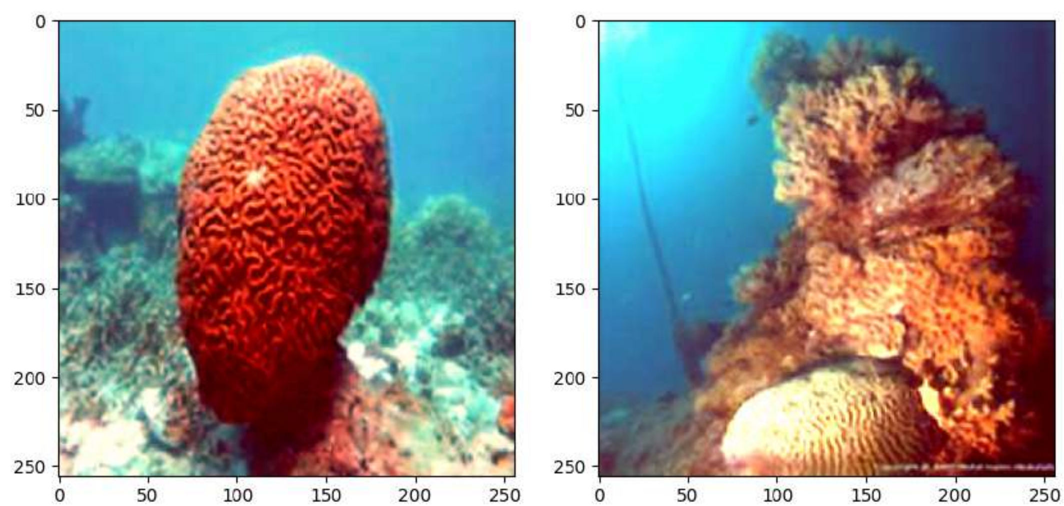

Fig. 5.9: (C) & (D)

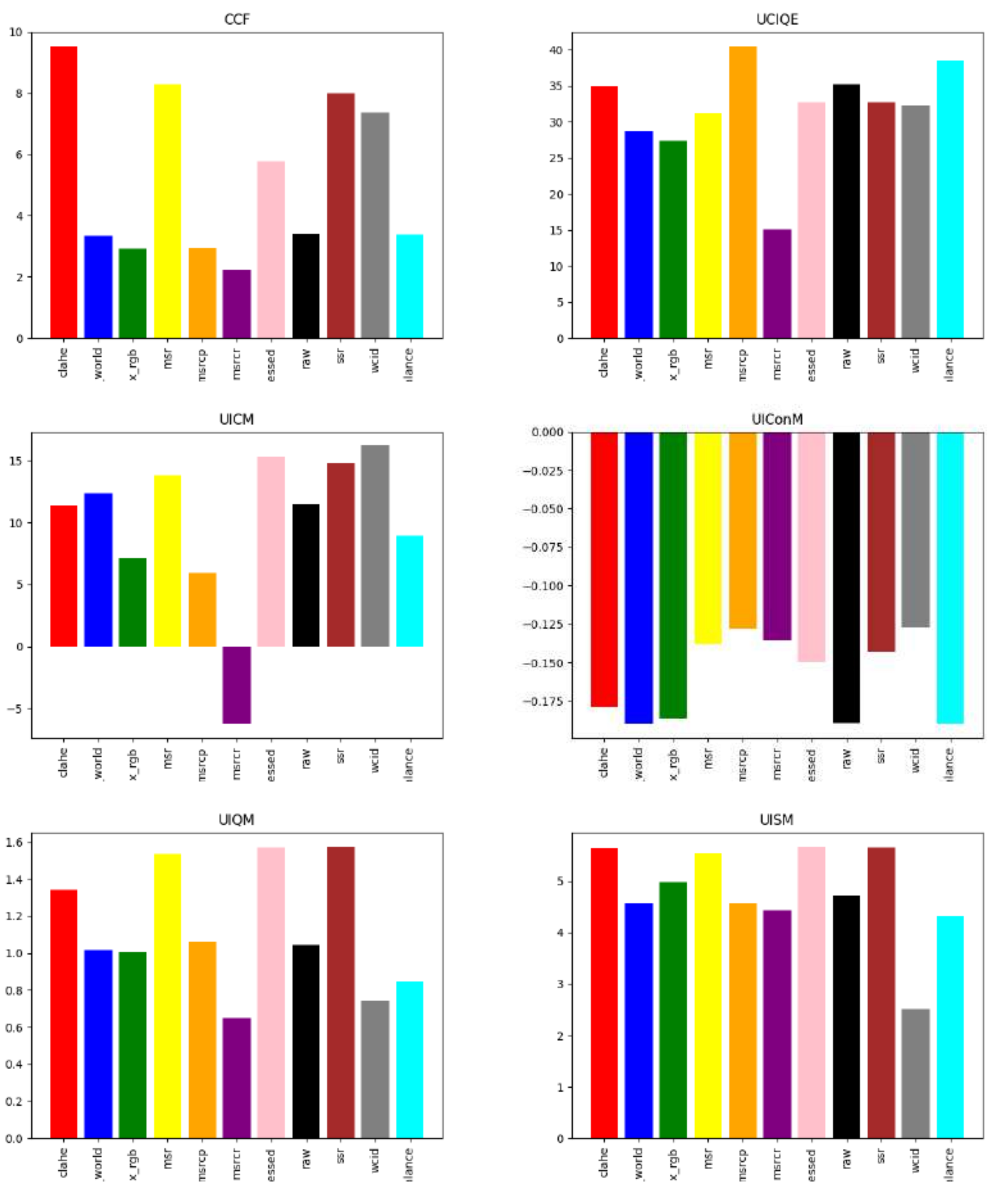

Fig. 5.10: Metric comparison for image A

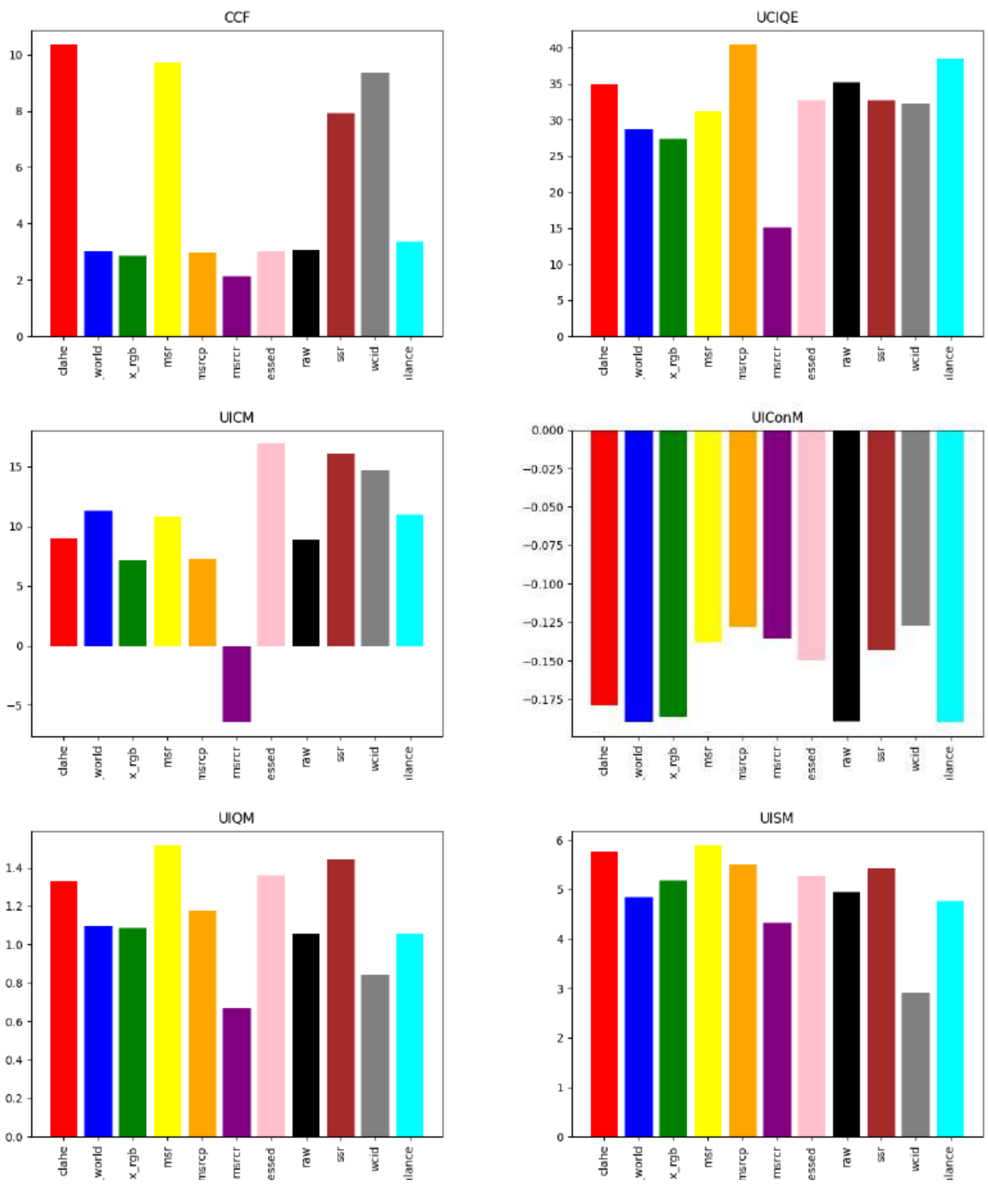

Fig. 5.11: Metric comparison for image B

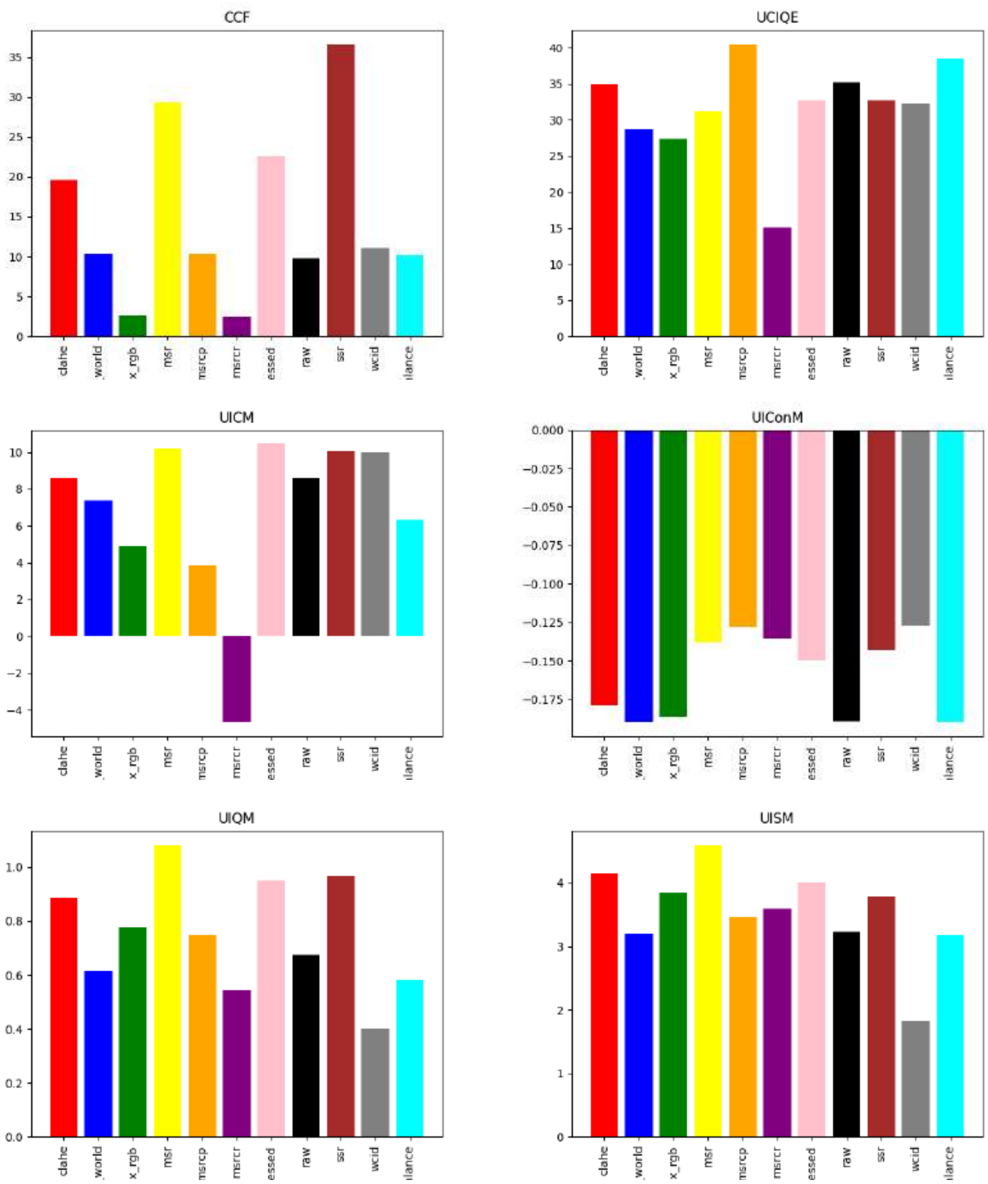

Fig. 5.12: Metric comparison for image C

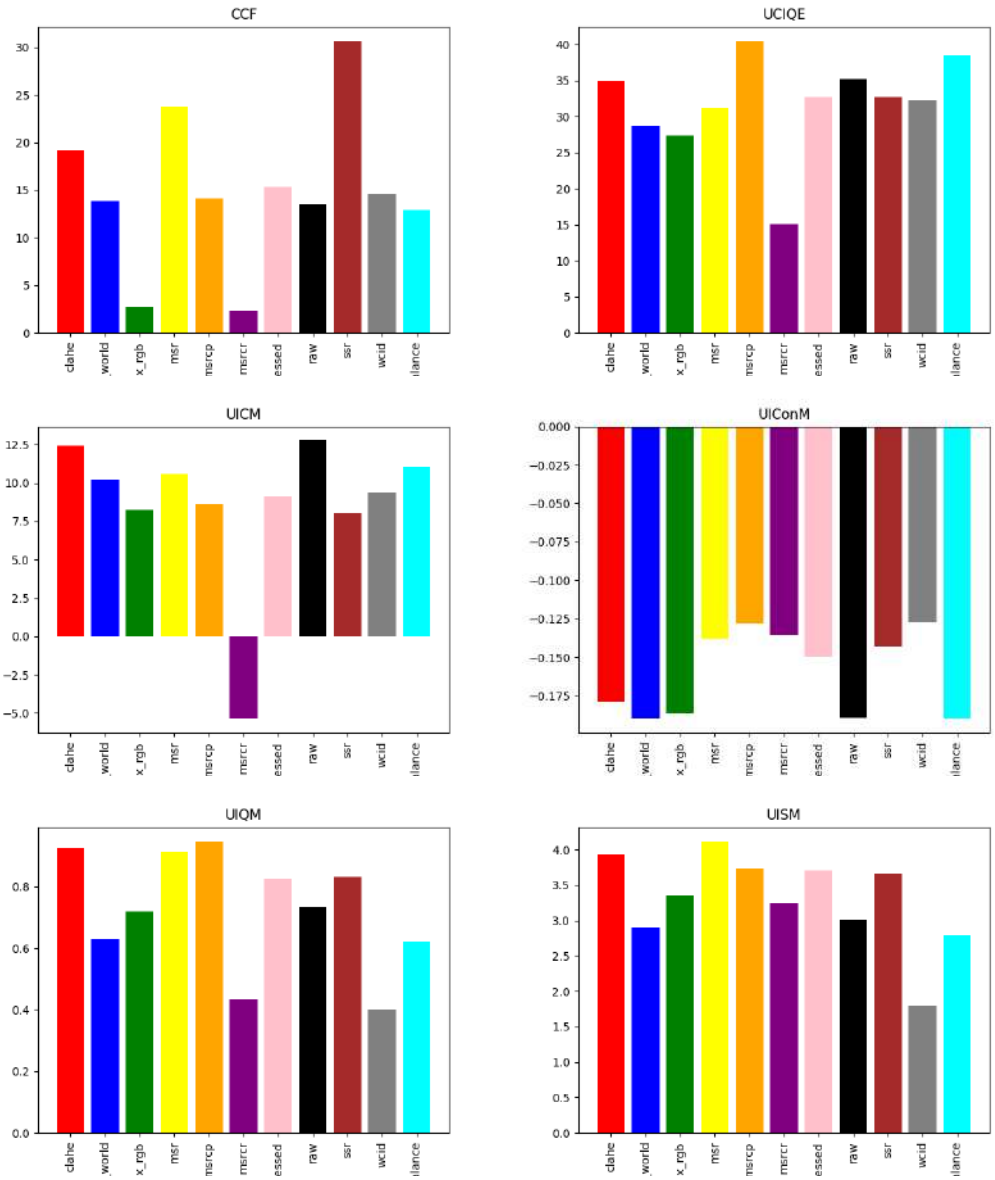

Fig. 5.13: Metric comparison for image D

Chapter 6

Conclusion and Future Work

Conclusion

The described method employs a straightforward yet powerful color correction technique to eliminate color casts and restore natural coloration, making it a valuable preprocessing step for other underwater enhancement techniques. It introduces an enhancement model based on bayesian retinex that incorporates first order gradient on Refectance and second order gradient on Illumination. This strategy enhances fine details and structures by incorporating spatial smoothness and linear approximations. To tackle the complexity of the problem, the method decomposes it into three simpler denoising subproblems, which are addressed using an efficient alternating optimization algorithm. This approach is underpinned by rigorous mathematical convergence analysis. Notably, the described model operates on a pixel-wise basis, which means it processes each pixel independently without relying on prior knowledge of specific underwater imaging conditions. Thorough analyses confirm the method's effectiveness in terms of color accuracy, its capability to manage difficult scenes, evaluations of parameters, algorithm convergence, and enhancements compared to baseline methods, as illustrated by an ablation study. In summary, the proposed model exhibits robust and promising performance in enhancing underwater images.

Future Work

For future work, I am planning to explore deep learning based approaches for underwater image enhancement, specifically methods like MDGAN (Multi-Decoder Generative Adversarial Network) and UGAN (Underwater Generative Adversarial Network). These techniques leverage the power of generative adversarial networks (GANs) and convolutional neural networks (CNNs) to learn a non-linear mapping between degraded underwater images and their enhanced counterparts from large training datasets.

MDGAN employs multiple decoupled decoders to separately recover the three color channels, allowing the network to better capture the unique distortions in each channel. UGAN introduces a underwater image degradation model as a domain translation process, learning a mapping between rendered underwater style images and their corresponding clear versions.

References

[Aga16] Karen Panetta; Chen Gao; Sos Agaian. Human-visual-system-inspired underwater image quality measures. *IEEE Journal of Oceanic Engineering*, July 2016.

[Buc80] G. Buchsbaum. A spatial processor model for object colour perception. *Journal of the Franklin Institute*, July 1980.

[Bur09] Wilhelm Burger ; Mark J. Burge. Principles of digital image processing fundamental techniques. *Springer-Verlag*, 2009.

[Can86] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Nov 1986.

[CC12] John Y. Chiang and Ying-Ching Chen. Underwater image enhancement by wavelength compensation and dehazing. *IEEE TRANSACTIONS ON IMAGE PROCESSING*, 2012.

[Din14] Xueyang Fu; Peixian Zhuang; Yue Huang; Yinghao Liao; Xiao-Ping Zhang; Xinghao Ding. A retinex-based enhancing approach for single underwater image. *IEEE International Conference on Image Processing (ICIP)*, oct 2014.

[Job15a] Bo Jiang; Glenn A. Woodell;Daniel J. Jobson. Novel multi-scale retinex with color restoration on graphics processing unit. *Springer Link*, 2015.

[Job15b] Bo Jiang; Glenn A. Woodell;Daniel J. Jobson. Novel multi-scale retinex with color restoration on graphics processing unit. *Springer Link*, 2015.

[Lan77] E.H Land. Maximum response in each channel of rgb caused by a white patch. *Sci. Amer.*, 1977.

[M07] Ebner M. Color constancy. *wiley*, 2007.

[Men18] Wang Yan; Li Na ; Li Zongying ; Gu Zhaorui; Zheng Haiyong; Zheng Bing ; Sun Mengnan. An imaging-inspired no-reference underwater color image quality assessment metric. *Computers Electrical Engineering*, Aug 2018.

[PE10] S. Boyd; N. Parikh; E. Chu; B. Peleato; and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, Nov 2010.

[San12] Sudharsan Parthasarathy; Praveen Sankaran. An automated multi scale retinex with color restoration for image enhancement. *2012 National Conference on Communications (NCC)*, 2012.

[SHA48] C. E. SHANNON. A mathematical theory of communication. *The Bell System Technical Journal*, 1948.

[Sow15] Miao Yang; Arcot Sowmya. An underwater color image quality evaluation metric. *IEEE Transactions on Image Processing*, Oct 2015.

[Vas13] Vasamsetti. Contrast limited adeptive histogram equalization. *Science Direct*, 2013.

[Wu21] Peixian Zhuang; Chongyi Li ; Jiamin Wu. Bayesian retinex underwater image enhancement. *Engineering Applications of Artificial Intelligence*, Jan 2021.