International Journal of Computer Trends and Technology
ISSN: 2231–2803 / https://doi.org/10.14445/22312803/IJCTT-V73I4P104

Volume 73 Issue 4, 28-34, April 2025
© 2025 Seventh Sense Research Group®

Original Article

Enhancement of Underwater Images Using Neural Style Transfer

Apeksha Jain[1], D.A. Mehta[2]

[1,2]*Department of Computer Engineering, Shri Govindram Seksaria Institute of Technology & Science., Indore, Madhya Pradesh, India.*

[2]*Corresponding Author : mehta_da@hotmail.com*

Received: 05 March 2025 Revised: 06 April 2025 Accepted: 19 April 2025 Published: 29 April 2025

Abstract - *The quality of underwater images is considerably significant in the computer vision arena to understand sea life and assess the geological environment and archaeology beneath the water. Owing to the physical properties of underwater environments, capturing sharp underwater images becomes a challenging task. These images mostly undergo color distortion and visibility deprivation because of light absorption and scattering. The current approaches are time-consuming and require large datasets to achieve reasonable results in enhancing underwater images. This paper presents a technique for enhancing underwater images using neural style transfer. The resultant output image is less hazy, and the content loss is very less. A comparison has also been made between the output image obtained with and without segmentation. In addition, the content loss has been reduced, and the loss percentage and histogram showing the haze difference between the input and generated output image have been displayed.*

Keywords - *Underwater image enhancement, Neural style transfer, Color correction, Haze removal.*

1. Introduction

Underwater imaging is essential in ocean scientific research. Visually guided autonomous underwater vehicles and remotely operated vehicles are widely used in a variety of applications, including monitoring marine species migration [16], inspecting underwater infrastructure like submarine cables [17], underwater scene analysis, etc. [18].

Due to the underwater conditions, clicking images underneath the water is often challenging. One of the major challenges for underwater robots is that even with the use of high-resolution cameras, visual sensing is still impacted owing to poor visibility, light refraction, absorption, and scattering [18, 19, 20].

Absorption reduces light penetration as the robot moves deeper into the water or away from the camera, and colors gradually fade based on their wavelengths. This leads to bluish or greenish color of images as the wavelengths of blue and green color are least attenuated inside water. The scattering effect alters the direction of the light towards the camera, resulting in a distinctive veil that superimposes itself on the image, hiding the scene and thereby blurring the objects. In addition, the amount of dispersed light is also increased by the presence of tiny floating particles known as marine snow.

Besides the magnitude of attenuation, scattering depends on many complex factors, including water temperature, salinity, and the quantity and density of particles inside the water. Serious degeneration makes recovering the appearance and color of underwater images difficult [1, 2]. But color is very important for underwater vision and research, and thereore effectively enhancing the quality of underwater images with their real color is a challenging problem.

The problems faced in enhancing underwater images are:

- A large dataset is required for training testing purposes.
- Requirement of a set of paired datasets.
- Processing speed is slow.
- Efficiency is less.

In order to help researchers and underwater robots research inside the ocean, a neural style transfer approach for enhancing underwater images is proposed for this study. Using this approach, the quality of these images will be improved with faster speed and minimum information loss. In neural style transfer, one image is composed in the style of another image. The reference or style image is a good-quality underwater image with less haze. By transferring the style of reference image into the input content image, haze is reduced, and the focus on the object inside the water is increased. Underwater images have blue or green color as the dominant color. Therefore, a histogram was plotted to find the dominance of the color. The hazy images are input images, and the good-quality images are reference images. The output image resembles an input image, which is hazy and is painted in the reference image's manner. In order to achieve this, the output image is optimized to match both the reference image's style statistics and the input image's content statistics [3]. A VGG network is used to extract the statistics from the images. The convolution layers of the VGG network extract features from the input image.

Content loss is the loss in the contents of the input image while generating output. Style loss is the loss of style in the output image. Total loss is calculated from style loss and content loss. The generated image is optimized for the total loss. Therefore, the final image obtained has less haze and minimal content loss.

2. Related Work

This section focuses on the basic concepts of convolutional neural networks and Generative networks. Additionally, various research articles presenting different methods to improve image quality are studied and listed to improve the image quality of underwater images.

A Convolutional Neural Network is a Deep Learning algorithm that takes an input image and makes the required changes to make it different from other input images. Earlier, image filters were hand-engineered to make the required changes on the input image, and they also required complex training. Convolutional Neural Networks overcome this drawback and can learn image filters and characteristics of the image. They have the ability to capture the Spatial and Temporal dependencies in an image using the appropriate filters.

The Convolutional Neural Network's role is to transform an image into something easier to process while retaining critical features to make a good prediction.

Generative Networks [14,15] are the commanding class of neural networks used for unsupervised learning.

Ian J. Goodfellow [7] created and introduced Generative networks. They are two neural network models competing against one another and can analyze, capture, and copy variations within a dataset.

Generative networks were developed because conventional neural networks could be easily misclassified by adding a small amount of noise to the original data. The addition of noise increases the chances of wrong predictions. The purpose of Generative networks is to overcome some of the known flaws of machine learning models, such as overfitting.

Generative Networks belong to the set of generative models. This means that generative networks can produce new images from existing ones. Generative networks automatically discover and learn the regularities or patterns in input data. The model can be used to generate new examples that are plausible to have come from the original dataset. There are two sub-models of Generative networks:

- The generator model is trained to generate new examples.
- The discriminator model attempts to categorize examples as either authentic or fraudulent. The generator attempts to deceive the discriminator by producing fraudulent images that appear to be drawn from authentic distribution. At the same time, the discriminator works to get better at discarding the fraudulent images, and ultimately, the generator learns to model the underlying distribution.

Generative networks are of the following types:

- Neural style transfer
- Deep Dream
- Deep Convolutional GAN
- Pix2Pix
- CycleGAN
- Variational Autoencoders

Classically, the quality of images was improved using handcrafted filters. Using them, local color constancy and contrast/lightness were improved. Z.-u. Rahman et al. in [4] proposed the multiscale Retinex with color restoration. In this technique, color constancy is combined with local contrast or lightness improvement to transform digital images into versions that tend towards the realism of direct scene observation.

The quality of underwater images can be enhanced by using Convolutional Neural Networks. The authors of [5] proposed a CNN-based network: UIE-Net. Color correction and haze removal are the two tasks used to train this network. In this combined training approach, it becomes feasible to learn a robust feature representation for both tasks simultaneously. In order to handle the training of UIE-net, two million training images have been synthesized based on the physical underwater imaging model.

The problems of using CNN for underwater image enhancement are:

- They require extensive data for training purposes.
- They have slow speeds and are less efficient.

Simonyan, K et al. proposed VGG networks in [6]. VGG network is a CNN model. It achieves 92.7% top-5 test accuracy in the ImageNet dataset. This dataset comprises more than 14 million images and belongs to 1000 classes. VGG is a good feature extractor; therefore, it has been used to extract features from the images.

Ian J. Goodfellow et al. in [7] propose generative networks, which comprise generative and discriminative models. The task of this model is to learn whether a sample comes from the model distribution or the data distribution. The generative model is supposed to generate new images from the existing ones. An analogy between counterfeiters and police can be set between generative and discriminative models. While the generator model attempts to produce counterfeit currency and utilizes it without detection, the discriminative model looks for counterfeit currency. The competition in this game drives both to improve their strategies until the counterfeits are indistinguishable from the genuine ones. In other words, the generator keeps generating new images, and the discriminator keeps differentiating between the true and fake images. The discriminator sometimes fails to differentiate between fake and true images. At this point, the generator successfully fools the discriminator.

In [8], the authors present a fast underwater image improvement approach using CycleGAN. The problem with cycleGAN is that it requires a large dataset. [8] Proposed a model that gives enhanced underwater images using cycleGAN. EUVP dataset has been presented. This dataset comprises a set of paired and unpaired images. The drawback of using this type of generative network is that it requires a paired set of datasets, and such datasets are not easy to find or create.

A Neural Algorithm of Artistic Style, as presented by Gatys et al. in [3], can separate and recombine images. It enables the creation of new images of high perceptual quality that blend the content of an arbitrary photograph with the appearance of various famous artworks. The authors proposed neural style transfer wherein the generated output image is the input image in the style of a reference image.

Ming Lu et al. [9] proposed a new interpretation by treating it as an optimal transport problem. The authors also show the relationship between their formulation and earlier work, such as Adaptive Instance Normalization (AdaIN) and Whitening and Coloring Transform (WCT). A closed-form solution named Optimal Style Transfer (OST) is derived, and the authors have given a self-made formulation by considering the content loss of Gatys. Additionally, Content loss is the loss of content from the original image.

3. Proposed Method

3.1. Architecture for Enhancing Underwater Images

Style transfer is used to transfer the background from a good quality image with less haze, termed a reference image, to the input image, termed a content image.

In Figure 1, Input image is any underwater image. It is a hazy image with blue or green haze suppressing the underwater habitat of that image, and the reference image is a good-quality image taken from Google Images. The purpose of using this image is to paint the input image in the same style as the reference image without losing its important content.

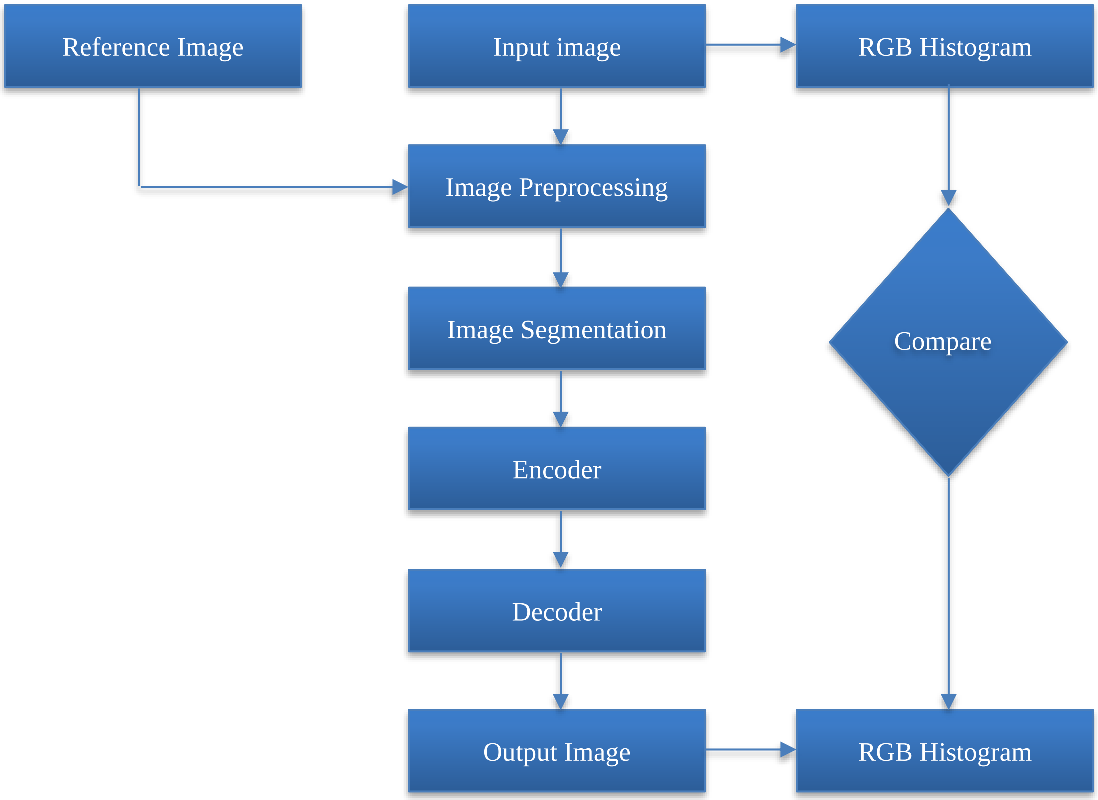

Fig. 1 Proposed approach: Flow chart

In the approach we have proposed, edge-based image segmentation was used. With this technique, edges have been detected in an image. These edges are used to identify the objects and are thought to be their boundaries. Sobel and canny edge detection algorithms are examples of edge-based segmentation techniques. SuimNet [10], a pre-trained model, is used for image segmentation. Image segmentation of both the content image and the reference image is done.

3.2. Encoder and Decoder Structure

Figure 2 shows the structure of the model that has been used to transfer the style. The model comprises an encoder and a decoder. The VGG network is good for feature extraction, so the encoder and decoder are VGG19 networks pretrained on the Microsoft COCO dataset. The maxpooling layer is replaced by Wavelet pooling, which reduces the loss of spatial information. It will only pass color information to further layers of the model. Moreover, the image is passed through this encoder-decoder pair only once, unlike some methods where multiple iterations occur. This will lead to the regeneration of content images with minimal content loss.

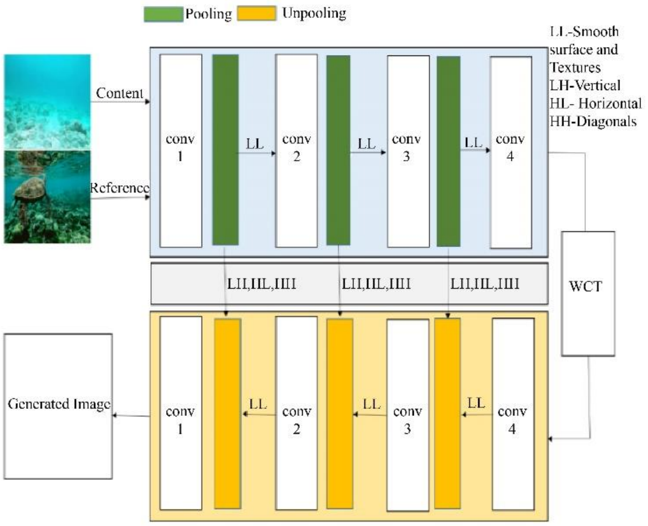

Fig. 2 Model structure

The task of the encoder is to extract the features from an image passed to it. In the encoder, all the extracted features relating to edges and corners, i.e. features that hold information about the actual contents of an image, are skipped. Only the features of color information are passed to further layers of an encoder. The wavelet pooling layer does this task. The extracted features containing content information are directly passed on to the decoder. Thus minimizing content loss throughout the process.

The task of the decoder is to combine features extracted by the encoder to regenerate the image with minimal loss in its content. Wavelet pooling and unpooling layers are responsible for this. They skip content information directly from the encoder to the decoder as it is. This reduces content loss.

3.3. Transfer Learning for Style Transfer

Transfer learning is done on the pre-trained VGG19 model to get the desired output. The maxpool layer is replaced by the wavepool layer [12]. The maximum element from the feature map's filter-covered area is chosen via max-pooling. Therefore, the output of the max-pooling layer is a feature map that contains the most prominent features of the prior feature map, whereas wavelet pooling extracts smooth surfaces and textures, vertical, horizontal and diagonal edges. For simplicity, each kernel's output is represented as LL, LH, HL and HH, respectively.

The max-pooling unspooling layer of the network has been replaced by wavelet pooling and unpooling. The COCO [11] pre-trained VGG-19 network [6] from the conv1_1 layer to the conv4_1 layer as the encoder has been used. The max-pooling layer has been replaced by wavelet pooling, in which the high-frequency components (LH, HL, and HH) are skipped to the decoder directly. The only low-frequency component (LL) is passed to the next encoding layer. The decoder's structure mirrors the encoder's, and the components are aggregated using wavelet unspooling. Between the encoder and decoder, the Whitening and Colouring Transfer (WCT) [13] layer is placed.

WCT has performed style transfer using arbitrary styles by directly matching the correlation between content and style in the VGG feature domain. Through the computation of Singular Value Decomposition (SVD), the content features are projected to the eigenspace of style features. The transferred features were fed into the decoder to create the final styled image. To make artistic style transfer better, using a multi-level stylization framework by applying WCT to multiple encoder-decoder pairs, as done by Gatys et al. and Lu et al. [3, 10], results in content loss. Therefore, the image is passed once through the encoder-decoder pair to reduce the content loss.

A function that accepts three arguments, such as input-content image C, produced image G, and layer L, whose activations were utilized to calculate the loss, has been used to compute the content loss.

The activation layer of a content image is *a[L](C)*, and the activation layer of the generated image is *a[L](G)*.

$$L_{content}(C,G,L) = \frac{1}{2}\sum_{ij}(a[L](C)_{ij} - a[L](G)_{ij})^2 \qquad (1)$$

The less the content loss, the better the results.

4. Results and Analysis

This section presents the results and discusses the performance of the developed application. It also provides a comparative study of the performance obtained using different methods.

4.1. Results With and Without Segmentation

It has been seen that the results obtained without segmentation are better than the results obtained with segmentation. The content loss varies between different images with and without segmentation. The output image generated without segmentation shows better results than that generated with segmentation. Figures 3, 4, 5 and 6 show the content image, style Image and results obtained with and without segmentation.

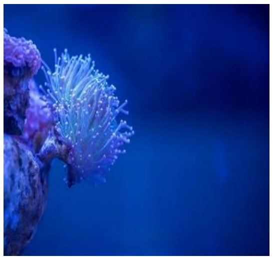

Fig. 3 Content Image

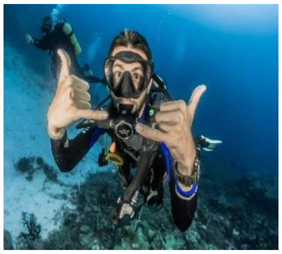

Fig. 4 Style Image

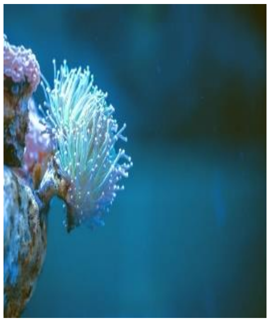

Fig. 5 Output Image With Segmentation

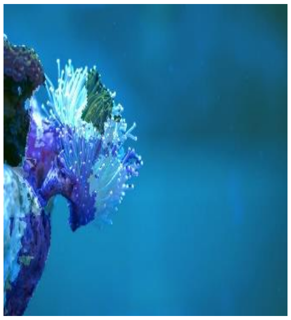

Fig. 6 Output Image Without Segmentation

4.2. CUDA and CPU

CUDA is used for using GPU. It is built on an NVIDIA graphics card. GPU requires less memory than CPU, and GPU is faster than CPU. Pytorch is used to switch between CPU and GPU platforms. If the graphic card does not support CUDA, it automatically gets switched to the CPU; however, when processing the image on the CPU, time is increased, and larger memory is required.

4.3. Histogram for Input and Output Image

Here, an image's histogram is a histogram of pixel intensity values. An image is formed using red, blue, and green colors. RGB histogram is used to show the image's pixel values. Since the sizes of the input and output images may vary, the Y-axis ranges may or may not vary. It depends on the size of the input image. Fig. 4.5 and 4.6 shows the histogram for input and output images, respectively.

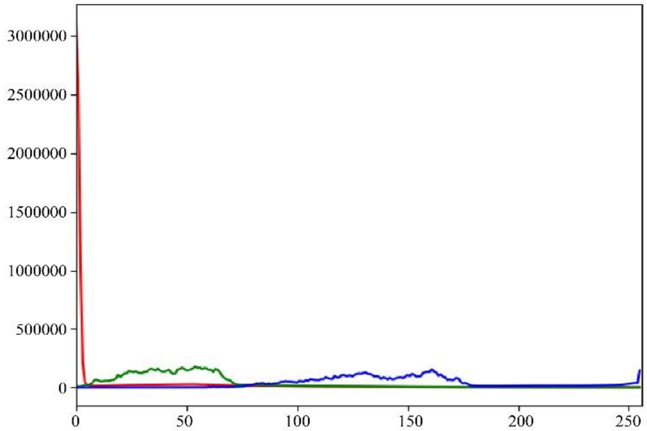

Fig. 7 Input image histogram

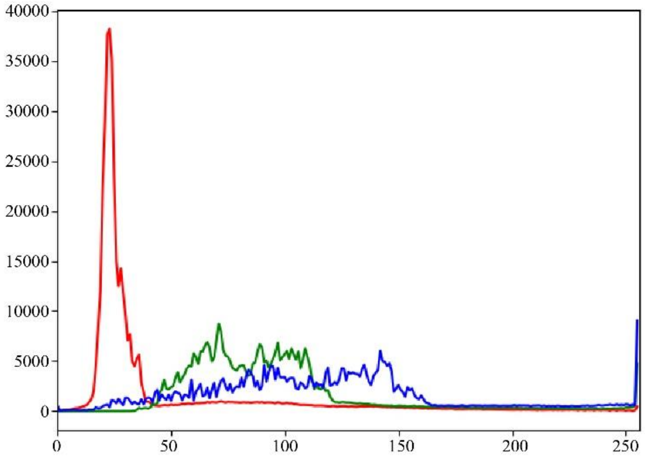

Fig. 8 Output image histogram

In Figure 7, it can be seen that the Y-axis scale varies because of the size. From the X-axis, it can be seen that the maximum range for blue color in Figure 7 is more than 150, whereas the maximum range for blue color in Figure. 8 is almost equal to 150. This shows that the contribution of the blue color is reduced. It can also be seen that the contributions of red and green colors have increased in Figure 8. The reduced contribution of blue color and increased contribution of red and green colors shows that blue haze from the input image is reduced and balanced by the mixture of red and green colors.

4.4. MaxPool Layer & Wavepool Layer

VGG19 network comprises a max pool layer. This layer extracts the maximum value from the image matrix. By using transfer learning, the max pool layer is replaced by the wave pool layer. The wave pool layer transfers the background details to the further layers of the network, whereas all the edges are transferred to the network's decoder.

Figures 9, 10, 11 and 12 show the content image, reference image and results obtained using the max pool layer and wave pool layer, respectively.

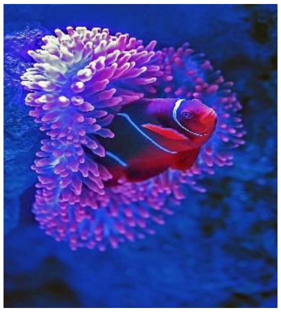

Fig. 9 Content Image

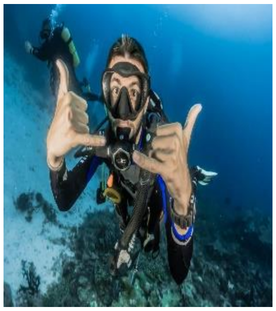

Fig. 10 Style Image

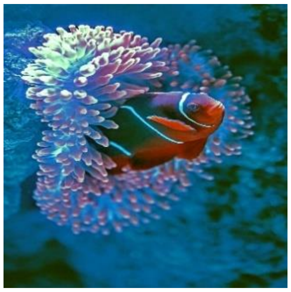

Fig. 11 Wavepool Layer Output

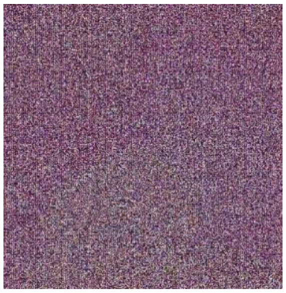

Fig.12 Maxpool Layer Output

From Figure 11 and Figure 12, it can clearly be seen that the results obtained using maxpool are poor. The average and variance of content loss of 12 underwater images were found to be 1.62% and 0.54%, respectively, whereas the content loss for results obtained using the max pool layer is more than 99%. Due to higher content loss for the max pool layer, this layer is being replaced by the wave pool layer. Using the wave pool layer, the image quality is improved, haze is removed, and the output image becomes clearer.

5. Conclusion

In this research, the image quality of underwater images has been improved after removing haze from the image. The generated output image is the input image coated as per the style of the reference image, i.e. in the style of a good-quality image. The average content loss of 12 generated good-quality underwater images was 1.62%. All the essential details of the image have been stored in the initial stages of the networks, and the background of the input image has been improved. The final output image obtained has contents similar to the input image, but the haze has been removed, and the image has become easier to understand. It has also been noticed that the results obtained without segmentation are better than those obtained with segmentation.

In future, images can be enhanced by improving the segmentation method and facilitating the subject-wise color transfer.

References

[1] Mark Shortis, Euan Harvey, and Dave Abdo, *A Review of Underwater Stereo-Image Measurement for Marine Biology and Ecology Applications*, Oceanography and Marine Biology, 1st ed., pp. 1-36, CRC Press, 2009. [Google Scholar] [Publisher Link]

[2] Chongyi Li, Jichang Guo, and Chunle Guo, "Emerging from Water: Undewater Image Color Correction Based on Weakly Supervised Color Transfer," *IEEE Signal Processing Letters*, vol. 25, no. 3, pp. 323-327, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[3] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge, "Image Style Transfer Using Convolutional Neural Networks," *2016 IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, pp. 2414-2423, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[4] Zia-ur Rahman, Daniel J. Jobson, and Glenn A. Woodell, "Retinex Processing for Automatic Image Enhancement," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 100-111, 2004. [CrossRef] [Google Scholar] [Publisher Link]

[5] Yang Wang et al., "A Deep CNN Method for Underwater Image Enhancement," *IEEE International Conference on Image Processing*, Beijing, China**,** pp. 1382-1386, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[6] Karen Simonyan, and Andrew Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv*, pp. 1-14, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[7] Ian J. Goodfellow et al., "Generative Adversarial Networks," *arXiv*, pp. 1-9, 2014. [CrossRef] [Google Scholar [Publisher Link]

[8] Md. Jahidul Islam, Youya Xia, and Junaed Sattar, "Fast Underwater Image Enhancement for Improved Visual Perception," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3227-3234, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[9] Ming Lu et al., "A Closed-Form Solution to Universal Style Transfer," *2019 IEEE/CVF International Conference on Computer Vision*, Seoul, Korea (South), pp. 5951-5960, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[10] Md Jahidul Islam et al., "Semantic Segmentation of Underwater Imagery: Dataset and Benchmark," *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Las Vegas, NV, USA, pp. 1769-1776, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[11] Tsung-Yi Lin et al., "Microsoft COCO: Common Objects in Context," *Proceedings, Part V 13th European Conference, Computer Vision – ECCV*, Zurich, Switzerland, pp. 740-755, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[12] Travis Williams, and Robert Li, "Wavelet Pooling for Convolutional Neural Networks," *ICLR Conference Blind Submission*, pp. 1-12, 2018. [Google Scholar] [Publisher Link]

[13] Yijun Li et al., "Universal Style Transfer via Feature Transforms," *31st Conference on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA, pp. 1-11, 2017. [Google Scholar] [Publisher Link]

[14] Phillip Isola et al., “Image-to-Image Translation with Conditional Adversarial Networks,” *2017 IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, pp. 5967-5976, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[15] Jun-Yan Zhu et al., “Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks,” *2017 IEEE International Conference on Computer Vision*, Venice, Italy, pp. 2242-2251, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[16] Florian Shkurti et al., “Multi-domain Monitoring of Marine Environments using a Heterogeneous Robot Team,” *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura-Algarve, Portugal, pp. 1747-1753, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[17] Brian Bingham et al., “Robotic Tools for Deep Water Archaeology: Surveying an Ancient Shipwreck with an Autonomous Underwater Vehicle,” *Journal of Field Robotics*, vol. 27, no. 6, pp. 702-717, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[18] Md Jahidul Islam, Marc Ho, and Junaed Sattar, “Understanding Human Motion and Gestures for Underwater Human-Robot Collaboration,” *Journal of Field Robotics*, vol. 36, no. 5, pp. 851-873, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[19] Shu Zhang et al., “Underwater Image Enhancement via Extended MultiScale Retinex,” *Neuro Computing*, vol. 245, pp. 1-9, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[20] Cameron Fabbri, Md Jahidul Islam, and Junaed Sattar, “Enhancing Underwater Imagery using Generative Adversarial Networks,” *IEEE International Conference on Robotics and Automation*, Brisbane, QLD, Australia, pp. 7159-7165, 2018. [CrossRef] [Google Scholar] [Publisher Link]