Assessment of the Utility of Image Enhancement For Underwater Tasks

A CAPSTONE PROJECT REPORT SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL OF THE UNIVERSITY OF MINNESOTA

BY

AMITABHA DEB

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE

JUNAED SATTAR

May, 2024

Contents

Acknowledgements

I would like to extend my deepest gratitude to my academic advisor, Prof. Junaed Sattar, whose unwavering support, encouragement, and guidance have been invaluable throughout my journey as an MS student.

My heartfelt appreciation goes to my esteemed exam committee members, Prof. Maria Gini and Prof. Nikolas Papanikolopoulaos, for graciously dedicating their time and providing invaluable insights. Additionally, I would like to express appreciation to Travis and Nicole for their guidance and support in both academic and administrative matters.

I am indebted to my dedicated collaborators, Sadman Sakib Enan, Sakshi Singh, Ying-Kun Hu, and Rishi Mukherjee, whose contributions to resources and ideas have significantly enriched my work.

I am also thankful to Prof. Maria Gini, Prof. Volkan Isler, Prof. Dongyop Kang, Prof. Ju Sun, and Prof. Nicholas Johnson for their exceptional instruction in critical subjects such as Artificial Intelligence, Machine Learning, Computer Vision, and Natural Language Processing.

Finally, I wish to express my heartfelt appreciation to my cherished family and friends. To my parents, sister, girlfriend, and close friends, your encouragement in times of success, steadfast presence during moments of adversity, and unwavering support in every conceivable manner have been truly uplifting.

Abstract

Recently, learning-based image enhancement methods have demonstrated promising performance in underwater conditions. However, these models often come with a high computational cost. In this study, we aim to evaluate the relevance of image enhancement based on the specific task at hand. Additionally, we strive to develop a predictive method to determine whether image enhancement is necessary. Enhanced images were generated using 'Fast Underwater Image Enhancement for Improved Visual Perception' (FUnIE-GAN) [1] and were evaluated quantitatively and visually. Further, bounding boxes were predicted using object detection models for both original and enhanced images. These predictions were analyzed and binary labels were created for the images to predict the necessity of image enhancement for the task. Despite our efforts, we have encountered challenges in devising an effective predictive method due to the limitations of the training set. This idea does show promise for further exploration with a specialized dataset to train all the models involved.

Chapter 1

Introduction

Underwater vehicles are employed in a variety of tasks which are broadly classified into commercial, oceanographic research, military, and engineering research missions [2]. These vehicles are classified into Remotely Operated vehicles(ROVs) and Autonomous underwater vehicles(AUVs) based on the manner they navigate. In either case, these robots rely on vision for their control system feedback and to perform computer vision tasks such as object detection, pose estimation, and visual odometry to achieve their end goal.

Several factors such as the turbidity of water due to floating sedimentation and debris, the depth of the scene, and others cause severe degradation in camera output in underwater conditions. The image appears blueish and blurry as a result of poor visibility, ambient light, and frequency-dependent scattering and absorption. [3]

Traditionally underwater image processing focused on recovering the original image from the observed image by incorporating knowledge from underwater optical properties [4], dehazing algorithms [5], and color adjustment. Recently, learning-based methods such as Convolutional Neural Networks(CNNs) and Generative Adversarial Networks(GANs) provide state-of-the-art performance in approximating the underlying mapping to enhance the quality of images [6] [7]. Specifically for underwater scenes GAN-based models such as 'Fast Underwater Image Enhancement for Improved Visual Perception' (FUnIE-GAN) [1] provide enhanced images that improve the performance of underwater object detection, saliency prediction, and pose estimation.

Due to the wide range of variability and difficulties in underwater scenes large-scale data and models are necessary to capture this variability. Thus, the image enhancement

models are often computationally heavy, and using these algorithms in unnecessary conditions not only results in excess energy expenditure but also induces noise to undistorted images. Further, for different tasks, the quality of the image required to make inferences varies. For example, if we are trying to detect large marine animals a moderately blurry image might be sufficient but that might not be the case if we are detecting a small object such as a cup.

In this work, we attempt to determine when to use these image enhancement filters for real-time applications based on the specific tasks undertaken by underwater robots. By discerning the requirements of each task, we aim to devise a framework that dynamically determines the effectiveness of employing these filters. The problem of determining whether image enhancement is required can be treated as a binary classification problem, where label 0 can be assigned if no enhancement is needed and 1 if enhancement is needed. This approach aims to ensure that image enhancement is employed judiciously, enhancing the efficiency of underwater robotic operations.

Chapter 2

Related Works

Several methods based on deep adversarial networks [8] [9] have considerably increased the performance of enhancing underwater colored images in late 2010s. These methods leveraged GANs and structural loss functions to produce images of varied quality. However, limitations due to their small-scale training dataset and slow inference hindered their practical usability. This has been addressed by 'Fast Underwater Image Enhancement for Improved Visual Perception' (FUnIE-GAN) [1], a model trained on large-scale underwater data.

FUnIE-GAN is a conditional GAN, which has a generator network based on UNet [10] and a discriminator based on Markovian Patch-GAN [11]. The network achieves efficiency by employing fewer skip connections within the generator and presuming pixel independence beyond the patch level in the discriminator. These adjustments result in the reduction of parameters within the model enabling real-time usage. The conditional adversarial loss function paired with L1 loss and content loss preserves the global contrast and sharpness. The model successfully demonstrated color and contract rectification in underwater images resulting in better object detection, pose estimation, and saliency prediction. [1]

In underwater robotics, it is often desired to enhance the images while increasing their resolution for better analysis and interpretation. In the paper "Simultaneous Enhancement and Super-Resolution of Underwater Imagery for Improved Visual Perception" [12], the authors introduce a learning-based algorithm called DEEP-SESR, which can produce enhanced images with up to 4x resolution. DEEP-SESR is a residual-in-residual generative network that has residual dense blocks (RDBs), a feature extraction network (FENet), and an auxiliary attention network (AAN). The enhanced images are produced by a con-

volution layer following the FENet, while the higher resolution images are produced by upsampling the output through a series of convolution and convolution layers.

An underwater color image quality evaluation (UICQE) [13] introduces a real-time algorithm to evaluate underwater images based on chroma, contrast, and saturation. It can successfully predict the difference between the enhancement results with better correlation. The UCIQE metric is given by,

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

where σ_c is the standard deviation of chroma, con_l is the contrast of luminance and μ_s is the average of saturation, and c_1, c_2, c_3 are weighted coefficients.

From a general point of view, UICQE gives a great basis to decide whether image enhancement is required. The scores can be plotted for images from a variety of scenes and a threshold can be set to trigger the enhancement algorithms. Further, learning-based algorithms can be employed to predict the UICQE score of the potential enhanced image from the original image, to predict its usefulness. In the work, 'Machine Vision for Improved Human-Robot Cooperation in Adverse Underwater Conditions' [14], the author leverages entangled light (E-light) discriminator, a lighter discriminator module that is entangled with the discriminator (D) module of the GAN while training. This module learns to assess image quality guided by the D, which can be later decoupled for real-time analysis.

In practical scenarios, most underwater robots have predetermined tasks such as object detection, pose estimation, and saliency prediction. Using the knowledge of the tasks in hand and pre-analysis of the performance of enhanced images in such situations might help leverage the enhancement modules more efficiently. We explore a few datasets to further investigate these insights.

The Trashcan dataset [15] consists of 7000+ underwater images with various instances of trash, ROVs, and undersea flora and fauna. The objects are annotated using both bounding boxes and segmentation masks and provide a platform for robust underwater trash detection algorithms. The Common Objects Underwater (COU) dataset is a recent high-quality dataset that consists of common objects in pool and lake conditions. It is designed to evaluate the performance of various computer vision algorithms underwater.

Conventionally, bounding boxes are employed to indicate detected objects within an image. Segmentation aids in refining the localization of the detected object, providing more precise boundaries. The YOLACT model is a fully convolutional model that performs real-time instance segmentation [16]. It can be used to generate bounding boxes with high accuracy without the need for multi-scale training, optimized anchor boxes, cell-based regression encoding, and objectness scores. The YOLO model archives rapid object detection by running only a single convolutional network on the image. This unified model predicts the bounding boxes and class labels and thresholds them with the model confidence [17]. For pose estimation, YOLO uses object keypoint similarity as a metric for training and predicts a set of keypoints and their confidence scores [18].

For image classification, Convolutional Neural Networks (CNN) are very effective due to their ability to learn hierarchical features [19]. Further, with the introduction of deep residual learning and skip connections, the issue of vanishing gradient was addressed. This advancement revolutionized deep learning by enabling the training of much deeper networks. Each residual block consists of two or three convolutional layers along with a shortcut connection between its input and output. These skip connections prevent information loss and model optimization for deep networks becomes significantly easier. Moreover, this breakthrough led to state-of-the-art performance in image classification and other computer vision tasks[20].

Chapter 3

Methods

Underwater images are mostly degraded and enhancing them can improve color correction and increase the sharpness. But these distorted images can sometimes be sufficient for the end task. The objective is to determine when image enhancement is needed to eliminate unnecessary computations and save energy. Understanding the specific task at hand aids in the decision process as some tasks such as segmentation or pose detection rely heavily on the details of the image. The problem of identifying whether enhancement is required for a particular scene to perform a given task is framed as a binary classification problem.

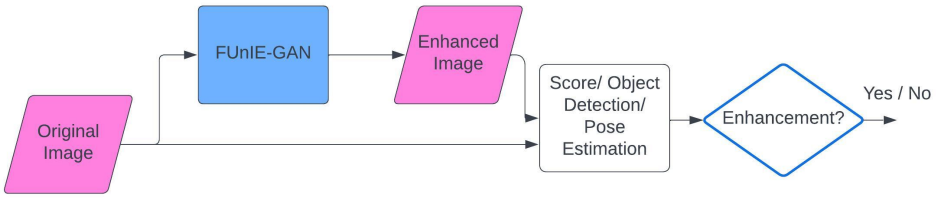


Figure 3.1: Overview of the Proposed Method: Underwater images and their corresponding enhanced versions (generated using FUnIE-GAN) are evaluated using detection and estimation models. The success of the models on both images is recorded, providing insights into the necessity of enhancement. This mapping can be learned using image classification techniques.

A task-based quality assessment strategy was developed to achieve the goal which involves the following steps.

3.1 Collecting underwater images for various tasks and various conditions

The first step in our analysis is to acquire underwater datasets. In this work, we place significant emphasis on object detection, as it is a common task in underwater imagery analysis. Initially, we utilize the Trashcan dataset, which consists of over 7000 images of trash, ROVs, and a wide variety of undersea flora and fauna, annotated with both bounding boxes and segmentation masks. However, due to limitations in the Trashcan dataset in terms of image quality and scenes, we subsequently shifted our focus to a newer dataset called COU, which provides bounding boxes for the images. For COU, we specifically use the images of the 'cup' class as it has the highest intersection with the COCO dataset (which would be used for training the object detection model). The primary metric for evaluating performance in this context is the Intersection over Union (IoU) of bounding boxes. Additionally, we explore the new diver pose dataset 6.2 for diver keypoint detection.

3.2 Getting enhanced images for these tasks using FUnIE-GAN

FUnIE-GAN was employed to generate enhanced images for all the images in each of the datasets. Since FUnIE-GAN has an output size of 256 x 256 the input images are also resized to 256 x 256 pixels to ensure that no bias is developed for the higher resolution images.

Figure 3.2: Sample of original (top row) and enhanced (bottom row) images for Trashcan (left), COU 'cups' (middle), and diver pose datasets (right).

The UCIQE score for the original and enhanced images is then calculated for the entire dataset. The trained coefficients used from the paper [13] are c1=0.4680, c2=0.2745, c3=0.2576

3.3 Inference on both original and enhanced images

Object detection using Trashcan and Yolact

A YOLACT model trained on original images from the Trashcan dataset is used to make inferences on the original and enhanced images. The bounding box predictions are then used to compute the IoU scores against the labels.

Object detection using COU and Yolov8

A pre-trained YOLOv8 model is fine-tuned on COCO 'cup' and COU 'cup' (COU-Cups) pool images and is used to make bounding box predictions. COU 'cup' images, obtained by artificially placed cups in Lake Superior (that were subsequently removed), are then used as a testing set to calculate IoU scores for analysis. This model is initially trained on 9189 images containing cups from the COCO dataset for 25 epochs. It was further fine-tuned on pool images containing cups from COU. Finally, testing was conducted on lake images containing cups from COU to examine the need for enhancement.

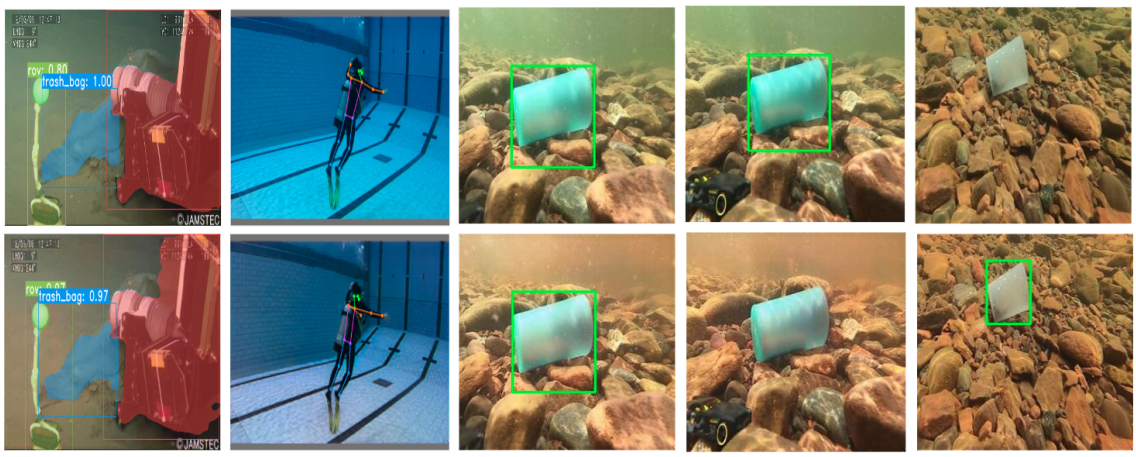


Figure 3.3: Detections on original (top row) and enhanced (bottom row) images for Trashcan (leftmost), diver pose dataset (2nd column), and COU 'cups' (last three columns).

Pose estimation using yolo v7

An off-the-shelf pre-trained YOLOv7 pose model is used to predict 17 keypoints on the divers. The predictions are made on the new diver pose dataset 6.2 and the Mean Eu-

clidean Distance computed, as well as visual analysis, is conducted.

3.4 Creating a dataset from success and failure

From the results of the above detectors the four cases identified are identified are:

Original Image	Success for original image	Success for enhanced image	Action
Case 1	Yes	Yes	Both images pass so no need for enhancement
Case 2	No	Yes	Only the enhanced image works, the enhancement module should be triggered
Case 3	No	No	Both images fail so enhancement does not improve detection
Case 4	Yes	No	Should not happen ideally

We work on only object detection from here on. If the detector cannot detect an object that it has been trained on it can be considered a failure. However, in many cases, the detector might predict bounding boxes for the objects on both original and enhanced images. So, we consider the fitness of these bounding boxes as a measure of success.

We have analyzed the outputs based on the percentage improvement of IoU scores of the bounding boxes on the enhanced image. Thus labels have been generated as '1' i.e. the image needs enhancement when the IoU score of the object detected in the enhanced image is either 5% or 10% more than the one in the original image. All other images are labeled as '0'.

3.5 Design a deep learning architecture to learn the patterns

Now we try to learn from the original images and the generated labels to predict when enhancement is needed using learning-based models. Ideally, on a robot, this model should be running in real-time to predict if enhancement is needed and should consume low energy. Thus the model should be very light and efficient.

First, we explore if learning a mapping from the input images to the binary labels is possible using a deep residual convolutional network. We employ a ResNet-18 model, consisting of 18 layers that include convolutional, pooling, and fully connected layers, along with shortcut connections. We utilize pre-trained weights from the Imagenet dataset.

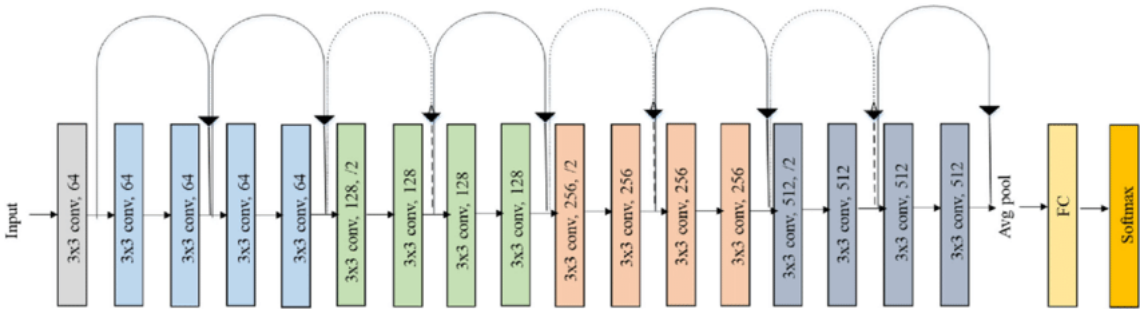


Figure 3.4: ResNet-18 architecture with skip connection after every 2 convolutional layers [21]. The output from the convolutional layers is fed to a fully connected layer which predicts the binary labels.

Subsequently, these weights are fine-tuned on the binary labels generated from the previous section, allowing the model to learn whether image enhancement is needed for the input frame.

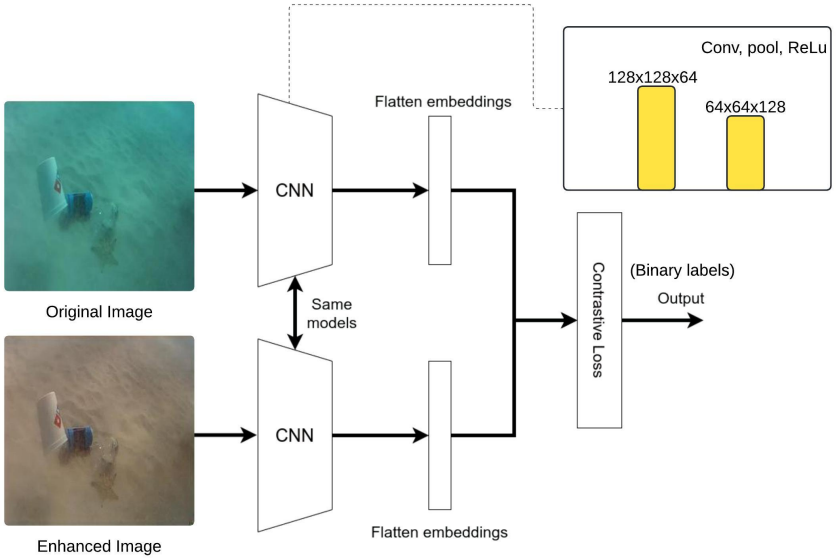


Figure 3.5: Siamese architecture using contrastive loss function to learn the difference between the original and the enhanced image. This difference should correspond to our generated labels. Hence, the model should learn to predict whether enhancement is needed.

Further, we explore the possibility of leveraging a Siamese-based neural network to perform paired learning using the original and enhanced image pair. We formed a Siamese network consisting of two blocks of convolutional max-pooling layers, and two fully connected networks, to capture the distinctions between the original and enhanced image pairs. The resulting output is subsequently guided by a contrastive loss function for the prediction of binary labels.

Chapter 4

Results and Discussions

Enhanced Image Analysis

On average, FUnIE-GAN takes approximately 20 milliseconds to generate enhanced images using a Tesla T4 GPU. However, as observed in figure 3.2, the enhanced images do not visually appear to be better and exhibit a yellow tint. This tint arises because FUnIE-GAN is trained using the EUVP dataset [1], which predominantly contains ocean images with a bluish tint. Consequently, FUnIE-GAN corrects this bluish tint, inadvertently introducing a yellow tint in the process.

We further compare the generated images and original images using the UCIQE scores. We obtain a mean score of 0.4815, and 0.4867 on the original and enhanced images of the Trashcan dataset.
We obtain a mean score of 0.5261, and 0.5391 on the original and enhanced images of the diver pose dataset 6.2.

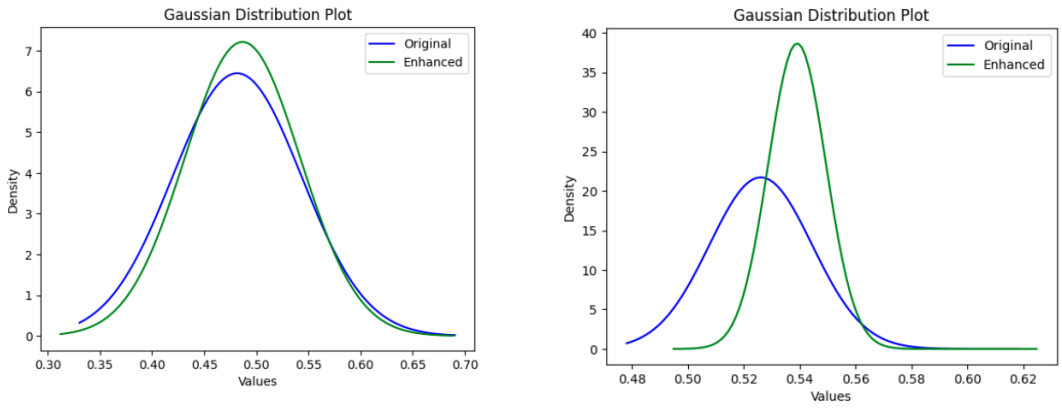


Figure 4.1: UCIQE scores for Trashcan dataset(left) and the diver pose dataset(right).

Thus, we see that even though there is a slight improvement in the UCIQE scores, FUnIE-GAN does not perform at the level it is capable of. This can be rectified by training FUnIE-GAN with representative datasets.

Detection Analysis

As we see from the figure 3.3, the detection does not show significant improvement for the enhanced images. In the case of pose estimation on the diver pose dataset, the predicted keypoints seem better on the original data. This is probably because currently, the new diver dataset consists of only pool images, on which FUnIE-GAN is not trained. Enhancement adds unnecessary noise to the image. YOLOv7 is also unable to distinguish between the left and right parts of the body in various scenes. This makes it very hard to evaluate it against the ground truth. This can be possibly addressed by fine-tuning the prediction model on the annotated images and adding ocean images to the dataset for better-enhanced images.

	Training Set		Validation Set	
% Improvement	Class 1	Class 0	Class 1	Class 0
10	81	5984	56	1091
5	329	5736	161	986
0	1822	4243	412	735

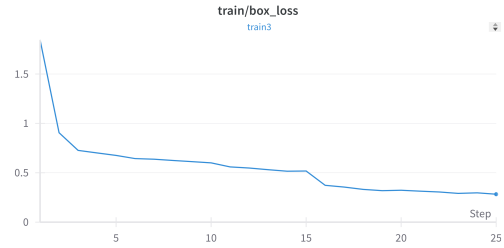
Table 4.1: Class count based on YOLACT results on Trashcan

The above table 4.1 is created using the bounding box predictions from YOLACT on the Trashcan dataset. The class labels are created using the percentage improvement threshold. Class 1 in the first row corresponds to an improvement of 10% or more on IoU score of the predicted bounding box of any of the objects. From the table, we can see that the detection is much worse in the case of enhanced images. This can be attributed to the fact that the YOLACT model has been trained on the original images and it has learned the features for the same.

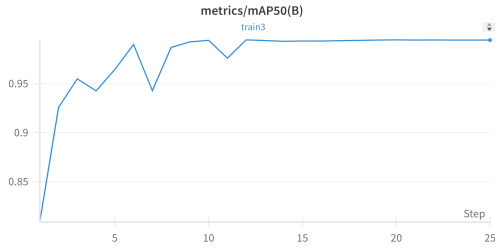
We trained a YOLOv8 bounding box predictor on the pool images from COU-Cups. The testing set comprising of lake images was kept separate to prevent model bias that we encountered while working with Trashcan. We achieved a mAP50 score of 0.995 and a mAP50-95 score of 0.979 on the training set and a mAP50 score of 0.995 and a mAP50-95 score of 0.981 on the validation set.

After training we use the model to draw inferences on the testing set. However, the testing set comprised only lake images and was also not suitable for FUnIE-GAN because of its training set. We form a similar table 4.2 as above to gauge the benefit of enhancement.

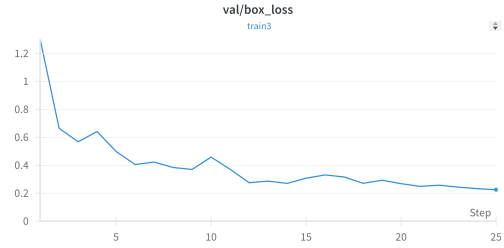
Out of a total of 636 lake images in COU-cups, cups were detected in 287 original



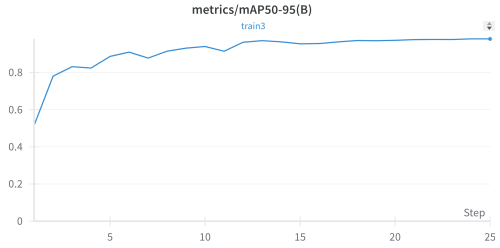
(a) Training box loss



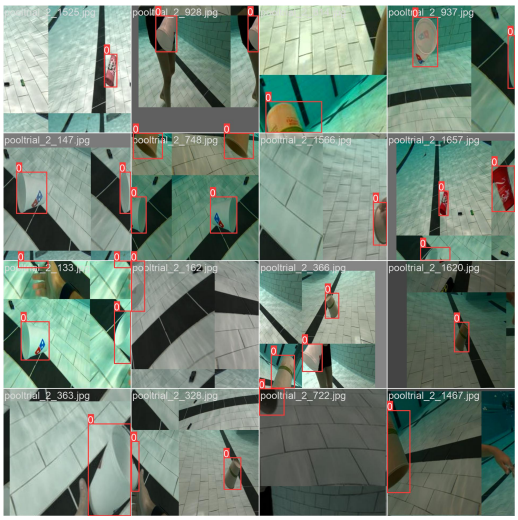
(b) mAP50 metric during training



(c) Validation box loss



(d) mAP50-95 metric during training



(e) Predictions on training set

(f) Predictions on validation set

Figure 4.2: YOLOv8 training results on pool images containing cups (COU dataset)

images and 248 enhanced images with a confidence of 0.4. The average IoU score of detections in the original image is 0.839 and in the enhanced images is 0.826. The average Confidence Score of detections in the original image is 0.719 and in the enhanced images is 0.711. This again shows that since the FUnIE-GAN model has not learned how to enhance lake images, unnecessary noise is introduced, degrading the image quality.

Classification Analysis

Based on the binary labels generated as mentioned in the previous section, we try to train a ResNet network. We use a pre-trained ResNet-18 model which has 18 convolutional layers and is trained on million-plus images to detect 1000 object categories. We then fine-tune

% Improvement	Class 1 based on IoU	Class 1 based on Confidence
10	24	38
5	30	57
0	130	104

Table 4.2: Class count based on YOLOv8 results on COU-cups test set

the model for 15 epochs using a weighted cross-entropy loss function on the Trashcan dataset. The model has a test f1 score of 0.51, which can be attributed to the quality of the labels generated. Based on the detection analysis we can further say that the few samples where enhanced images yielded better results were random.

We further tried to leverage a Siamese base network [22] with two blocks of convolutional and max pooling layers to learn the difference between the original and enhanced image pair. The output is then fed to a fully connected network to predict the binary labels. The major issue encountered is translating this model to a single input (camera output) model for real-time use. We have observed potential better learning in this case and achieved a f1 score of 0.62 on the trashcan test set.

Based on our evaluations, we draw the following conclusions:

- Despite the computational efficiency of FUnIE-GAN, its performance in enhancing underwater images is constrained due to training set limitations.
- The UCIQE scores indicate slight improvement post-image enhancement; however, visual analysis reveals inconsistencies in image quality.
- Detection analysis demonstrates that enhancements do not significantly benefit object detection and pose estimation tasks, potentially due to noise introduced by FUnIE-GAN.
- Classification results show limited success, with minor improvements observed in prediction performance.

Chapter 5

Conclusion

Task-based approaches for image quality enhancement have not been explored much previously. This approach is more practical as different tasks need different quality images to be effective. The approach can then be generalized by generating comprehensive binary labels for the need for enhancement for various tasks.

We see that the major limiting factor has been our training set for the enhancement algorithm as well as the detectors. In the future, this can be addressed by creating a dataset specialized for this workflow. That would require creating a dataset where the test set for the detectors has a large intersection with the training set of the enhancement model. Further, the detectors should not be trained on poor-quality images to prevent unnecessary bias. Lastly, we should consider training an image enhancement model with a larger output size (currently 256 x 256) so that we obtain a higher resolution enhanced image that is at par with current camera resolutions.

Chapter 6

Contributions

6.1 Developing a Web Application for FUnIE-GAN, DEEP-SESR, and SVAM

The image enhancement models often tend to have an overwhelming code base and is hard for people from other domains to access these models. For convenience and increased usage by people who want to use them for research and other applications, We have developed a flask-based app to easily interact with these models through the web and get the output. This lays down a platform to easily analyze the outputs of the model for future work. Github

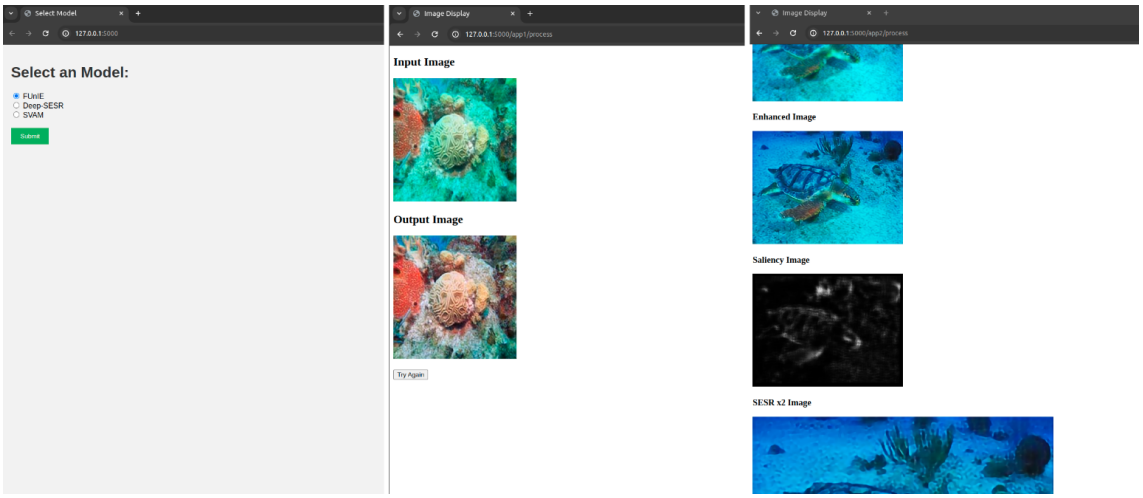


Figure 6.1: Webapp for FUnIE-GAN, DEEP-SESR, and SVAM.

6.2 Creating a Dataset of Diver Pose Keypoints

We are creating a dataset of diver poses for stereo-image pairs. For that, we have identified 20 key points on the diver that we have labeled. The labeling work is now ongoing and

the current progress is around 4000 images. The labeled dataset includes stereo pairs of images from indoor pools, lakes in Minnesota, and ocean images from Barbados. The end goal is to have a dataset with 10000+ images. This dataset should be sufficient for detecting 3D diver pose in a variety of scenes and has a lot of applications.

Figure 6.2: Samples from diver pose dataset.

Bibliography

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

[2] Louis Whitcomb. Underwater robotics: Out of the research laboratory and into the field. volume 1, pages 709 – 716 vol.1, 02 2000.

[3] G. Dudek, M. Jenkin, C. Prahacs, A. Hogue, J. Sattar, P. Giguere, A. German, Hui Liu, S. Saunderson, A. Ripsman, S. Simhon, L.-A. Torres, E. Milios, P. Zhang, and I. Rekletis. A visually guided swimming robot. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3604–3609, 2005.

[4] Weilin Hou, Deric Gray, Alan Weidemann, Georges Fournier, and J.L. Forand. Automated underwater image restoration and retrieval of related optical properties. pages 1889 – 1892, 08 2007.

[5] John Y. Chiang and Ying-Ching Chen. Underwater image enhancement by wavelength compensation and dehazing. *IEEE Transactions on Image Processing*, 21(4):1756–1769, 2012.

[6] Andrey Ignatov, Nikolay Kobyshev, Radu Timofte, Kenneth Vanhoey, and Luc Van Gool. Dslr-quality photos on mobile devices with deep convolutional networks, 2017.

[7] Yu Sheng Chen, Yu-Ching Wang, Man-Hsin Kao, and Yung-Yu Chuang. Deep photo enhancer: Unpaired learning for image enhancement from photographs with gans. 06 2018.

[8] Cameron Fabbri, Md Jahidul Islam, and Junaed Sattar. Enhancing underwater imagery using generative adversarial networks. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7159–7165, 2018.

[9] Xiaoli Yu, Yanyun Qu, and Ming Hong. Underwater-gan: Underwater image restoration via conditional generative adversarial network. In *CVAUI/IWCF/MIPPSNA@ICPR*, 2018.

[10] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[11] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks, 2018.

[12] Md Jahidul Islam, Peigen Luo, and Junaed Sattar. Simultaneous enhancement and super-resolution of underwater imagery for improved visual perception. *CoRR*, abs/2002.01155, 2020.

[13] Miao Yang and Arcot Sowmya. An underwater color image quality evaluation metric. *IEEE Transactions on Image Processing*, 24(12):6062–6071, 2015.

[14] Md. Jahidul Islam. Machine vision for improved human-robot cooperation in adverse underwater conditions. 2021.

[15] Jungseok Hong, Michael Fulton, and Junaed Sattar. Trashcan: A semantically-segmented dataset towards visual detection of marine debris, 2020.

[16] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation, 2019.

[17] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection, 2016.

[18] Debapriya Maji, Soyeb Nagori, Manu Mathew, and Deepak Poddar. Yolo-pose: Enhancing yolo for multi person pose estimation using object keypoint similarity loss, 2022.

[19] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks, 2015.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[21] Farheen Ramzan, Muhammad Usman Khan, Asim Rehmat, Sajid Iqbal, Tanzila Saba, Amjad Rehman, and Zahid Mehmood. A deep learning approach for automated diagnosis and multi-class classification of alzheimer's disease stages using resting-state fmri and residual neural networks. *Journal of Medical Systems*, 44, 12 2019.

[22] Gregory R. Koch. Siamese neural networks for one-shot image recognition. 2015.

All code and data can be found at: Github