ROVer Optometry

Aditya Kulkarni Farman Ali Sruti Karthikeyan Parisa Nawar

Introduction

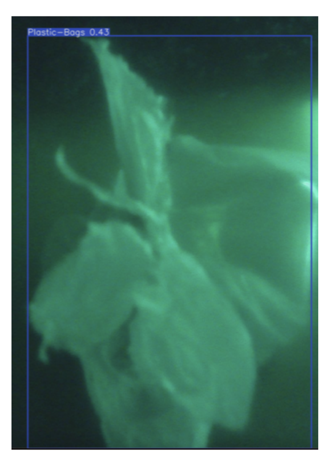
To achieve high-accuracy underwater image classification, we employed a low light analog camera mounted inside an airtight waterproof container. The contraption enabled us to capture images of underwater debris, such as plastic bags, metal cans and various other plastic objects. These images were then processed using a convolution neural network (CNN) model, that had been trained to detect trash in a marine-like environment.

Methods

1. Research | Before we started to create a model, we utilized online resources regarding YOLOv7, PyTorch, and OpenCV. Furthermore, our prior knowledge was aided through numerous tutorial articles, pages of official documentations, and academic research papers.

2. Creating the dataset | We compiled an underwater dataset with over 13,000 images of underwater trash such as plastic wrappers and water bottles in order to train our model.

(a) Example of an image in the dataset.



(b) Image of the bounding box around the experimental environment.

3. Training the model | Utilizing a custom Linux machine, our team was able to train 200 epochs with a batch-size of 4 in order to provide the proper weights and bias' for the resulting detection mechanism.

4. Simulating a marine environment | We prepared a low light analog camera to be stored inside Tupperware that was sealed tight in order to be waterproof underwater. We further filled up a fish tank with diluted water and utilized food coloring to emulate the environment represented in the underwater dataset images.

5. Testing the dataset | Once the YOLOv7 model was done training, and the camera and environment were prepared, we started our experimentation. We placed various plastic wrappers and items underwater in order to test the accuracy of our underwater camera on the provided dataset we had used. Our team repeated the trial and error process until an accuracy score that was satisfactory was met.

Computer Vision and Object Detection

We believe that **YOLOv7** is the most precise, in tune version of the YOLO architecture when it comes to underwater object detection.

The constant currents in the ocean ensures that everything is constantly moving. It is unrealistic to think that something in the ocean will stay there forever, which is why we are prioritizing real-time object detection. The currents tend to vary in terms of speed due to wind, temperature or any natural disaster, any object moving at such turbulent speed must be identified as soon as possible. To prioritize the efficiency at which an object is detected under such conditions, we needed a model that prioritized low-inference times.

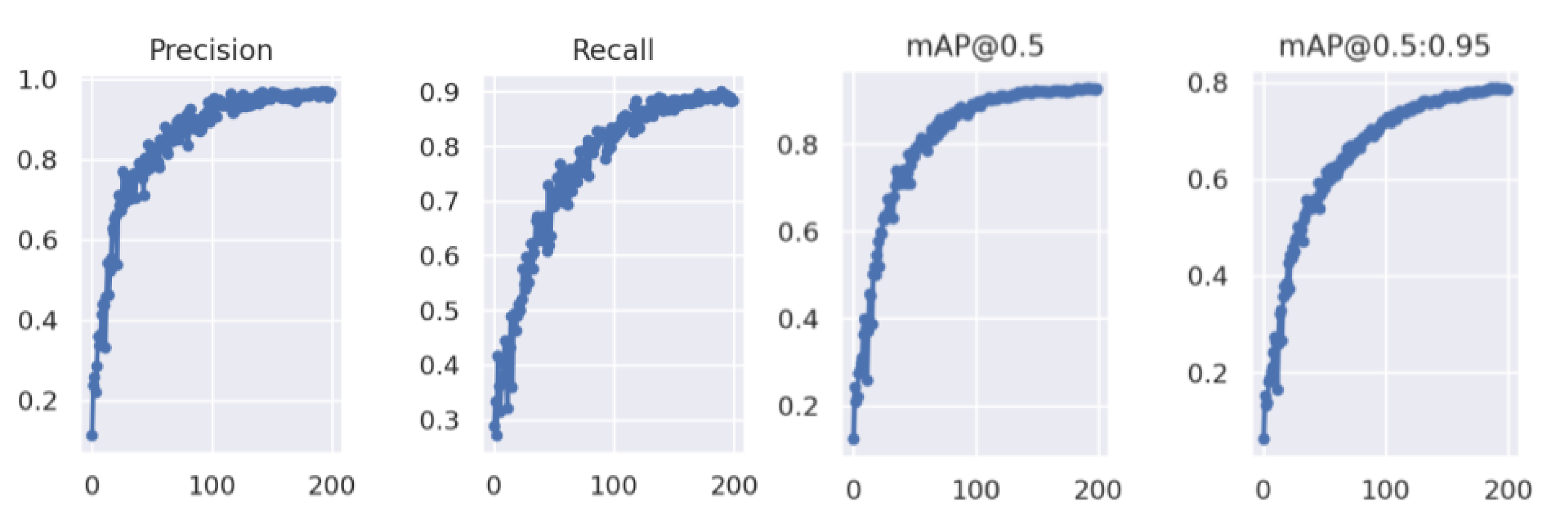
YOLOv7 not only provides real-time object detection and low-inference times, but it also works well underwater [1].

Results

After the model was trained even further we evaluated its performance using several metrics such as precision, recall, and mAP (Mean Per Average). When the IOU (Intersection Over Union) are at 0.5 and at 0.95. **Higher IOU readings generally entail more accurate predictions.**

The data below shows that training our models to 200 epochs from 50, drastically improved its accuracy. The model gave us a validation precision score at 96.05 and a recall score of 90.85. With mAP scores of 88.5 at an IOU of 0.5 and 79.32 at an IOU of 0.95.

These improvements to our data confirm that our approach to underwater object detection are correct.



(a) Graphs detailing the statistics of the model.

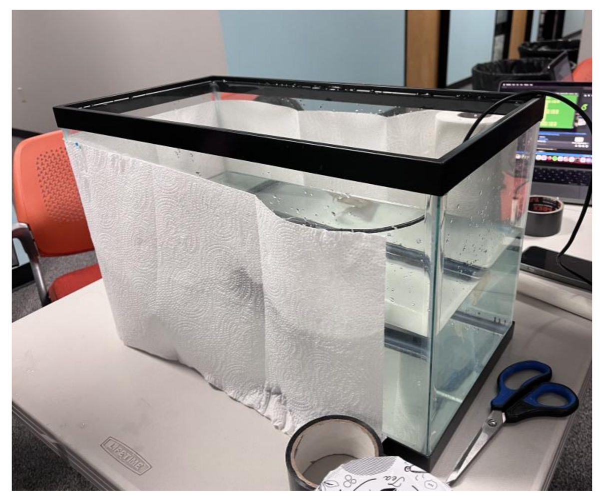
Analysis

Based on the precision and recall graphical representations, we can conclude an estimated 96% precision of our model when classifying and detecting objects, and an estimated 91% recall of our model. Though these numbers are not up to industry standard, they are satisfactory in terms of an undergraduate research project. Thus, our team felt content with the provided results gained from training and utilizing the dataset and model.

Implementation

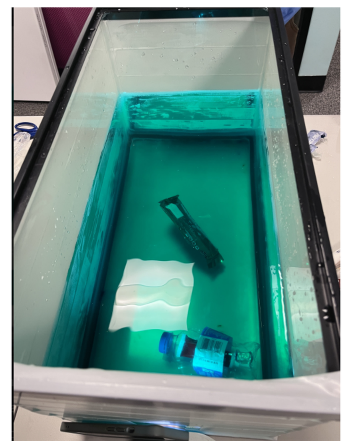
We initially trained a dataset imported from Roboflow onto our model with approximately 50 epochs worth of iterations. Utilizing our low light analog camera, we sealed it in a Tupperware container and epoxied the bottom to prevent leakage.

Setup of Experimental Environment



(a) A wider view of the simulated environment.

Inside of the Tank



(b) Top down view of the fishtank.

In our simulated marine environment, the dataset was recognizing various objects as simply plastic in the beginning; to traverse this problem we continued the GPU-intensive training so that it could consistently recognize other objects. With a total epoch count of 200, our model began to classify and detect objects that were not plastic as well.

Conclusions

Our findings indicate that YOLOv7 is satisfactory for underwater object detection. Nevertheless, it is of essence to note that the model's accuracy will depend on the dataset provided, and that the dataset must consist of multiple angles, approaches, color gradients, and other factors that would influence the likelihood of correct detection.

[1] Design and Implementation of Autonomous Underwater Vehicles' Software Stack. (2023, February 10). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/10085802

Team Lead: Aarian Ahsan, Faculty Advisor: Dr. Yonas Tadesse