Technical Report: Underwater Plastic Object Detection using YOLOv8 on Raspberry Pi 5

1. Abstract

This project focuses on the detection of underwater plastic pollution using the YOLOv8 Medium model. The model was specifically trained to identify three distinct classes of plastic debris: plastic bottles, plastic bags, and plastic masks. For real-time inference capabilities, the trained model was deployed on a Raspberry Pi 5, demonstrating its potential for embedded AI applications. However, initial observations indicated a relatively low frames per second (FPS) during operation.

2. Introduction

Underwater plastic pollution poses significant threats to marine ecosystems, leading to habitat degradation and the introduction of microplastics into the food chain, which ultimately impacts both wildlife and human health. Effective identification and monitoring of plastic waste in underwater environments are crucial for environmental protection, yet these tasks remain challenging due to the inherent complexities of underwater conditions.

In response to this challenge, this project developed a YOLOv8 object detection model tailored for recognizing plastic objects underwater. The primary focus was on common plastic items such as bottles, bags, and masks. Following training, the model was deployed on a Raspberry Pi to facilitate testing with both static images and real-time video streams, thereby showcasing the feasibility of automated underwater plastic detection using low-cost, embedded hardware.

3. Dataset

Two primary datasets were utilized for the training and validation of the YOLOv8 model:

- **Kaggle: Underwater Plastic Pollution Detection**: A publicly available dataset focusing on various forms of underwater plastic pollution.
- **Roboflow: Underwater Plastic Detection**: A comprehensive dataset providing annotated images of underwater plastic debris.

From these extensive datasets, only three specific classes of plastic objects were selected for training, based on their prevalence and environmental impact: Plastic Bottle, Plastic Bag, and Plastic Mask. A total of 630 images were used for training across these selected classes. All images were uniformly resized to a resolution of 640x640 pixels to ensure consistency during model training.

4. Model Development

The object detection model employed in this project was YOLOv8 Medium, a state-of-the-art model recognized for its balanced performance in terms of speed and accuracy in real-time object detection tasks. The training process was conducted under the following specifications:

- **Epochs**: 50
- **Batch Size**: 16
- **Image Size**: 640x640 pixels
- **Hardware**: Google Colab with T4 GPU

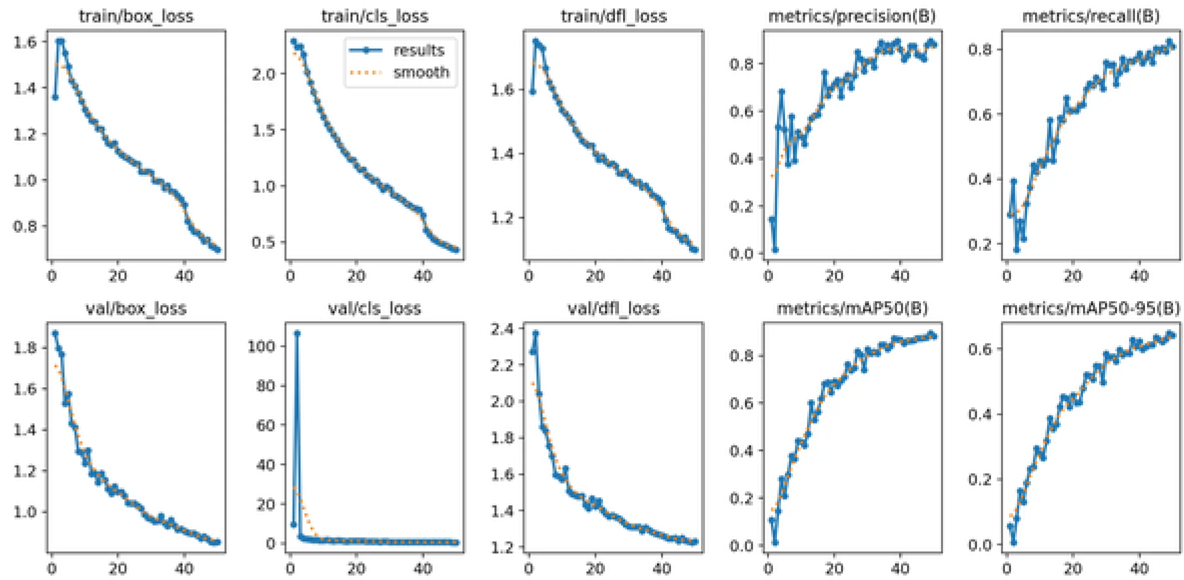
Upon completion of the training, the model achieved the following key performance metrics:

- **Precision (B)**: 0.88071
- **Recall (B)**: 0.80879
- **mAP50 (B)**: 0.88160
- **Validation Box Loss**: 0.85474
- **Validation Class Loss**: 0.58648

- **Validation DFL Loss**: 1.2304

Figure 1 illustrates the training curves, including loss, mAP, precision, and recall, over the 50 epochs of training.

Figure 1. Training metrics (loss, mAP, precision, recall) over 50 epochs.

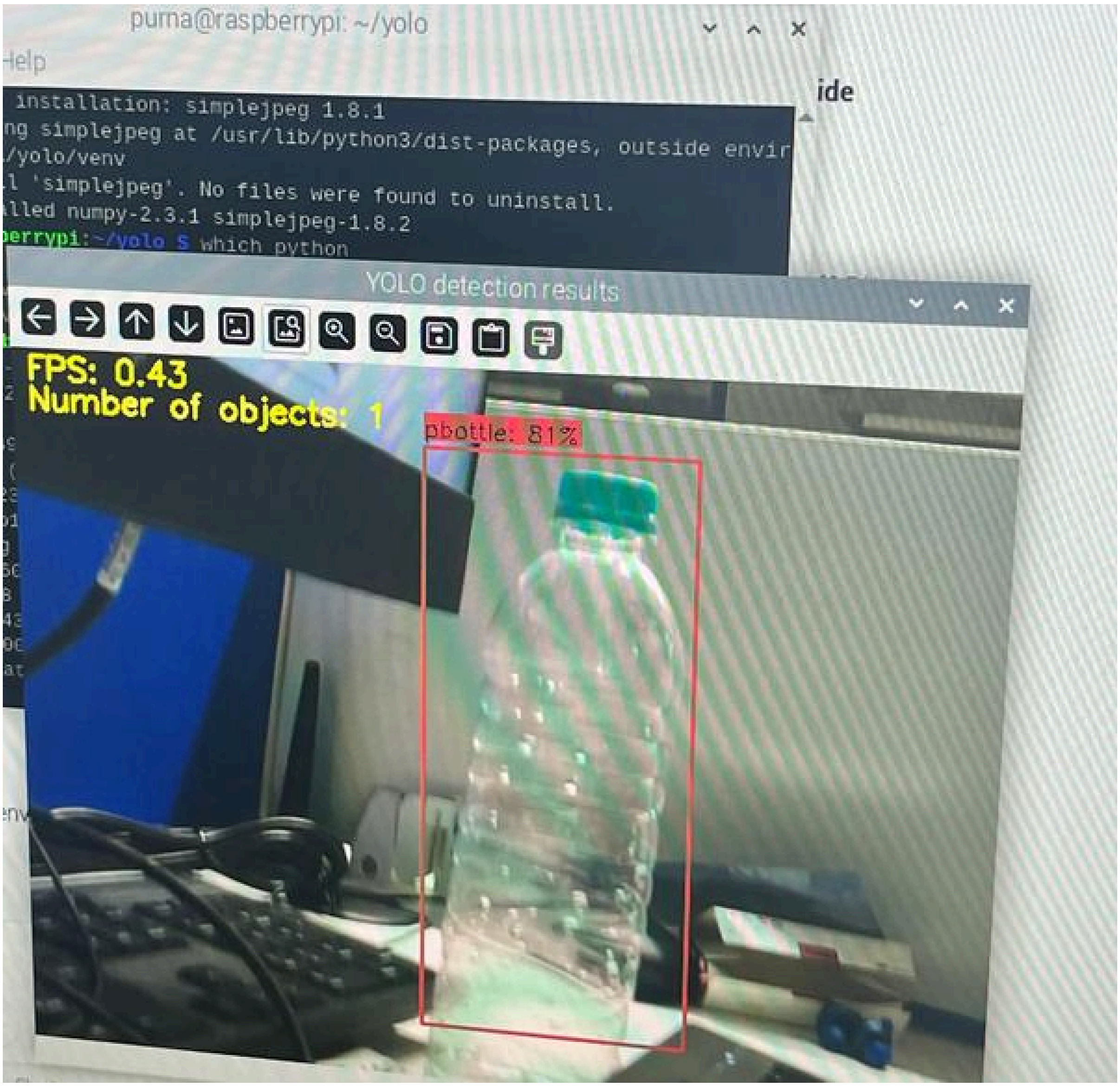


5. Deployment on Raspberry Pi 5

For real-time inference, the trained YOLOv8 Medium model was deployed on a Raspberry Pi 5 running Raspberry Pi OS. The `detect.py` script, part of the YOLOv8 framework, was utilized for performing inference on the device.

Initial testing involved using 2-3 static mobile photos, where the model successfully demonstrated correct predictions, accurately identifying the trained plastic object classes. Subsequently, a Pi camera was connected to the Raspberry Pi 5 to test real-time detection capabilities. The system proved functional, capable of performing live object detection. An example of real-time detection is shown in Figure 2. However, the observed frame rate was approximately 0.43 FPS, which, while indicative of real-time operation, is relatively low for practical applications requiring high-speed processing. The deployment utilized the standard PyTorch model, which likely contributed to the lower frame rate on the embedded system.

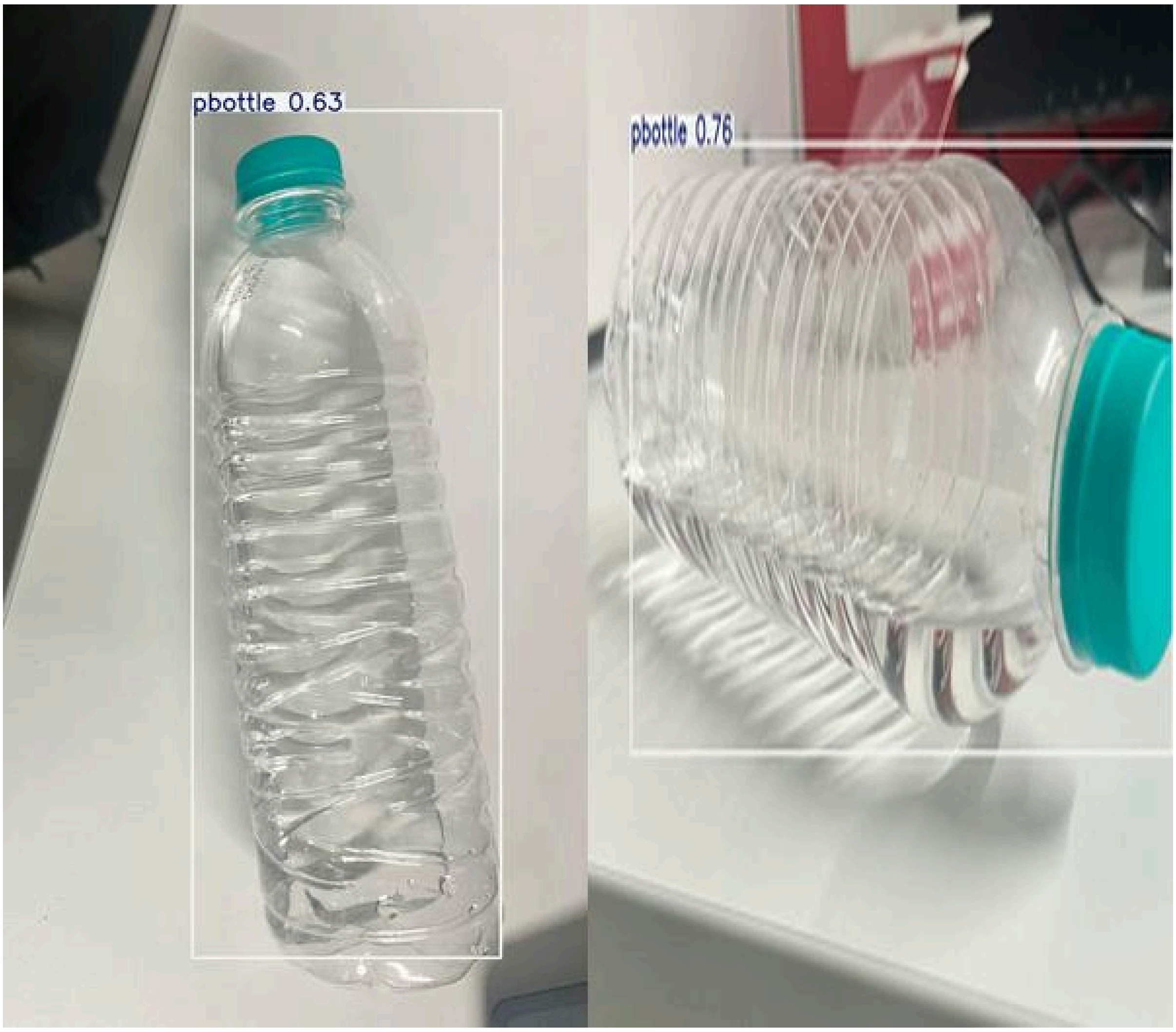
Figure 2. Real-time detection on Raspberry Pi 5.



6. Results and Discussion

The project successfully demonstrated the capability of the YOLOv8 Medium model to detect underwater plastic objects. The model performed commendably on static images captured by mobile phones, accurately identifying plastic bottles, bags, and masks. This indicates the model's robustness and generalization capabilities on pre-captured data. Examples of detections on phone-captured images are shown in Figures 3 and 4.

Figure 3 & 4. Detections on phone-captured images.



Real-time detection on the Raspberry Pi 5, while functional, exhibited a relatively low frame rate of approximately 0.43 FPS. This performance, while sufficient for proof-of-concept, may limit its practical utility in scenarios demanding high-speed processing or coverage of large areas.

7. Conclusion

This project successfully trained a YOLOv8 Medium model for the detection of three specific underwater plastic classes: plastic bottle, plastic bag, and plastic mask. Furthermore, the project achieved the crucial step of deploying this model on a Raspberry Pi 5, demonstrating the feasibility of using embedded systems for environmental monitoring. While real-time operation was achieved, the observed low frame rate highlights areas for future optimization and hardware acceleration to enhance practical applicability.