Project Report on Object Detection using Yolov8 for Aerial Maritime Dataset

Submitted by –
Soham Sandeep Mangore

Submitted to :
Aethrone Aerospace

Index

Sr. No	Title	Page No.
1	Dataset Overview	3
2	Annotation Process	3
3	Model Training	3
4	Evaluation	4
5	Object Count Analysis	6
6	Comparative Study	6

1. Dataset Overview

The dataset was taken from the given link. It consists of 3 folders test, train and valid respectively along with a yaml file. This dataset contains images of aerial maritime photographs taken via a Mavic Air 2 drone, consisting of **docks, boats, lifts, jetskis, and cars**.

Data Distribution	**Number of Images**
Train	371
Test	32
Valid	105

2. Annotation Process

The annotation of the dataset was carried out using **Roboflow**, a tool designed for creating and managing labeled datasets for computer vision tasks. The first step in the process involved manually drawing **bounding boxes** around the objects of interest in each image, which included *boat*, *car*, *dock*, *jetski*, and *lift*. These bounding boxes were placed to tightly encapsulate the objects, ensuring accuracy for object detection. Each bounding box was then labeled with the corresponding class to clearly identify the objects within the images. The annotated dataset was then exported in **YOLOv8** format, which includes the bounding box coordinates and class labels, ready for model training.

3. Model Training

For the object detection task, I selected the **YOLOv8s** (YOLOv8 Small) – a state-of-the-art deep learning model designed for real-time object detection in computer vision applications. This particular model was taken due to its high accuracy and speed.

The model was trained in **Google Colab**, leveraging its GPU resources to accelerate the training process.

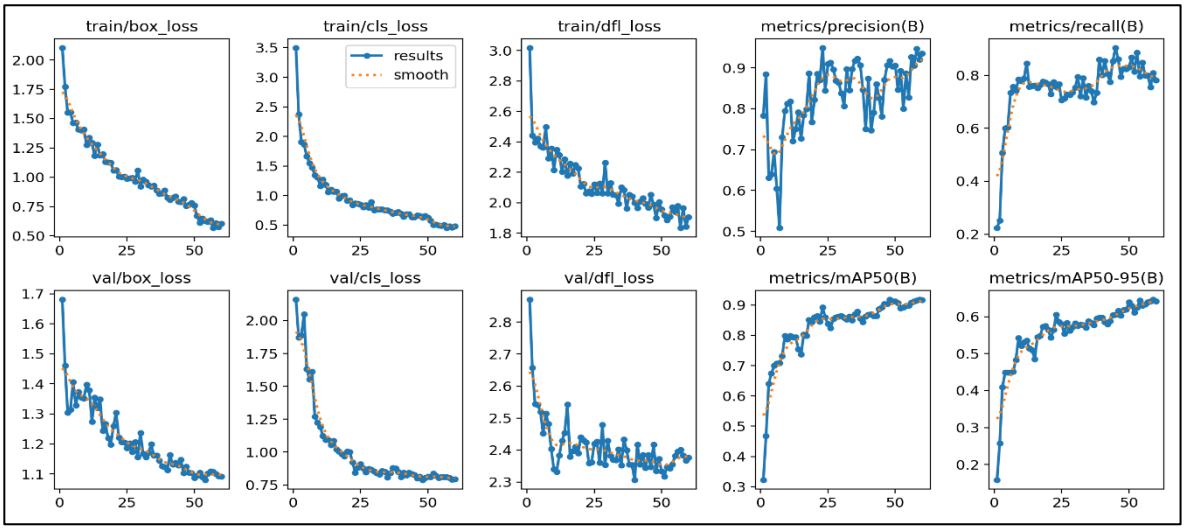
Parameters for training –

- Epochs : 60
- Imgsz (image size): 640
- Optimizer : Adams Optimizer

4. Evaluation

The model's performance was evaluated on the **validation dataset**. The evaluation process involved the use of several standard object detection metrics to assess the model's ability to detect and classify the objects accurately.

Classes	Precision	Recall (in %)	Mean Average Precision (mAP) (in %)
All	92.1	81	91.7
Boat	83.5	71.4	82
Car	100	73.1	99.5
Dock	94.9	79.2	88.7
Jetski	84.1	96.5	97.3
Lift	97.9	84.6	9.13

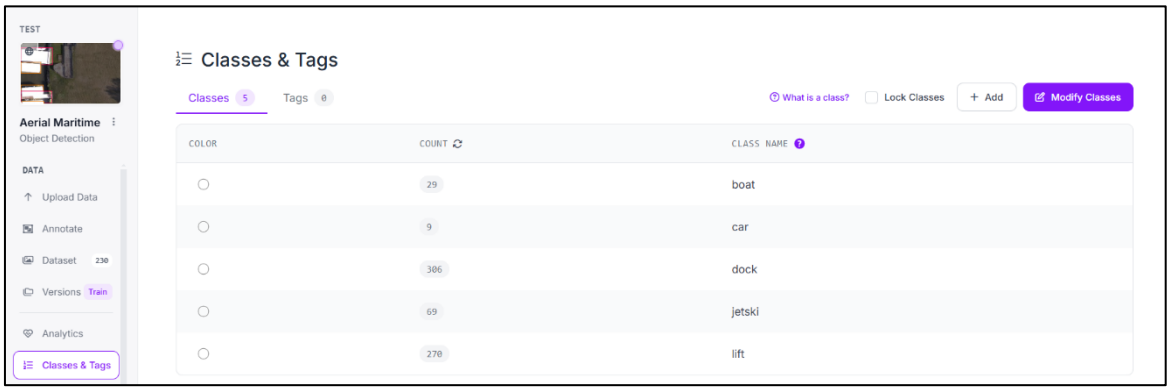


4.1 Output


lift 0.8
jetski 0.8
dock 0.9
jetski 1.0
dock 0.6
lift 0.9
dock 0.3
jetski 0.7
dock 1.0
lift 0.8
dock 0.9
jetski 0.4
car 0.9
dock 0.8
boat 0.9
lift 0.4
boat 0.9
lift 0.9
dock 0.6
dock 0.7
dock 0.5
car 0.8

5. Object Count Analysis

The object count analysis was performed using the built-in class count feature in Roboflow. This feature provided an overview of the distribution of objects across the dataset.



6. Comparative Analysis

The comparative analysis was done between two models. **Model 1**, trained on five distinct classes and **Model 2**, which combines *boat* and *jetski* into *watervehicle*.

Here is the comparative analysis between two models :

Metric	**Model 1 (5 classes)**	**Model 2 (4 classes)**
Classes	Boat, Car, Dock, Jetski, Lift	Car, Dock, Lift, Watervehicle
Box (P)	0.921	0.895
Box (R)	0.810	0.873
mAP50	0.917	0.912
Inference Speed	7.2 ms/image	5.6 ms/image
Postprocess Speed	17.4 ms/image	16.0 ms/image

**Postprocess speed* *is the time taken per image to perform operations like non-maximum suppression, formatting outputs, visualizing results, and saving detections after the model's inference.*

It is observed that both models perform similarly in terms of mAP50, with Model 2 being faster in inference and postprocessing while simplifying the class structure by merging "boat" and "jetski" into "watervehicle." Model 1, however, achieves higher precision for individual classes like "boat" and "jetski," offering more specific detections. Overall, Model 2 is optimized for speed and generalized class detection, while Model 1 focuses on detailed, class-specific predictions.