Object Detection on Military Assets using YOLOv8

Team Name: Problem Solver

Raj Kumar
Sardar Vallabhbhai National Institute of Technology, Surat

Sahil Sejal Shah
Sardar Vallabhbhai National Institute of Technology, Surat

1 Introduction

Object detection plays a critical role in modern computer vision applications such as surveillance, defense monitoring, and automated situational awareness. Accurately identifying and localizing military and civilian assets from imagery is essential for decision-making in complex and dynamic environments.

In this work, we develop a multi-class object detection pipeline using YOLOv8, a state-of-the-art one-stage detector known for its balance between accuracy and computational efficiency. The objective of this project is to train a robust object detection model capable of generalizing well to unseen test data while remaining efficient enough for real-time or resource-constrained deployment scenarios.

2 Dataset Description

The dataset provided for the hackathon consists of annotated images in standard YOLO format. Each image is associated with a text file containing normalized bounding box coordinates and class identifiers. The dataset contains twelve distinct object categories relevant to military and civilian environments:

- camouflage_soldier
- weapon
- military_tank
- military_truck
- military_vehicle
- civilian
- soldier

- civilian_vehicle
- military_artillery
- trench
- military_aircraft
- military_warship

The data is organized into training, validation, and test splits. The training split is used for learning model parameters, the validation split is used for performance monitoring and early stopping, and the test split is used exclusively for inference during submission. Ground-truth annotations for the test set are not provided.

3 Methodology

3.1 YOLOv8 Architecture

YOLOv8 is a modern one-stage object detection framework that performs bounding box regression and classification in a single forward pass. Its anchor-free design and optimized convolutional blocks improve localization accuracy and training stability while maintaining fast inference speed.

3.2 Model Selection

Among the available YOLOv8 variants, the YOLOv8-Small (YOLOv8s) model was selected for this project. This model offers a practical trade-off between detection accuracy and computational efficiency. With approximately 11 million parameters, YOLOv8s enables faster training and inference compared to larger variants, making it suitable for deployment on limited GPU or CPU-class hardware.

3.3 Training Setup

The model was initialized using COCO-pretrained weights to leverage transfer learning and improve convergence speed. Training was performed using an input resolution of 640×640 pixels and a batch size of 16. The maximum number of epochs was set to 80, with early stopping enabled using a patience value of 15 epochs to prevent overfitting.

All experiments were conducted on a Tesla P100 GPU using automatic mixed precision (AMP), which reduced memory usage and improved computational efficiency.

3.4 Data Augmentation and Robustness

To improve robustness under varying lighting conditions, object scales, and partial occlusions, standard data augmentation techniques provided by YOLOv8 were employed during training. These include mosaic augmentation, random horizontal flipping, HSV color augmentation, and scale transformations.

Such augmentations expose the model to diverse visual variations and enhance its ability to generalize to real-world scenarios, directly addressing the robustness requirements outlined in the problem statement.

4 Evaluation Metrics

Model performance was evaluated using commonly used object detection metrics, including Precision, Recall, and mean Average Precision (mAP). The primary evaluation metric considered for this project is mAP@50.

A predicted bounding box is considered a true positive if its Intersection over Union (IoU) with the corresponding ground-truth box is at least 0.5. The mAP@50 metric represents the mean of average precision values computed across all classes at this IoU threshold.

5 Results and Analysis

During training, the model exhibited stable convergence, with losses decreasing consistently over epochs. Early stopping was triggered after no further improvement was observed on the validation set, ensuring optimal generalization.

The final trained model achieved the following validation performance:

Table 1: Validation Performance Metrics

Metric	Value
Precision	0.66
Recall	0.51
mAP@50	0.54
mAP@50–95	0.36

The results demonstrate that the model successfully detects a wide range of object categories, with particularly strong performance on large and visually distinct classes such as military tanks and aircraft. Lower performance on rare or small objects is primarily attributed to class imbalance and limited sample availability.

6 Inference and Submission

After training, the best-performing model checkpoint was selected automatically based on validation performance. Inference was conducted on the test dataset using a confidence threshold of 0.25.

For each test image, a corresponding prediction file was generated in YOLO format, containing class labels, normalized bounding box coordinates, and confidence scores. All prediction files were packaged into a single ZIP archive for submission, strictly following the hackathon guidelines.

7 Conclusion and Future Work

This project demonstrates the effectiveness of YOLOv8 for multi-class object detection in military imagery. By leveraging transfer learning, data augmentation, and early stopping, the proposed approach achieves reliable detection performance while maintaining computational efficiency.

Future work may explore larger YOLOv8 variants, improved handling of class imbalance, or extension to video-based object detection for enhanced situational awareness in real-time applications.