Project Report: Vehicle Image Classification

1. Introduction

The objective of this project is to classify different vehicle types using the Vehicles-OpenImages Object Detection Dataset. It utilizes the YOLOv8 (You Only Look Once version 8) object detection framework for this task to adapt the pre-trained model to classify vehicles in the dataset.

2. Methodology

It was identified that the dataset [1] contains 627 images, each annotated with object labels for different vehicle classes. The specified classes include cars, buses, motorcycles, ambulances, and trucks. Understanding the dataset's use cases provided valuable insights into the expected diversity of vehicle instances and the challenges associated with the task. All images in the dataset have been resized and auto-oriented. This preprocessing step ensures uniformity in image dimensions and corrects any orientation variations that might exist in the original images.

The YOLOv8 model by Ultralytics [2] was chosen for its efficiency in object detection tasks. The decision to opt for YOLOv8 was based on several compelling reasons, including:

- Efficiency: The model maximizes hardware usage, whether on a single-GPU setup or scaling across multiple GPUs.
- Versatility: YOLOv8 supports training on custom datasets, allowing flexibility beyond commonly used datasets like COCO, VOC, and ImageNet.
- User-Friendly: The model provides a simple yet powerful Command Line Interface (CLI) and Python interfaces for a straightforward training experience.
- Hyperparameter Flexibility: YOLOv8 offers a broad range of customizable hyperparameters, allowing fine-tuning of model performance to suit specific requirements.

The training process involved setting hyperparameters, specifying the number of epochs, and addressing challenges encountered during the training phase. The model was trained for 25 epochs, considering the relatively small size of the dataset.

3. Result

The precision-confidence curve shown below (Fig. 1) is a graphical representation of how precision changes as a function of the confidence level in the classification model. Precision is a measure of the accuracy of the positive predictions made by the model. It is the ratio of true positives to the sum of true positives and false positives. This curve helps visualize how the precision of a model varies at different classification thresholds.

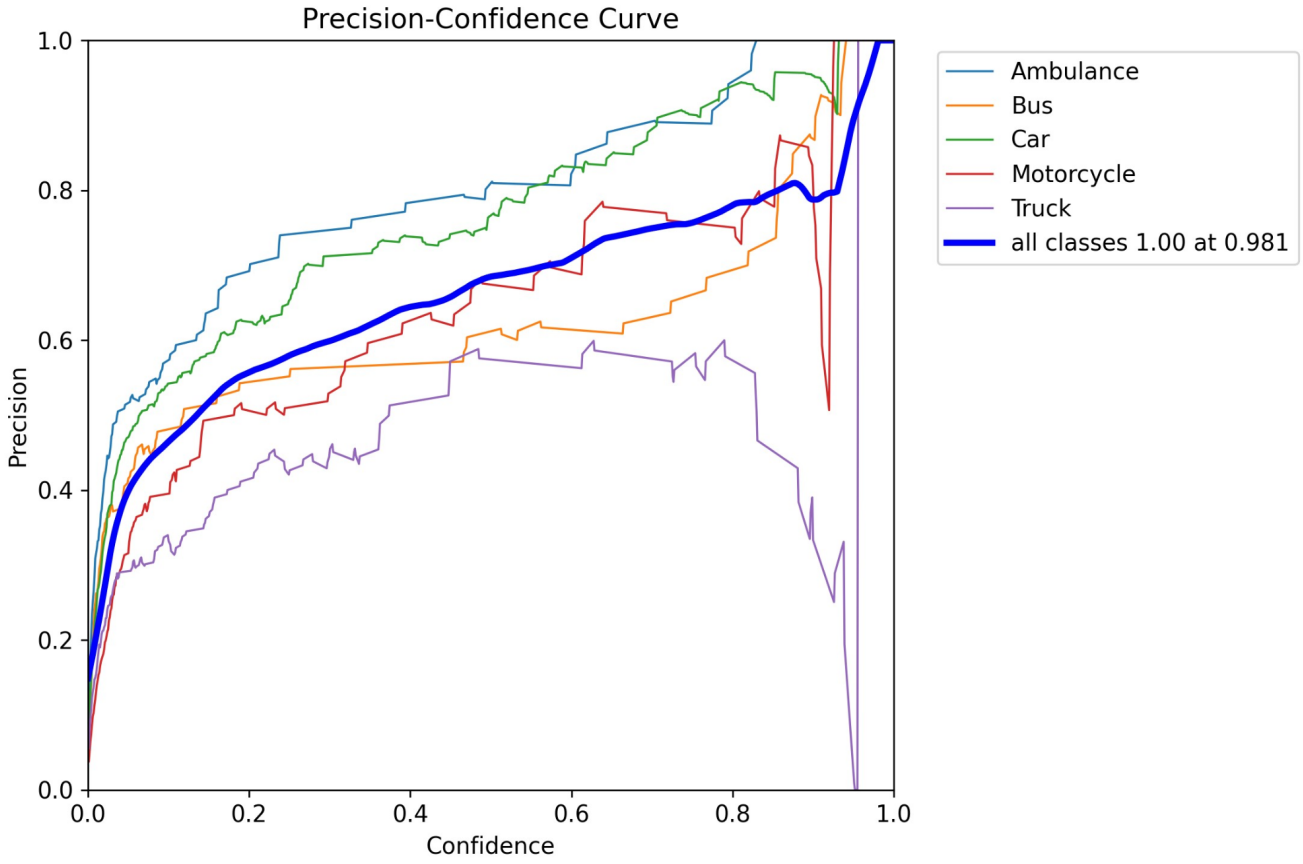

Fig 1. Precision curve

At a confidence threshold of 0.981, the model is making predictions that are all correct, showing perfect precision across all classes.

The confusion matrix shows the prediction of the model in various classes vs the true class (Fig. 2). It is observed that the model predicts the car as the background almost 43% of the time. This could be improved by increasing the number of epochs and reducing the learning rate.

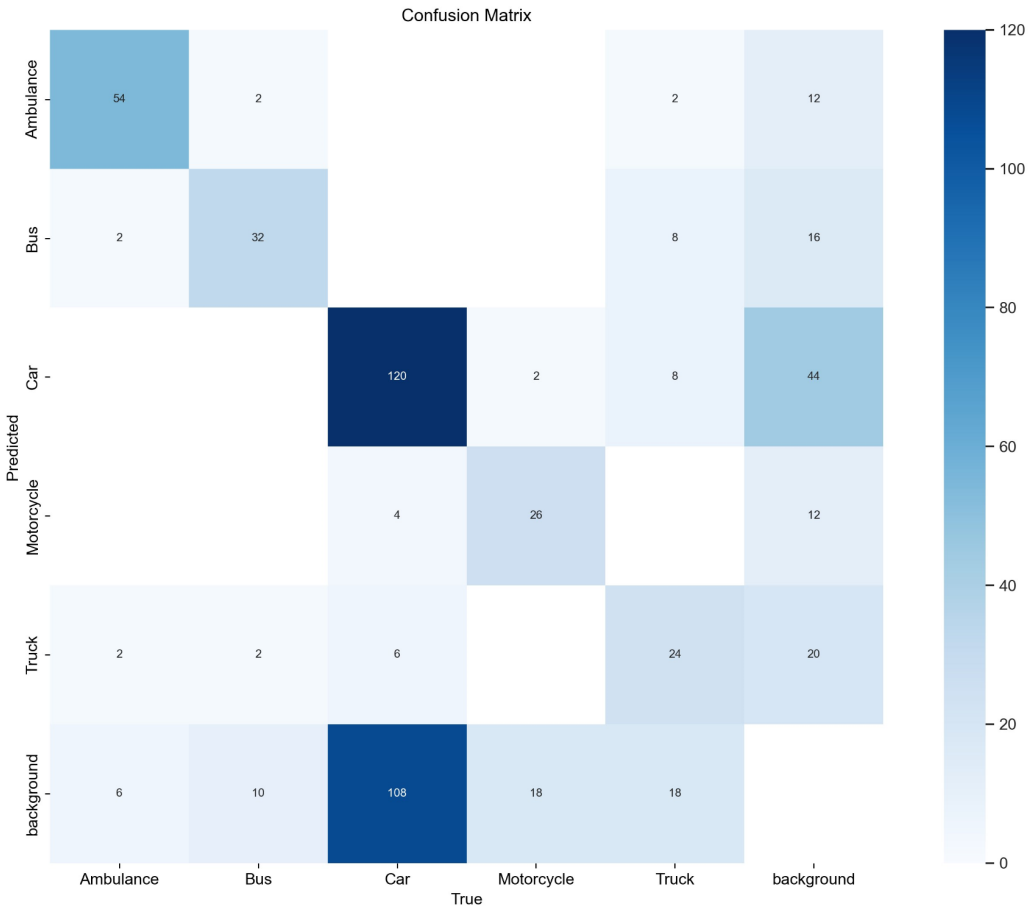

Fig 2. Confusion matrix

4. Comparative Analysis

In a previous attempt, a ResNet prebuilt model achieved a testing accuracy of 62%, without incorporating the YOLOv8 model. A critical distinction is that YOLOv8 excels in drawing bounding boxes around detected vehicles, providing not only classification but also precise localization information. This feature is absent in the ResNet pre built model. The preference for YOLOv8 is driven by its efficiency, versatility, user-friendly interfaces, and its ability to perform object localization.

5. Conclusion

In conclusion, this project successfully employed the YOLOv8 model to classify diverse vehicle types using the Vehicles-OpenImages dataset. The precision-confidence curve highlighted the model's strong precision performance at a confidence threshold of 0.981, indicating robust predictions. However, a specific challenge emerged: misclassifying cars as the background in 43% of cases. Future enhancements involve optimizing training parameters to address this misclassification pattern. Overall, the project lays a solid foundation for further refinement, emphasizing the importance of iterative improvements in real-world object detection scenarios.

References

[1] Dataset Version Preprocessing > resize (roboflow.com)
[2] Home - Ultralytics YOLOv8 Docs