COMP1002-Advanced Python Course
Final Project Report

Project Groups

Yasin Şamil Memnun : 232010020126
Mert Altundal : 232010020039
Cihangir Kanat : 232010020011
Toygun Galyan : 232010020130

Github Link: Github link: https://github.com/toy41/Smart-Vehicle-Detection-and-Counting-System

1. Introduction

This project focuses on vehicle detection and counting using computer vision techniques. The main goal is to detect and track cars in a video stream, count each unique vehicle, and generate a report of the results. The system is built using Python and leverages state-of-the-art object detection (YOLOv8) and tracking (SORT) methods to achieve real-time performance. This project was developed for the Advanced Python course to demonstrate the practical application of OOP principles.

2. Technologies Used

- Python 3.8+
- OpenCV: For video processing and visualization
- YOLOv8 (Ultralytics): For object detection
- SORT: Simple Online and Realtime Tracking algorithm
- ReportLab: For PDF report generation
- CSV: For logging detection results

3. Implementation Details

The application starts by loading a pretrained YOLOv8 model and opening the input video file. Each video frame is passed through the model to detect cars. The detected cars are filtered based on class labels and confidence scores. Detected bounding boxes are sent to the SORT tracker, which assigns unique IDs to each car. These tracked IDs are used to count only distinct vehicles once. Each detection is logged into a CSV file with frame number, class label, count, and track ID. At the end of the video, a PDF report is automatically generated based on the CSV output.

4. Challenges & Solutions

One challenge was ensuring that only cars were counted and that the same vehicle was not counted multiple times. This was addressed by filtering detections by class and maintaining a set of unique track IDs. Another issue was the performance bottleneck in real-time detection and display. This was mitigated by using YOLOv8n (a smaller and faster version of YOLO) and optimizing OpenCV display logic. Error handling was also implemented for missing files and video issues.

5. Conclusion & Future Work

This project successfully demonstrates how modern computer vision and tracking techniques can be integrated into a functional system for vehicle detection and counting. The use of YOLOv8 and SORT allowed the system to work efficiently with good accuracy. In the future, improvements can be made by including additional vehicle types, integrating license plate recognition, or deploying the system in a web or desktop interface with live analytics and alerting capabilities.