Deep Learning

Institute of Business Administration

Department of Mathematics and Computer Science

Speed Estimation using DL and Vision techniques

Authors:
Student 1 (ID: Hafiz Muhammad Owais Raza(29637))
Student 2 (ID: Farjad Ahmed(29405))

Date: January 4, 2025

Abstract

This report presents a comprehensive analysis of a project aimed at estimating the speed of objects in video sequences by leveraging the YOLOv8 object detection model and the Supervision library. The project focuses on perspective analysis using a single video (*vehicles.mp4*) with available metadata. The integration of these advanced tools facilitates detection, tracking, and speed estimation using a calibrated perspective-based approach. The report delves into the project's objectives, methodologies, results, and potential applications, providing a detailed understanding of its contributions to the field of computer vision.

1 Introduction

Accurate speed estimation of moving objects in videos is pivotal for applications such as traffic monitoring, autonomous vehicle navigation, and sports analytics. Traditional methods often struggle with challenges like occlusions, varying lighting conditions, and complex backgrounds. The advent of deep learning models, particularly the YOLO (You Only Look Once) series, has revolutionized object detection by enabling real-time performance with high accuracy. The latest iteration, YOLOv8, offers enhanced capabilities in terms of speed and precision. Complementing this is the Supervision library, a Python package designed to streamline computer vision tasks, including object tracking and video analysis. This project integrates YOLOv8 and Supervision to develop a robust system for estimating object speeds in a calibrated perspective-based analysis using the video *vehicles.mp4*.

2 Objectives

The primary objectives of this project are:

- **Object Detection**: Utilize YOLOv8 to accurately detect and classify objects in each frame of the video *vehicles.mp4*.
- **Object Tracking**: Implement a reliable tracking mechanism to maintain object identities across consecutive frames.
- **Speed Estimation**: Calculate the speed of detected objects by analyzing their movement using a perspective analysis approach.
- **Real-Time Analysis**: Demonstrate the system's capability to operate effectively on a single video.

3 Methodology

The project workflow comprises several key components:

3.1 Data Acquisition

The video *vehicles.mp4*, sourced from Roboflow, was selected due to its available metadata and suitability for perspective-based analysis. The video features moving vehicles in a traffic scenario, providing a controlled environment for speed estimation.

3.2 Object Detection with YOLOv8

YOLOv8, renowned for its real-time object detection capabilities, was employed to identify vehicles within each video frame. The model was used with minimal fine-tuning, relying on its pre-trained weights for vehicle detection.

3.3 Perspective Calibration and Analysis

A perspective calibration approach was applied using metadata from the video. A known reference distance in the scene was used to establish a scale for converting pixel displacements into real-world distances. This step is critical for accurate speed estimation.

3.4 Speed Estimation

The speed of each vehicle was estimated by calculating the displacement between consecutive frames and converting pixel movement into real-world measurements using the calibrated perspective scale. The formula $\text{Speed} = \frac{\text{Displacement}}{\text{Time}}$ was applied, where displacement is measured in real-world units.

3.5 Implementation Details

The system was implemented in Python, leveraging the following libraries:

- **YOLOv8**: For object detection.
- **Supervision**: For object tracking and video processing utilities.
- **OpenCV**: For video frame manipulation and perspective analysis.
- **NumPy**: For numerical computations.

4 Results

The system was evaluated on the single video *vehicles.mp4*, and its performance was assessed based on detection accuracy, tracking reliability, and speed estimation precision.

4.1 Detection Accuracy

YOLOv8 demonstrated high accuracy in detecting vehicles within the video. The model effectively identified and classified objects, maintaining robustness against challenges like occlusions and varying lighting conditions.

4.2 Tracking Reliability

The integration of the Supervision library ensured reliable tracking of vehicles across frames. The system maintained consistent object identities, which is critical for accurate speed estimation.

4.3 Speed Estimation Precision

The speed estimation component achieved a high degree of precision. The calibration process ensured that pixel movements were accurately translated into real-world distances. Table 1 summarizes the results for a few sample vehicles.

Table 1: Speed Estimation Results (Sample Vehicles)

Vehicle ID	Displacement (m)	Time (s)	Speed (km/h)
1	40.0	1.2	120.0
2	39.6	1.2	118.8
3	22.2	1.0	80.0

4.4 Comparison with Background Subtraction Approach

An alternate approach using background subtraction and centroid-based tracking was implemented for benchmarking. The comparison has been divided into two tables to address layout issues.
The YOLOv8-based approach offers higher accuracy and robustness, particularly in dynamic scenarios with occlusions or varying lighting conditions. In contrast, the background subtraction method is computationally lightweight and suitable for static camera setups but struggles in complex environments.

Table 2: Comparison of YOLOv8-Based and Background Subtraction

Aspect	YOLOv8-Based Approach	Background Subtraction
Detection Method	Deep Learning (YOLOv8)	Traditional (Background Subtraction)
Accuracy	High	Medium (dependent on lighting and motion)
Performance	Relatively slower, GPU recommended	Faster, works well on CPU
Scalability	Suitable for diverse scenarios	Limited to static backgrounds
Setup Complexity	Requires pre-trained model weights	Simple setup using OpenCV
Real-Time Processing	Achieves real-time with optimization	Achieves real-time efficiently
Sensitivity to Environmental Changes	Robust	Sensitive to lighting and shadows

5 Discussion

The project's focus on a single video with perspective-based calibration highlights the practicality of using YOLOv8 and Supervision for targeted speed estimation tasks. The results demonstrate the system's accuracy and reliability in controlled scenarios. However, the reliance on metadata and calibration limits its scalability to other videos without similar metadata.

6 Conclusion

This project successfully demonstrates the integration of YOLOv8 and the Supervision library to develop a system for estimating the speed of vehicles in a single video using perspective analysis. The approach leverages metadata for calibration, ensuring high precision in speed estimation. Future work could explore automating the calibration process and extending the methodology to handle multiple videos or diverse scenarios.

7 Future Work

While the system performs effectively on a single video, there are avenues for further enhancement:

- **Automated Calibration**: Develop methods to automate the perspective calibration process.
- **Scalability**: Extend the system to handle videos without predefined metadata.
- **Environmental Adaptability**: Address challenges posed by varying lighting conditions and weather scenarios.