World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(Review Article)

# Speed sense: Smart traffic analysis with deep learning and machine learning

Parwateeswar Gollapalli, Neha Muthyala, Prashanth Godugu, Nikitha Didikadi and Pavan Kumar Ankem *

*Department of CSE (Data Science), ACE Engineering College, Hyderabad, Telangana, India.*

World Journal of Advanced Research and Reviews, 2024, 21(03), 2240–2247

Publication history: Received on 12 February 2024; revised on 23 March 2024; accepted on 25 March 2024

Article DOI: https://doi.org/10.30574/wjarr.2024.21.3.0896

## Abstract

The main reason for many road accidents in modern times is speeding and negligent driving. This project aims to identify vehicles that exceed the speed limit and employs a machine learning algorithm for this purpose. It eliminates the need for manual checks by the police to identify speeding vehicles. The project involves vehicle detection and tracking as key steps, enabling us to classify the type of vehicle, their respective speeds, and the count of vehicles passing through given region. Counting the number of vehicles helps manage traffic, allowing us to identify peak traffic times and take necessary precautions to avoid long traffic jams. Vehicle tracking is the process of detecting a moving vehicle using a camera. Capture vehicle in video sequence from surveillance camera is demanding application to improve tracking performance. This technology is increasing the number of applications such as traffic control, traffic monitoring, traffic flow etc. Video and image processing are vital for traffic surveillance, analyzing, and monitoring in urban areas. Recent speed estimation methods prioritize accuracy and cost-effective hardware implementation.

**Keywords:** Vehicle Detection; Classify the type of vehicle; Count of Vehicle; Traffic Jams; Video and image processing.

## 1. Introduction

Nowadays, with the continuous increase of vehicle in the road, traffic management authority requires better traffic surveillance system. As the volume of vehicles increases annually, so does the frequency of road accidents. Speed is now the single cause of road accidents. Enforcing speed limit is one of the ways to eliminate speed related accidents. Traffic surveillance systems are pivotal for enforcing speed limits through vehicle detection and speed measurement. They play a crucial role in providing relevant information for the traffic control such as vehicle speed, traffic count etc.

Those systems are divided in intrusive and non-intrusive sensors. Intrusive sensors are mostly based on inductive loop detectors. Despite their widespread usage, these sensors pose challenges with complex installation, high maintenance requirements, susceptibility to asphalt deterioration, and vulnerability to wear and tear damage. Intrusive sensors require the line sight connection between vehicle and the equipment.

Additionally, the high cost of equipment and lower accuracy have contributed to the declining popularity of these systems. Non-intrusive sensors contain doppler radars and laser meters, which avoid intrusive sensors problems, but these require frequent maintenance and more expensive. The work presented in this project aims to overcome this challenging task of speed determination by providing an economic solution. Because of the availability of cheaper cameras which produces images with higher quality, video-based systems are by far the most inexpensive alternate for non-intrusive speed measurement. Indeed, current systems frequently integrate with video cameras to capture license plates of speeding vehicles, leveraging pre-existing infrastructure in many instances. Utilizing image processing, these systems efficiently analyze videos and extract pertinent information

* Corresponding author: A. Pavan Kumar

## 2. Literature Review

Image segmentation involves dividing a digital image into segments (super pixels) to simplify its representation for analysis. Developing accurate segmentation algorithms is challenging, as there's no universal method for diverse images. This work proposes robust histogram thresholding to aid segmentation, highlighting semi-atomic algorithms. Various histogram thresholding methods are illustrated, emphasizing their application in image segmentation. Keywords include Image Segmentation, Histogram Thresholding, Methods, and Thresholding Foundation [1].

This paper introduces an Intelligent Transportation System (ITS) using Java and OpenCV for real-time traffic data collection in smart cities. The adaptive video-based tool detects, classifies, counts, and measures vehicle speed, aiding traffic flow monitoring and network management. Employing MOG2 background subtraction, it achieves over 80% accuracy in classifying various vehicles in different conditions in Dhaka city. The system extracts data in csv/xml format, contributing to intelligent transport management by supporting route information, safety, and electronic payment systems [2].

Addressing the escalating traffic challenges due to population growth, this study employs YOLO and OpenCV for real-time vehicle detection, classification, and counting. Essential for accident prevention, the approach enhances accuracy and efficiency using convolutional neural networks and machine learning. By processing input video through OpenCV, the model identifies and categorizes vehicles, ensuring quicker and more accurate results compared to other algorithms. This technology not only bolsters driving safety and optimizes traffic flow but also paves the way for advancements in autonomous driving [3].

This paper emphasizes the growing importance of accurately estimating vehicle speed, driven by increased speed camera installations and the role of traffic monitoring in smart cities. Vision-based systems, while posing challenges, offer cost advantages and precise vehicle identification. The review explores terminology, application domains, and a comprehensive taxonomy of works related to vision-based vehicle speed estimation. Additionally, it covers performance evaluation metrics and available datasets, contributing to the understanding and development of this critical aspect for road safety and traffic optimization [4].

This survey addresses the foundation of computer vision research, focusing on moving object detection. Unlike previous surveys emphasizing detection accuracy, this one evaluates methods from a practical perspective, considering diverse application tasks. It distinguishes between the detection of seen and unseen scenes, reviewing recent technologies for each. The survey provides valuable insights for researchers and technicians, defining tasks, reviewing relevant detection methods, and suggesting future directions for choosing suitable algorithms in practical applications [5].

## 3. Existing System

The current traffic monitoring systems mainly depend on two type-s of devices to measure- speed: intrusive and non-intrusive devices. Intrusive Devices: These appliance-s are generally based on induction loops built into the road. They require roadwork for installation. They work by detecting changes in the magnetic field when a vehicle passes over the loop. While they have been widely used, intrusive sensors suffer from several limitations. They require complex installation procedures, leading to higher setup costs. Additionally, they demand regular maintenance, which can be time-consuming and expensive.

Furthermore, they promote asphalt deterioration over time, affecting the road's overall lifespan. Furthermore, direct contact sensors need a clear view betwe-en the vehicle and the equipment, making them less adaptable for different road designs. Non-contact Sensors: This group involves technologies for example laser detectors and doppler radars. Non-contact sensors provide some benefits over direct contact versions. They don't need road surface installations, reducing the risk of asphalt harm. However, non-intrusive sensors tend to be more expensive, and they often demand frequent maintenance to ensure accuracy and reliability. This higher cost and maintenance burden have limited their widespread adoption, making them less favorable for budget-conscious traffic management authorities.

## 4. Proposed System

The setup includes a camera that takes pictures of traffic. A computer looks at the pictures and uses a detection model called YOLO v8 to find vehicles. YOLO v8 was trained on many vehicle pictures so it can spot, name, count, and estimate the speed of cars and trucks. This system only needs one- camera to film traffic. That makes it cheaper than using lots of cameras. It is made to quickly understand the live video. This allows it to work in real-time for real applications.

### 4.1. SCOPE

The goal of this project is to create, build, and put in place an advanced and affordable video-based traffic monitoring system with picture handling capacities for speed finding and traffic control. The venture expects to defeat the disadvantages of current disturbing and non-obtrusive sensor frameworks by giving a moderate and dependable arrangement for authorizing speed limits and improving traffic administration.

Picture Handling Proce-dures: The undertaking will include, the usage of progresse-d picture handling calculations to break down the video footage and concentrate pertinent data, for example, vehicle speed and other traffic parameters.

### 4.2. System Architecture

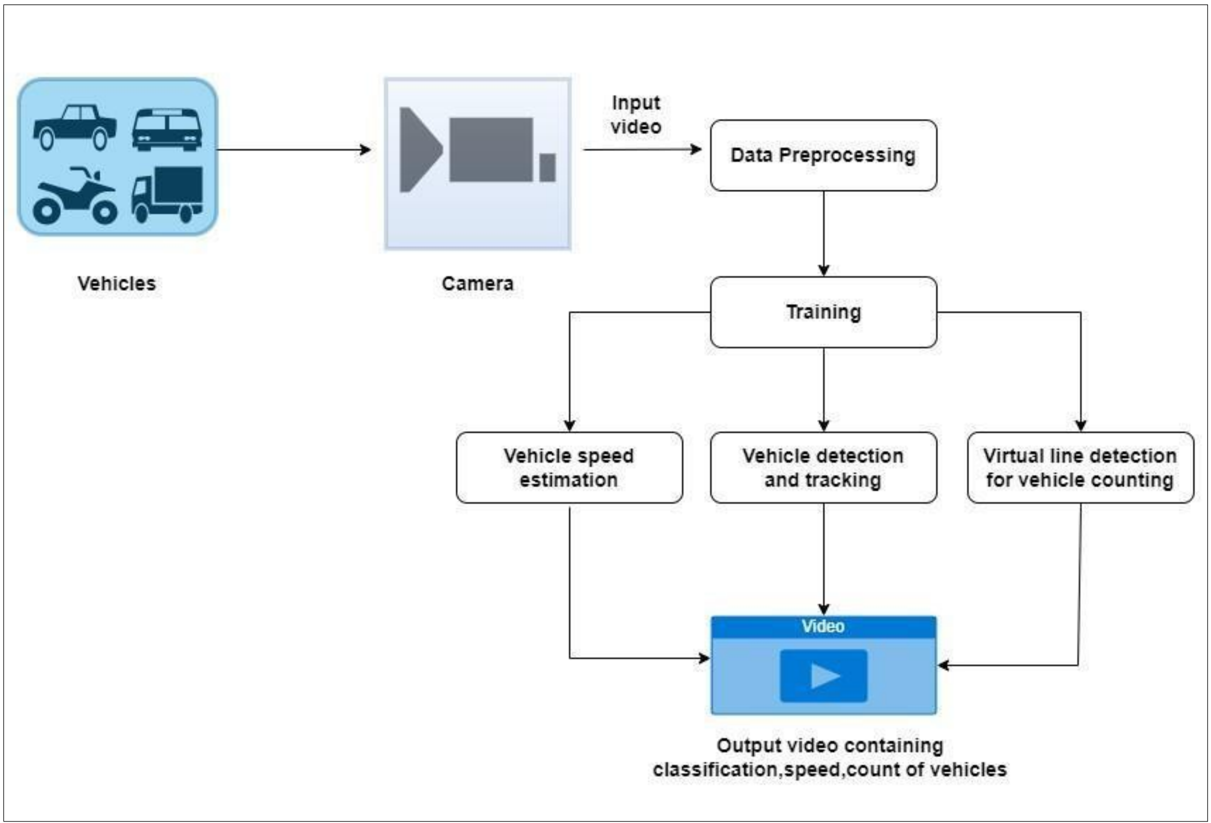


**Figure 1** System Architecture

### 4.3. Architecture Description

- Video Input: High-quality video footage captured by cameras placed strategically on the road.
- Object Detection: Advanced algorithms to detect vehicles and license plates in the video.
- Speed Calculation: Using image timestamps and vehicle positions to calculate speed accurately.
- Data Storage: Storing processed data, including user name and password for authentication.
- Scalability: Designing the architecture to support the addition of more cameras and traffic Areas.

## 5. Exploratory Data Analysis

Exploratory Data Analysis (EDA) for vehicle speed estimation and counting using YOLOv8 and Deep SORT involves gaining insights into the datasets, understanding the distribution of key variables, and identifying patterns or challenges in the data. Here's an outline for conducting EDA

### 5.1. Data Loading and Inspection

Load the annotated datasets for YOLOv8 and Deep SORT. Inspect the structure of the datasets, including image or video files, annotations, and tracking information. Check for missing or incomplete data.

### 5.2. Visualization of YOLOv8 Detection

Visualize a subset of images or video frames with YOLOv8 detections. Examine the quality of object detection, ensuring that vehicles are accurately localized with bounding boxes.

### 5.3. Tracking Visualization with Deep SORT

Visualize the tracking results obtained from the Deep SORT algorithm. Track a few vehicles across consecutive frames to ensure smooth and accurate tracking. Identify challenging scenarios, such as occlusions or abrupt changes in vehicle movement.

### 5.4. Speed Estimation Analysis

Analyze the distribution of vehicle speeds in the speed estimation dataset. To comprehend the range and variety of vehicle speeds, create visualizations such kernel density charts or histograms. Also find any anomalies or outliers in the speed data.

### 5.5. Vehicle Counting Analysis

Analyze the distribution of vehicle counts in the counting dataset. Visualize the temporal patterns of vehicle counts over time. Explore variations in vehicle counts based on different regions or time intervals.

### 5.6. Correlation Analysis

Examine the relationship between car speed and other factors, including traffic volume or the time of day. Analyze the impact that changes in one variable (like speed) have on another (like vehicle count).

### 5.7. Performance Metrics Evaluation

Analyze the YOLOv8 and Deep SORT algorithms' performance metrics. Determine object detection parameters like recall, precision, and F1 score. Assess tracking metrics, including tracking accuracy and ID switches, for Deep SORT.

### 5.8. Identify Challenges and Limitations

Identify any challenges or limitations observed during the EDA. Consider factors that may affect the accuracy of speed estimation and vehicle counting, such as environmental conditions or camera perspectives.

### 5.9. Interactive Visualization

Create interactive visualizations or dashboards to allow users to explore the data dynamically. Include features like sliders or filters to focus on specific subsets of the data.

### 5.10. Documentation

Document key findings, insights, and potential areas for improvement. Provide a brief summary of the benefits and drawbacks of using Deep SORT and YOLOv8 together for counting and estimating vehicle speed. You may evaluate the effectiveness of your algorithms, gain a deeper understanding of the properties of your datasets, and make well-informed judgements regarding the improvement and optimization of your car monitoring system by conducting thorough exploratory data analysis.

## 6. Model Development

### 6.1. ALGORITHM

#### *6.1.1. Deep SORT*

Deep SORT (Simple Online and Realtime Tracking) stands as a sophisticated algorithm employed for the multi-object tracking within videos or sequences of images. It builds upon the concepts of object detection and association to track objects accurately and robustly over time. The key steps involved in the Deep SORT algorithm are as follows.

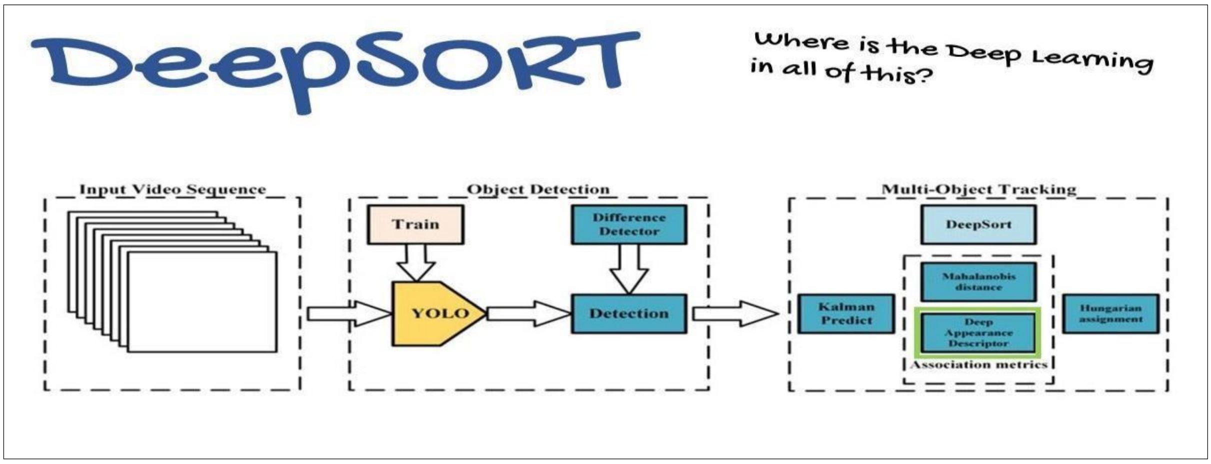


**Figure 2** Deep Sort

**Step 1:** Detection: Every frame is examined for items using an object detection model (YOLO, SSD, etc.). The detector returns the class probabilities and bounding box coordinates for each object it finds.

**Step 2:** Feature Extraction: Rich feature representations of the observed objects are extracted from them using a deep neural network by Deep SORT. These features encode distinctive characteristics of each object, enabling efficient matching in subsequent frames.

**Step 3:** Data Association: The main challenge in multi-object tracking is associating the detected objects across frames correctly. Deep SORT utilizes the Kalman Filter and Hungarian algorithm to associate the detected objects in the current frame with the previously tracked objects, taking into account motion dynamics and feature similarity.

**Step 4:** Track Management: Deep SORT maintains and manages tracks for each object by updating their state estimates and handling occlusions or temporary disappearance. It also handles track termination when an object leaves the scene or becomes untraceable.

**Step 5:** Output: The Deep SORT algorithm's final result is a set of consistently monitored item trajectories across time. Numerous applications, including activity analysis, driverless vehicles, and monitoring, benefit from this data.

*6.1.2. YOLOv8*

You Only Look Once, or YOLO, is a well-liked real-time object detection approach that is effective and accurate. The most recent version, known as YOLOv8, enhances the original YOLO concept's utility even more.

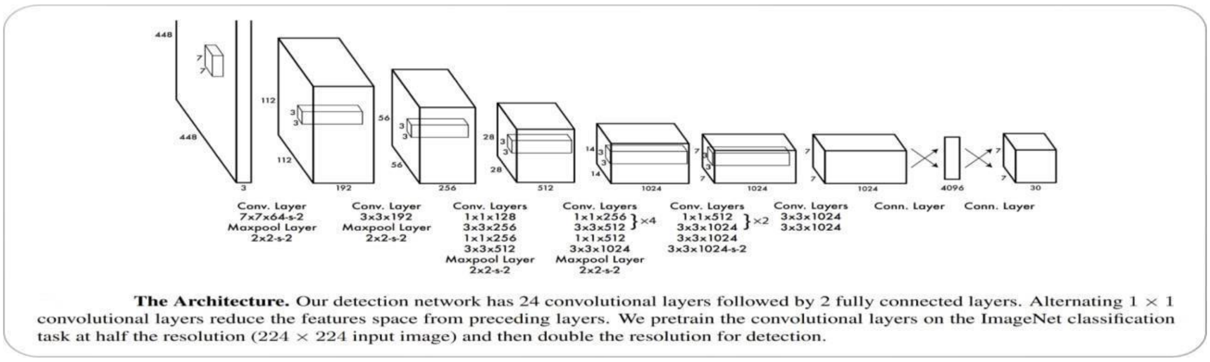


**Figure 3** YOLOv8

**Step 1:** Backbone Network: YOLOv8 employs a powerful backbone network, often based on Darknet or CSP Darknet, to extract high-level features from the input image efficiently.

**Step 2:** Pyramid of Features To collect object properties at several scales, the algorithm creates a feature pyramid, which enables YOLOv8 to accurately detect objects of varying sizes.

**Step 3:** Forecasting The input image is divided into a grid of cells using YOLOv8, which then forecasts bounding boxes and class probabilities for each cell.

**Step 4:** Loss Function: To train the model effectively, YOLOv8 uses a combination of localization loss (to improve bounding box accuracy) and classification loss (to enhance class prediction accuracy).

**Step 5:** Efficient Inference: YOLOv8 is designed to be computationally efficient, enabling real-time object detection on various platforms, including CPUs and GPUs.

**Step 6:** Improved Performance: YOLOv8 introduces architectural improvements and finetuning techniques that enhance object detection performance, making it one of the state-of-the art algorithms in the field.

In summary, YOLOv8 is an advanced object detection algorithm known for its real-time capabilities and remarkable accuracy. When combined with Deep SORT for multi-object tracking, it provides a powerful solution for real-world applications requiring robust and efficient object detection and tracking in video streams.

## 7. Module description

- Pandas
- NumPy

### 7.1. Pandas

Pandas offers a rich array of Series and Data Frames, empowering seamless organization, exploration, representation, and manipulation of data. Its intelligent alignment and indexing capabilities ensure precise organization and labeling of data. Furthermore, Pandas includes specialized features for effectively managing missing data or values, providing comprehensive data handling solutions. The above package offers a clean code that even people with no or basic knowledge of programming can easily understand and work with it.

Pandas offers a comprehensive suite of built-in tools for reading and writing data across various web services, data structures, and databases. Supporting a wide array of formats including JSON, Excel, CSV, HDF5, and more, Pandas facilitates versatile data management. Furthermore, Pandas enables seamless merging of multiple databases simultaneously, enhancing data integration capabilities.

### 7.2. NumPy

NumPy arrays provide advanced mathematical functionalities for handling vast amounts of data efficiently. By simplifying project execution, NumPy streamlines complex tasks with ease. Offering masked arrays alongside general array objects, NumPy enhances flexibility in data manipulation. Additionally, it boasts a wide range of features including logical shape manipulation, discrete Fourier transform, general linear algebra operations, and numerous other capabilities, contributing to its indispensability in scientific computing. Whenever you change the shape of any N-dimensional arrays, NumPy will create new arrays for that and delete the old ones.

This python package provides useful tools for integration. It is easy to integrate NumPy with C, C++ programming languages.

NumPy offers functionalities that rival those found in MATLAB. These both allow users to get faster with operations.

## 8. Results

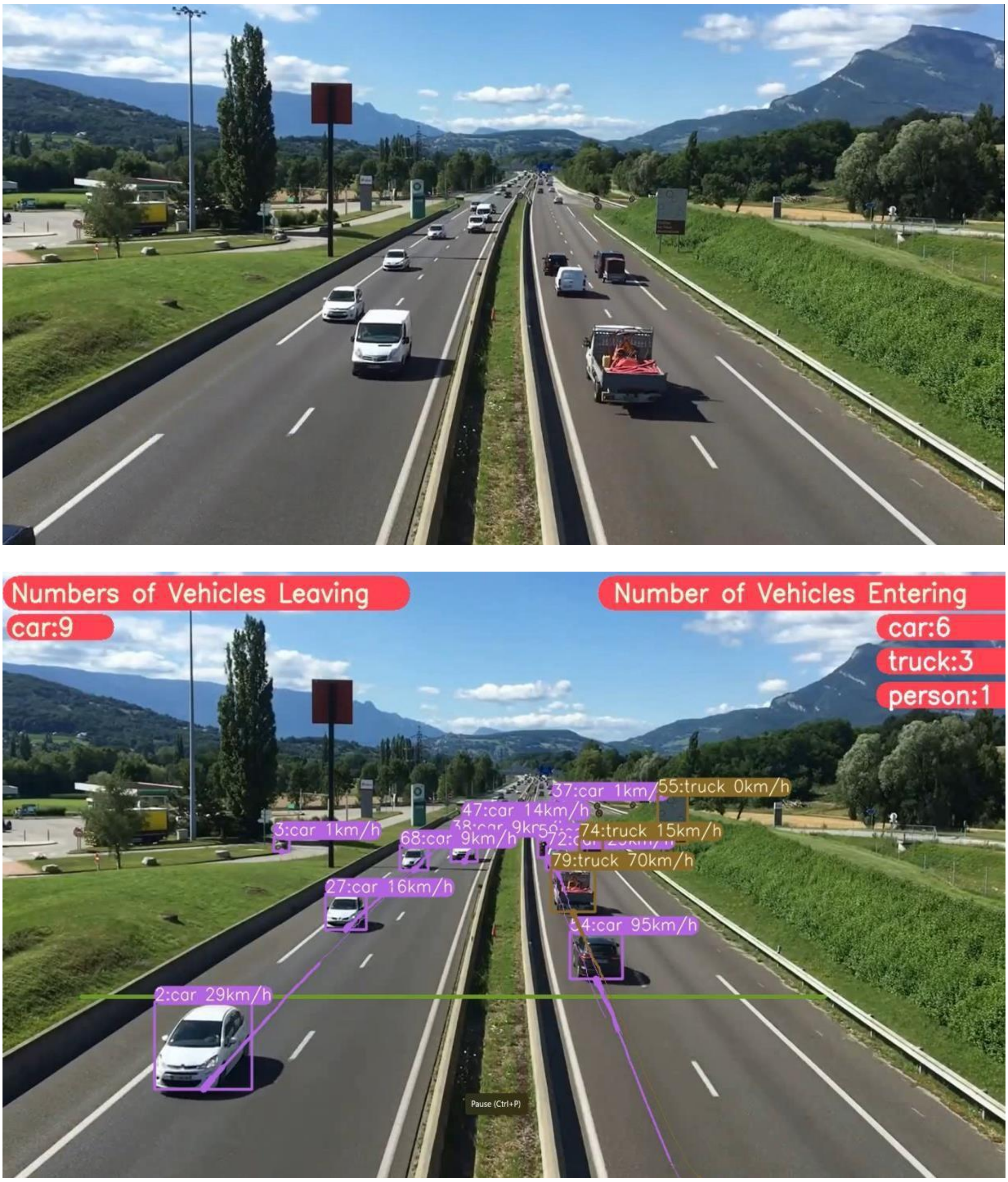


**Figure 4** Test Case and Result

## 9. Conclusion

A real-time traffic monitoring system is proposed, utilizing a virtual detection zone and YOLO v8, to enhance the efficiency of vehicle counting, detection, and classification. Additionally, the distance and time traveled by each vehicle are utilized to estimate their speed. YOLO v8 demonstrates superior accuracy in both classification and detection when compared to other algorithms, and deep sort is employed alongside YOLO v8 to track vehicles effectively. As a result, this method can be effectively applied to real-life scenarios for vehicle counting, speed estimation, and classification.

The YOLOv8 and Deep SORT approach offers a promising solution for vehicle speed estimation and counting, contributing to the broader landscape of smart city technologies. Despite existing challenges, continuous development and optimization efforts can further enhance the system's effectiveness in real-world scenarios, addressing the evolving needs of traffic management and urban planning. In summary, the integration of YOLOv8 for object detection and Deep SORT for object tracking in vehicle speed estimation and counting represents a potent solution for real-time traffic monitoring.

The combination offers strengths in accurate and simultaneous object detection, robust online tracking, and the ability to provide comprehensive insights into traffic dynamics. However, challenges such as sensitivity to environmental conditions and occasional tracking errors need careful consideration. Continuous optimization and potential integration with additional sensors are essential for adapting to evolving traffic scenarios. Overall, this approach holds promise for contributing to smart city initiatives and enhancing traffic management strategies.

## Compliance with ethical standards

### *Disclosure of conflict of interest*

No conflict of interest to be disclosed.

## References

[1] Raju, P. Daniel Ratna, and G. Neelima. "Image segmentation by using histogram thresholding." International Journal of Computer Science Engineering and Technology 2, no.1 (2012): 776-779.

[2] Amit Ghosh, MD. shahinuzzaman, Hamudi Hasan Sonet, Swakkhar Shatabda. "An Adaptive Video-based Vehicle Detection, Classification, Counting, and Speed-measurement System for Real-time Traffic Data Collection".2019 IEEE Region 10 Symposium (TENSYMP).

[3] Akhil Reddy Kalva, Jyothi Swarup Chelluboina, B. Bharathi." Smart Traffic Monitoring System using YOLO and Deep Learning Techniques". 2023 7th International Conference on Trends in Electronics and Informatics (ICOEI).

[4] David Fernandez Liorca, Antonio Hernandez, Ivan Garcia Daza." Vision-based vehicle speed estimation: A survey". © 2021 The Institution of Engineering and Technology © 2021 The Authors. IET Intelligent Transport Systems published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology.

[5] Xinyue Zhao, Guangli Wang, Zaixing He." A survey of moving object detection methods: A practical perspective". Neurocomputing Volume 503, 7 September, 2022, Pages 28-48.