**TRAFFIC RULES VOILATION DETECTION SYSTEM**

***Project synopsis submitted to***

***Guru Jambheshwar University of Science and Technology, Hisar for the partial award of the degree***

***of***

**Bachelor of Technology**

**in Computer Science and Engineering (Artificial Intelligence and Machine Learning)**

***by***

**Nidhi Sahani Dr. Ritu**

**(210010150056) Associate Professor**

**Himanshi (210010150059)**

****

**Department of Computer Science & Engineering GURU JAMBHESHWAR UNIVERSITY OF SCIENCE AND**

**TECHNOLOGY, HISAR**

**January, 2025**

# DECLARATION

I , **Nidhi Sahani(210010150056)** and my team member **Himanshi*(*2100101500059)**, affirm that the work contained in this major project synopsis is original and done by me. This work has not been submitted to any other institute for the purpose of obtaining degree and we have followed the ethical practices and other guidelines provided by the Computer Science and Engineering Department in preparing the major-project synopsis.

***Signature***

Nidhi Sahani (210010150056)

Himanshi (210010150059)

Department of Computer Science and Engineering

Guru Jambheshwar University of Science and Technology, Hisar

# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **Sr.No.** | **Topic** | **Page No.** |
| 1. | Introduction | 1 |
| 2. | Background Details and Literature Reviews | 2-3 |
| 3. | Problem Formulation and Objectives | 4-5 |
| 4. | Methodology and Tool used | 6-8 |
| 5. | Reference/ Biblography | 9 |

**INTRODUCTION**

In today's rapidly growing urban environments, traffic management and enforcement are crucial challenges due to the increasing number of vehicles on the roads. Traditional methods like manual patrols and stationary cameras are often inefficient and can lead to delays or errors in detecting violations. This necessitates the development of automated systems that can detect and report traffic violations in real-time.

### The REAL-TIME TRAFFIC RULES VIOLATION DETECTION SYSTEM is an

innovative solution that utilizes computer vision and machine learning to detect traffic violations, specifically speeding, using live video surveillance feeds. By tracking vehicles' movements and estimating their speed, the system can effectively monitor and identify violations without human intervention.

The project integrates multiple advanced technologies, including vehicle detection, real- time tracking, and speed estimation. It works by continuously analyzing video footage, detecting vehicles, tracking their positions across frames, and calculating their speed. If a vehicle exceeds the speed limit, the system flags it for potential violation, offering a more efficient alternative to manual enforcement.

This automated approach offers several advantages: it minimizes the need for human monitoring, improves accuracy in violation detection, and provides real-time alerts for enforcement. As part of the growing trend of smart cities and intelligent traffic systems, this solution promises to enhance road safety and streamline traffic management.

# BACKGROUND DETAILS AND LITERATURE REVIEWS

## Background Details:

The exponential increase in urban traffic, combined with inadequate traffic management systems, has resulted in a growing need for effective and efficient methods to monitor and enforce traffic rules. Traditional methods of enforcement, such as traffic police officers manually monitoring and ticketing violators, often lead to human error, inconsistencies, and are labor-intensive. Additionally, these methods are reactive rather than proactive, failing to detect violations in real-time. As a result, there is a pressing need for automated systems that can provide continuous, real-time monitoring and enforcement of traffic laws.

The challenge of speed violations, one of the most prevalent traffic offenses, is particularly critical. Speeding contributes to a significant number of road accidents and fatalities, with consequences that often extend beyond the individual driver to affect the broader public. Conventional speed detection methods, such as radar guns or speed cameras, often lack the ability to monitor multiple lanes or enforce speed limits in real- time over large areas. Automated systems, powered by computer vision and machine learning, can address these shortcomings by continuously monitoring and analyzing traffic data to identify violations as they occur.

Technologies like computer vision, object detection, and tracking algorithms have emerged as solutions to automate the detection of traffic violations. With the capability to process live video streams, these systems can detect vehicles, track their movements, and calculate their speed to determine if they are exceeding the permissible limits. Furthermore, such systems can be integrated with databases to record violations, generate reports, and trigger immediate alerts for further action by law enforcement. These advancements offer a more scalable, accurate, and efficient alternative to traditional traffic monitoring methods, and can ultimately improve road safety by providing faster and more reliable enforcement.

This project, the REAL-TIME TRAFFIC RULES VIOLATION DETECTION

SYSTEM, seeks to leverage state-of-the-art computer vision and tracking algorithms to provide real-time detection of traffic violations, specifically speeding. By processing high-definition video feeds and analyzing vehicle movements, this system aims to automate the enforcement of traffic rules, offering real-time alerts for violations and contributing to safer, more efficient roadways.

The increasing number of vehicles on the roads has made traffic management an increasingly complex and demanding task. With growing traffic congestion, the need for efficient traffic monitoring systems has become more critical than ever. Traditional methods of monitoring traffic violations, such as manual patrols, traffic police officers, and stationary cameras, often have limitations. These methods may be prone to human error, require significant manpower, and fail to provide real-time monitoring and reporting.

One of the most common traffic violations is speeding, which not only poses a danger to the driver but also significantly increases the likelihood of accidents and fatalities. To address this, there has been a growing interest in developing automated systems that can monitor traffic violations, especially speeding, in real-time, providing an efficient and scalable solution for modern cities. The advent of technologies like computer vision, deep learning, and high-resolution video surveillance has made it possible to develop systems that automatically track vehicles, measure their speed, and detect violations without human intervention.

### The REAL-TIME TRAFFIC RULES VIOLATION DETECTION SYSTEM

leverages these advancements in computer vision and machine learning to provide a solution for real-time detection and reporting of traffic violations. The system processes live video feeds, identifies vehicles, tracks their movements, and calculates their speed using algorithms designed to estimate motion based on the changes in vehicle positions across frames.

## Literature Review:

A significant body of work has been done in the field of traffic monitoring and violation detection using computer vision. One prominent approach involves the use of **object detection algorithms** to identify vehicles in video feeds. Methods such as Haar Cascade Classifiers, which are used in this project, have been widely adopted for their ability to detect vehicles in real-time under various environmental conditions. Haar cascades are often combined with machine learning models to improve their accuracy in detecting objects, even in complex scenes.

In recent years, the integration of **deep learning** techniques, particularly **Convolutional Neural Networks (CNNs)**, has shown promise in vehicle detection tasks. These methods outperform traditional approaches in terms of accuracy and robustness, especially when dealing with cluttered backgrounds or varying lighting conditions. However, they tend to require substantial computational resources, making them less feasible for real-time applications without optimized hardware.

For speed estimation, techniques such as **optical flow** and **tracking algorithms** like the **Kalman filter** and **correlation-based trackers** (like dlib used in this project) are commonly used. These trackers follow the detected vehicles across multiple frames, estimating their velocity by analyzing the displacement of the vehicles over time. Optical flow methods track pixel movements between consecutive frames, while tracking algorithms focus on matching vehicles in succesive frames, offering higher accuracy for speed detection in real-time systems.

Several studies have explored the application of traffic rule violation detection systems in smart city projects. In one study by **Bai et al. (2019)**, the authors proposed an intelligent traffic monitoring system using deep learning for vehicle classification and speeding detection, integrating sensors with computer vision. Their system demonstrated the potential of deep learning models in identifying speeding vehicles, but also highlighted the computational challenges in processing real-time data from high- resolution cameras.

# PROBLEM FORMULATION AND OBJECTIVES

## Problem Formulation :

The increasing number of vehicles on the roads, along with insufficient traffic monitoring and enforcement mechanisms, leads to an uptick in traffic violations, particularly speeding. Speeding is a major cause of road accidents and fatalities, necessitating an automated, real-time solution to detect and address such violations effectively. Traditional traffic monitoring methods, such as manual enforcement by police or static speed cameras, are often insufficient due to human error, high labor costs, and limited coverage. These methods are reactive and fail to identify violations as they occur, creating delays in enforcement and posing safety risks to other road users. The challenge addressed in this project is the need for a real-time automated system that can detect traffic violations, particularly speeding, using video data. The system must be capable of accurately detecting vehicles, tracking their movement, estimating their speed, and providing alerts or reports on violations. This must be done in real-time to enable immediate action. The system must also be scalable, adaptable to various road conditions and environments, and computationally efficient enough to run on standard hardware without significant delay.

To solve this problem, the project leverages a combination of computer vision techniques, such as Haar Cascade Classifier for vehicle detection, and dlib correlation tracker for vehicle tracking. Speed estimation is carried out by calculating the displacement of vehicles in consecutive frames and converting this into real-world distance using the pixels per meter (ppm) ratio. The system then computes the speed of each vehicle and checks for violations by comparing the estimated speed with the speed limit. The entire process happens in real-time, with the system providing speed-related alerts when necessary.

## Proposed Solution:

The proposed solution is a REAL-TIME TRAFFIC RULES VIOLATION DETECTION SYSTEM, which incorporates the following key features:

Vehicle Detection: The system uses a Haar Cascade Classifier to detect vehicles from video frames.

* Object Tracking: Using the dlib correlation tracker, the system tracks vehicles across frames, ensuring accurate identification of vehicles as they move.
* Speed Estimation: The system estimates the speed of detected vehicles by calculating the displacement between consecutive frames and converting this into real-world distance.
* Violation Detection: Speeding violations are detected by comparing the estimated vehicle speed with predefined speed limits. The system then generates alerts if the speed exceeds the permissible limits.
* Real-Time Processing: The entire system operates in real-time, ensuring immediate detection of violations as they occur.
* Scalability: The system can handle multiple vehicles simultaneously, making it scalable for use in urban environments with heavy traffic. This solution provides a comprehensive, automated approach to monitoring traffic in real time, enhancing the accuracy and efficiency of traffic law enforcement.

## Objectives :

The primary objectives of the REAL-TIME TRAFFIC RULES VIOLATION DETECTION SYSTEM are as follows:

* Real-time Traffic Monitoring: Develop a system that can process live video feeds to detect and track vehicles in real-time. Ensure the system can monitor multiple vehicles simultaneously without compromising accuracy.
* Vehicle Detection: Implement a reliable and efficient method for detecting vehicles in video frames using the Haar Cascade Classifier, enabling accurate vehicle identification.
* Object Tracking: Employ dlib correlation tracker to track detected vehicles across multiple frames, ensuring consistent and precise tracking of vehicles as they move through the scene.
* Speed Estimation: Calculate the speed of detected vehicles by analyzing the change in position between consecutive frames, using the pixels per meter (ppm) ratio to convert pixel displacement into real-world distance. Ensure accurate speed measurement under various conditions such as different vehicle types, road environments, and lighting conditions.
* Violation Detection and Alerts: Automatically identify vehicles that exceed the speed limit and generate alerts or reports for traffic enforcement.Provide accurate feedback on violations in real-time, contributing to improved road safety.
* System Scalability and Performance: Design the system to handle multiple vehicles in real-time, ensuring it performs well even in high-traffic situations. Optimize the system to run efficiently on standard hardware, ensuring smooth operation without significant delays or processing lag.
* User-Friendly Output: Provide clear, readable output, including vehicle speeds and violation notifications, through visual cues such as text overlays or alert pop- ups. Ensure the system can be easily integrated with existing traffic enforcement frameworks, allowing for immediate reporting of violations.

# METHODOLOGY AND TOOLS USED

## System Architecture:

### The REAL-TIME TRAFFIC RULES VIOLATION DETECTION SYSTEM

employs a combination of computer vision techniques and machine learning models to detect, track, and monitor vehicles in real-time. The methodology consists of the following major steps:

* 1. **Video Input**: The system receives video footage (live or recorded) as input, which serves as the data source for processing. This footage is captured through video surveillance cameras or any other relevant devices that can stream video data.
  2. **Vehicle Detection**: The system uses the **Haar Cascade Classifier** to detect vehicles in each frame of the video. Haar Cascades are pre-trained classifiers that efficiently identify objects, such as vehicles, within a frame.
  3. **Object Tracking**: Once a vehicle is detected, the system uses **dlib's correlation tracker** to track the vehicle across successive frames. This tracker maintains the identity of each vehicle, allowing the system to monitor the vehicle's movement across the video.
  4. **Speed Estimation**: The system estimates the vehicle's speed by calculating the displacement between two consecutive positions of the vehicle in the video frames. This displacement is converted into real-world measurements using a **pixels per meter (ppm)** ratio, allowing the system to estimate the vehicle's speed in kilometers per hour (km/h).
  5. **Violation Detection**: The system compares the estimated vehicle speed against the predefined speed limits. If a vehicle exceeds the speed limit, the system flags it as a violation and optionally generates alerts or reports.
  6. **Real-Time Processing**: The entire process from detection to violation alert happens in real-time, allowing the system to monitor traffic continuously and efficiently.

## Tools Used :

To implement the system, various programming languages, libraries, and frameworks have been employed. These tools provide the functionality necessary to process video, detect objects, track moving vehicles, and calculate speed. Below are the main tools used:

### Programming Language: Python

* + - **Python** was chosen due to its simplicity, flexibility, and extensive support for computer vision and machine learning libraries. Python's rich ecosystem makes it ideal for rapid prototyping and implementing real- time systems.

### Computer Vision Libraries

* + - **OpenCV (Open Source Computer Vision Library)**: OpenCV is a powerful computer vision library used to process video frames, apply image transformations, and perform tasks like vehicle detection, speed estimation, and frame capture. It enables real-time video analysis and

visualization of the processed output.

* + - **dlib**: dlib is a robust toolkit for machine learning and computer vision tasks, particularly for object tracking. In this system, dlib’s **correlation tracker** is used to track the detected vehicles across frames, maintaining the vehicle IDs and positions over time.

### Vehicle Detection

* + - **Haar Cascade Classifier**: Haar Cascades are used for vehicle detection due to their efficiency in detecting objects in images. They work by classifying positive and negative images to identify patterns in vehicles at various scales and angles.

### Speed Estimation

* + - The speed estimation technique relies on the distance traveled by the vehicle in terms of pixel displacement across frames. The conversion from pixel displacement to real-world distance is done using a **pixels per meter (ppm)** ratio, which must be manually calibrated for different scenarios (i.e., road types, camera angles, etc.).

### Real-Time Performance and Video Output

* + - **cv2.VideoCapture**: This function is used to capture the video input from the camera or video file.
    - **cv2.VideoWriter**: This is used to save the output processed video to an AVI file. It ensures that the real-time analysis is recorded and can be reviewed later.
    - **cv2.imshow**: OpenCV’s imshow function displays the processed video frames in real-time, showing detected vehicles, their speeds, and violation alerts (if any).

### Hardware

* + - **Standard PC with Camera/Video Input**: The system can run on a standard computer with a connected camera or a recorded video file. The computation load, which involves real-time video processing, is supported by the CPU and GPU of the system.

## Workflow of the System :

### Capture and Preprocessing:

* + - The system starts by capturing frames from the video input source. Each frame is resized to a defined resolution (e.g., 1280x720 pixels) to ensure consistent processing. Grayscale conversion is applied for vehicle detection to simplify the processing.

### Vehicle Detection:

* + - The **Haar Cascade Classifier** is used to detect vehicles within the grayscale image. The classifier scans the image at multiple scales to detect vehicles that appear at different sizes.

### Vehicle Tracking:

* + - For each vehicle detected, the system initializes a **dlib correlation tracker** that follows the vehicle’s movement across subsequent frames. The tracker ensures that the same vehicle is identified consistently over time, even if there are slight changes in appearance or position.

### Speed Calculation:

* + - The displacement between consecutive frames is measured in pixels. This value is converted into real-world units (meters) using a calibration

factor, **pixels per meter (ppm)**.

### Violation Detection:

* + - The calculated speed is compared against the predefined speed limit. If a vehicle exceeds the limit, the system identifies it as a violation and provides an alert on the output video.

### Visualization and Reporting:

* + - The system displays the video output with annotations showing the detected vehicles, their speeds, and any violations. Alerts are triggered for vehicles exceeding the speed limit. The processed video can be saved for further analysis.

## Real-Time Constraints and Optimization :

* **Real-Time Processing**: The system is designed to operate in real-time, requiring optimization to process video frames at a high frame rate (FPS). This is crucial to ensure that vehicles are detected, tracked, and their speeds estimated without noticeable delays.
* **Efficient Object Tracking**: The **dlib correlation tracker** was chosen for its efficiency and accuracy in tracking objects across frames. Unlike traditional tracking algorithms that may struggle with fast-moving objects or occlusions, dlib provides robust tracking performance even in challenging conditions.
* **Speed Estimation Calibration**: The **ppm ratio** is calibrated based on the real- world camera setup. It is a critical parameter that determines the accuracy of the speed calculation. Variations in camera angle, road type, and vehicle distance from the camera affect this ratio, requiring periodic adjustment.



# REFERENCES/ BIBLOGRAPHY

1. A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz, “Robust real- timeunusual event detection using multiple fixed-location monitors,” IEEETransactions on Pattern Analysis and Machine Intelligence, vol. 30,no. 3, pp. 555–560, March 2008.
2. B. Duan, W. Liu, P. Fu, C. Yang, X. Wen, and H. Yuan, “Real- time on-road vehicle and motorcycle detection using a single camera,” in Procs.of the IEEE Int. Conf. on Industrial Technology (ICIT), 10-13 Feb 2009,pp. 1–6.
3. W. Hu, T. Tan, L. Wang, and S. Maybank, “A survey on visualsurveil-lance of object motion and behaviors,” IEEE Transactions onSystems, Man, and Cybernetics, Part C: Applications and Reviews,vol. 34, no. 3, pp. 334–352, Aug 2004.
4. J. Chiverton, “Helmet presence classification with motorcycle detec-tionand tracking,” Intelligent Transport Systems (IET), vol. 6, no. 3, pp.259–269, September 2012.
5. Z. Chen, T. Ellis, and S. Velastin, “Vehicle detection, tracking andclas-sification in urban traffic,” in Procs. of the IEEE Int. Conf. onIntelligent Transportation Systems (ITS), Anchorage, AK, Sept 2012,pp. 951–956.
6. R. Silva, K. Aires, T. Santos, K. Abdala, R. Veras, and A. Soares,“Automatic detection of motorcyclists without helmet,” in Com-putingConf. (CLEI), XXXIX Latin American, Oct 2013, pp. 1–7.