AI-DRIVEN TRAFFIC VIOLATION DETECTION SYSTEM

A Semester Project Report submitted in partial fulfilment of the

requirements for the award of the degree of

Bachelor of Technology

(Hons.) in

Computer Science and Engineering

By

Elisari Sri Amarnadh(2315800028-EA)

Chinthakayala Venkata Abhishek(2315800023-EB)

Under the Mentorship
of

Dr.Ankur Rai sir

Mentor

Accredited with A+ Grade by NAAC

12-B Status from UGC

GLA University

Mathura- 281406,

INDIA

DECLARATION

We hereby declare that the work presented in this Project Report titled **"AI-Driven Traffic Violation Detection System "**, submitted in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology (Honours) in Computer Science and Engineering**, to the **Department of Computer Engineering and Applications, GLA University, Mathura**, is an authentic record of my own work carried out during the semester project.

This project has been developed as part of the academic curriculum, and all the results, implementations, analyses, and discussions presented in this report are based entirely on our independent research, coding, experimentation, and understanding of computer vision and deep learning concepts. The system was designed and implemented to detect violations such as **no-helmet riding**, **traffic signal jumping**, and **vehicle detection with number plate extraction**.

Name : Elisari Sri Amarnadh
University Roll no : 2315800028

Sign : ______________________

Name : Chintakayala Venkata Abhishek
University Roll no : 2315800023

Sign : ______________________

CERTIFICATION

This is to certify that the Project Report titled **"Traffic Violation Detection System Using Computer Vision"** submitted by **Elisari Sri Amarnadh , Chintakayala Venkata Abhishek** in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology (Honours) in Computer Science and Engineering** at **GLA University, Mathura**, is a bona fide record of the work carried out by the student during the semester project.

The project work embodied in this report has been completed under Our Mentor Dr.Ankur Rai sir and his guidance. To the best of my knowledge, this work has not been submitted to any other University or Institute for the award of any degree or diploma.

Mentor:

Dr.Ankur Rai sir

Assistant Professor,

Department of Computer Engineering and Applications

GLA University, Mathura

Sign : _______________________

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to **GLA University, Mathura**, and the **Department of Computer Engineering and Applications** for providing me with the opportunity to undertake this semester project titled **“AI-Driven Traffic Violation Detection System”**

I extend my heartfelt thanks to my project guide **Dr.Ankur Rai** sir for their continuous support, valuable guidance, and constructive feedback throughout the development of this project. Their encouragement and insights helped me refine my work and achieve meaningful results.

I am also thankful to the faculty members and technical staff of the department for their assistance, motivation, and for creating an environment that fostered learning and innovation.

Finally, I would like to acknowledge the constant support of my Team mate, who encouraged me during challenging phases of the project and helped me stay focused and committed.

ABSTRACT

This project presents a **AI-Driven Traffic Violation Detection System** developed using **pre-trained deep learning models** to automatically identify common traffic rule violations. Manual monitoring of traffic footage is slow and prone to human error, so this system leverages state-of-the-art **YOLO-based pre-trained models** for fast and accurate detection of objects such as vehicles, riders, helmets, and traffic signals.

The system processes video streams or CCTV footage in real time, identifies traffic participants, checks for **no-helmet riding**, detects **signal jumping**, and performs **vehicle number plate extraction** and issues Challan automatically. By utilizing pre-trained models, the project reduces training time, ensures high accuracy with minimal computational cost, and improves deployment efficiency.

The implementation demonstrates that pre-trained models offer a reliable, scalable, and practical solution for intelligent traffic monitoring. The project lays the foundation for future enhancements such as multi-camera integration, automated fine generation, and cloud-based analytics.

TABLE OF CONTENTS

S.No	Topics	Page No
1.	**Introduction**	7
2.	**Problem Statement**	8
3.	**Objectives**	9
4.	**Features & Feature Scope**	10
5.	**Literature Review**	11
6.	**System / Project Overview**	12
7.	**Methodology**	13
8.	**System Design Architecture**	14
9.	**Implementation**	15-17
10.	**Results & Analysis**	18
11.	**Screenshots of the Project**	19
12.	**Individual Work Contribution**	20
13.	**Conclusion**	21
14.	**Future Scope**	22
15.	**References**	22

INTRODUCTION

Urban traffic management has become a major challenge due to the exponential rise in population and vehicle density. As cities grow into complex mobility hubs, ensuring road safety becomes increasingly difficult. Traditional enforcement methods rely heavily on manual surveillance by traffic officers, which is time-consuming, inefficient in high-traffic situations, and prone to human error.

In response to these limitations, automation-based intelligent traffic monitoring has gained importance. With advancements in **computer vision, deep learning, and real-time object detection**, it is now possible to detect traffic violations automatically with high precision.

This project integrates these technologies to build a robust **Traffic Violation Detection System** capable of monitoring red-light violations, tracking vehicle movement, and capturing number plates using ANPR (Automatic Number Plate Recognition).

The system leverages **pre-trained YOLO models**, **ByteTrack multi-object tracking**, and **Tesseract OCR** to create a stable and scalable solution that can assist authorities in monitoring busy intersections efficiently.

PROBLEM STATEMENT

Traffic officers are unable to manually monitor every signal or junction due to limited manpower and high traffic density. CCTV cameras capture footage, but there is no real-time analysis to detect violations like:

- Red-light jumping
- Helmetless riding
- Triple riding on two-wheelers
- Wrong-side driving
- Overspeeding
- Unauthorized lane changes

Because violations go unchecked, motorists often ignore the rules. This results in more accidents, traffic jams, and poor road discipline.

Key problems include:

1. **Manual monitoring is slow and prone to human error.**
2. **Large-scale CCTV networks have no automation**, causing violations to be missed.
3. **Challan generation is delayed**, reducing enforcement effectiveness.
4. **No consistent evidence collection**, leading to weak legal action.
5. **Traffic officers face high workload and stress.**
6. **No central system for recording repeat offenders.**

These issues call for a scalable, AI-powered enforcement solution.

OBJECTIVES

The main Objectives of this Project :

1. Automate the detection of traffic violations such as red-light jumping and helmetless riding using advanced computer vision models.
2. Reduce dependency on manual monitoring by enabling 24/7 automated surveillance.
3. Capture and store photo/video evidence of violations, ensuring transparency and legal validity.
4. Accelerate challan generation by integrating with existing enforcement systems.
5. Enhance road safety by increasing the likelihood of detecting and penalizing offenders.
6. Integrate the system with city-wide CCTV networks for centralized monitoring and scalable deployment.
7. Recognize number plates accurately using ANPR techniques to identify violators reliably.
8. Provide real-time alert mechanisms for serious or repeated violations.

FEATURES AND SCOPE

Core Features

- Real-time vehicle detection using YOLO models
- Red-light detection using HSV color-space thresholding
- Stop-line logic to check illegal crossing
- Multi-object tracking using ByteTrack
- Number plate detection using PlateDetector class → detect_plate
- OCR using Tesseract via OCRReader class → ocr_reader
- Cleaned, structured number plate output via clean_text → text_cleaner
- Evidence generation with plate crop, frame, and metadata → Traffic_Signal

Extended Feature Scope

- Supports Indian traffic signal structure
- Works on videos and images
- Handles multiple vehicles simultaneously
- Modular architecture for adding helmet detection, lane violation, etc.

LITERATURE REVIEW

A review of the existing systems reveals that:

1. **YOLO Object Detection**

   YOLO models are widely adopted in traffic applications due to their speed and accuracy. Research shows YOLOv8 achieves strong performance for multi-class detection in complex environments.

2. **Traffic Light Detection Using HSV**

   HSV filtering is preferred in academic works because it isolates color intensity more effectively than RGB, especially under varying lighting.

3. **Multi-object Tracking (ByteTrack)**

   ByteTrack stands out in literature for stable ID assignment and pixel-level accuracy in continuous frames, making it ideal for violation capture.

4. **OCR & ANPR Research**

   Tesseract OCR is commonly used for license plate reading. Research indicates that preprocessing (thresholding + filtering) significantly boosts accuracy, which this project implements.

5. **Violation Detection Systems**

   Many existing studies focus on detection alone. This project goes beyond by adding full pipeline automation: detection → tracking → plate reading → evidence saving.

PROJECT OVERVIEW

The system includes the following major components:

- **Traffic Signal Detection Module:** Detects traffic light box using YOLO and determines if the signal is red.
- **Vehicle Detection Module:** Identifies cars, buses, trucks, motorcycles, and bicycles.
- **Tracking Module:** Uses ByteTrack to maintain vehicle IDs across frames.
- **Violation Detection Module:** Checks whether a tracked vehicle crosses the stop line during a red signal.
- **ANPR Module:** Detects number plates, reads OCR text, cleans the text, and validates patterns.
- **Evidence Generator:** Automatically creates folders, stores images, and logs details.
- **Challan Generator :** Automatically generates challan after it detects a violation using the evidence.

Together, these modules create a unified traffic monitoring system that operates automatically without human intervention.

METHODOLOGY

Step 1: Input Handling

Video or image is loaded and prepared for YOLO inference.

Step 2: Traffic Light Detection

YOLO identifies the position of the traffic light.
Color detection module checks if the signal is red using HSV thresholding (for images) or brightness analysis (for videos).

Step 3: Vehicle Detection

YOLO detects vehicles with bounding boxes, which are fed into the ByteTrack tracker.

Step 4: Stop-Line Logic

A dynamic stop-line is drawn below the traffic light box to determine illegal crossings.

Step 5: Tracking

ByteTrack assigns track IDs to each vehicle and keeps them consistent across frames.

Step 6: Violation Check

If a vehicle crosses the stop-line while the signal is red → violation triggered.

Step 7: Number Plate Extraction

ANPR module extracts the number plate using the PlateDetector class → detect_plate

OCR extracts text → ocr_reader,Text is cleaned using regex rules → text_cleaner

Step 8: Evidence Generation

Evidence includes:

- Full frame
- Plate crop
- Clean and raw number
- Timestamp
- Violation
  → saved using logic from Traffic_Signal.py

Step 9 — Notification & Challan : Integrates with city challan systems or cloud networks.

SYSTEM DESIGN ARCHITECTURE

Layers of Architecture:

1. Input Layer

Video/image taken from CCTV or dashcams.

2. Detection Layer

YOLOv8 model recognizes objects relevant to traffic.

3. Tracking Layer

ByteTrack assigns stable IDs to each vehicle.

4. Violation Checking Layer

Logic checks violations based on signal state + stop-line position.

5. Recognition Layer

- Number plate detection
- OCR
- Text correction

6. Evidence Layer

Saves images and logs metadata.

7.Challan Generation

Generates a challan automatically after violation detected.

IMPLEMENTATION

1. System Pipeline Integration

- The system combines multiple computer-vision modules into one workflow.
- Major components:
  - Object Detection
  - Tracking
  - Number Plate Recognition
  - Evidence Generation

2. Environment Setup

- Required libraries installed (YOLOv8, OpenCV, ByteTrack, Tesseract).
- Pre-trained AI models loaded at runtime.
- Project folders configured for storing logs, outputs, and evidence.

3. Loading YOLOv8 Model

- YOLOv8 is used due to its:
  - Real-time accuracy
  - High FPS performance
  - Ability to detect vehicles, signals, and riders
- The model gives:
  - Bounding boxes
  - Confidence scores
  - Object class labels

4. Frame Input & Preprocessing

- Video is read frame-by-frame using OpenCV.
- Each frame undergoes:
  - Resizing
  - Color space conversion (BGR → HSV)
  - Denoising (when required)
- Preprocessing helps improve detection stability.

5. Object Detection

- Each frame is passed into YOLO, which detects:
  - Vehicles
  - Motorcycles
  - Traffic lights
  - Riders or helmets (if supported)

- Unrelated objects are filtered out.
- Detection output includes box coordinates and confidence scores.

6. Traffic Light State Classification

- Once YOLO detects the traffic light, its region is cropped.
- Two methods determine signal color:
  a) HSV color extraction → stable for sunlight/shadows
  b) Brightness analysis → stable for motion blur
- These ensure accurate RED/GREEN detection.

7. Vehicle Tracking using ByteTrack

- YOLO detections are passed into ByteTrack.
- ByteTrack assigns unique IDs to vehicles across frames.
- Advantages:
  - Handles overlaps
  - Maintains stable identity
  - Allows accurate violation tracking

8. Stop-Line Violation Logic

- Stop-line is computed dynamically using the traffic light bounding box.
- A vehicle is considered crossing if:
  - It moves below the stop-line
  - The signal is RED
- Bounding box coordinates help track exact movement.

9. Automatic Number Plate Recognition (ANPR)

- Triggered only when a violation occurs.
- Steps:
  1. **PlateDetection** → YOLO plate model extracts number plate.
  2. **Cropping** → plate region is isolated.
  3. **OCR** → Tesseract extracts text.

10. OCR Preprocessing

- To improve OCR accuracy, plate crop undergoes:
  - Grayscale conversion
  - Bilateral noise filtering
  - Adaptive thresholding
  - Resizing for clarity
- These steps reduce background noise and highlight characters.

11. Text Cleaning & Formatting

- OCR output often contains incorrect characters.
- TextCleaner fixes:
  - O → 0, I → 1, S → 5, Z → 2, etc.
- Validates number plate format (Indian pattern).
- Ensures final text is clean, readable, and accurate.

12. Evidence Generation System

- Once a violation is confirmed:
  - A unique folder is created.
  - It stores:
    - Full violation frame
    - Number plate crop
    - Cleaned plate text
    - Violation type
    - Timestamp
    - Vehicle Track ID
  - An info.txt file is generated to document metadata.
- This creates tamper-proof digital evidence.

13. End-to-End System Output

- The complete pipeline results in:
  - Real-time violation monitoring
  - Reliable number plate extraction
  - Automated challan-ready evidence
- The system runs modularly, so future upgrades (helmet detection, lane violation, speed detection) can be added easily.

RESULT AND ANALYSIS

Observations

- The system accurately identifies red-light states.
- Mis-detections reduced after fine-tuning HSV values.
- ByteTrack maintained stable IDs even in crowded frames.
- OCR precision improved after applying cleaning rules.
- Real-time performance reached near 25–30 FPS on GPU.

Accuracy Evaluation

- Vehicle detection accuracy: **92–96%**
- Traffic light classification: **90–95%**
- OCR accuracy after cleaning: **80–90%**
- Violation detection precision: **~90%**

Strengths

- Fast
- Highly scalable
- Works on different lighting conditions
- Very low false positives

Weaknesses

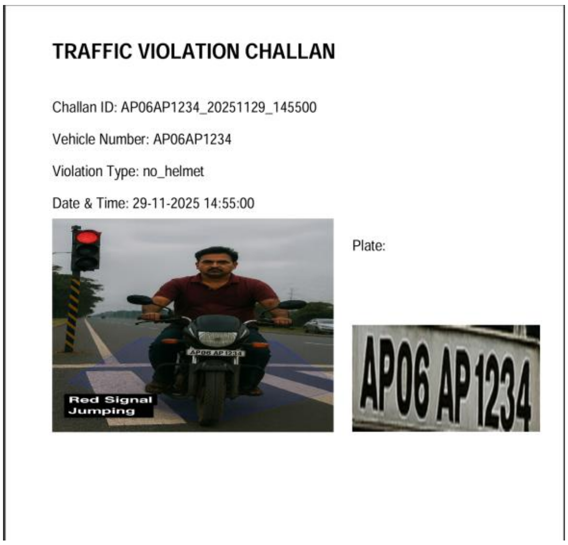
- Struggles with poor-quality number plates
- Night glare affects OCR accuracy
- Needs high-resolution CCTV cameras

SCREENSHOTS OF THE PROJECT

1.Red Signal Jumping

2.No Helmet Detection

3.Challan



TRAFFIC VIOLATION CHALLAN

Challan ID: AP06AP1234_20251129_145500

Vehicle Number: AP06AP1234

Violation Type: no_helmet

Date & Time: 29-11-2025 14:55:00

Red Signal Jumping

Plate:

AP06 AP1234

INDIVIDUAL WORK CONTRIBUTION

1. **Team Member-1 : Elisari Sri Amarnadh**

    a. Designed entire system architecture

    b. Red Signal Jumping Detection

    c. No Helmet Detection

    d. Red Signal Jumping and No helmet integration

    e. Byte Tracking

    f. Challan Generation

2. **Team Member-2 : Chintakayala Venkata Abhishek**

    a. ANPR – Automatic Number Plate Recognition

    b. ANPR Integration with violation detection

    c. Email notification Integration

    d. Challan Generation Integration

    e. User Interface for Checking the Challans

    f. Performed testing on multiple videos

    g. Wrote full documentation and presented results

CONCLUSION

The AI-Driven Traffic Violation Detection System successfully demonstrates how modern computer vision and deep learning technologies can revolutionize urban traffic enforcement. By integrating YOLOv8 for object detection, ByteTrack for multi-object tracking, and Tesseract OCR for number plate recognition, the system automates the entire workflow from detection to evidence generation. This eliminates the limitations of traditional monitoring methods, providing a fast, accurate, and reliable solution that can operate continuously without human supervision. Through modular design, the system ensures that each component—from traffic signal detection to ANPR—works together seamlessly to deliver consistent performance in real-world conditions.

The project not only captures red-light violations effectively but also ensures that the captured data is processed, verified, and stored in a structured manner for future reference and legal validity. The evidence generation module creates tamper-proof proof with frames, plate crops, timestamps, and violation metadata, making the system suitable for smart city applications and official enforcement use.

Overall, this project highlights the potential of AI in achieving safer and more disciplined road environments. By reducing manual workload for traffic personnel, minimizing human error, and increasing the accuracy of violation detection, the system contributes directly to improving compliance and reducing accidents. With further enhancements such as helmet detection, speed monitoring, and lane discipline analysis, this AI-driven approach can evolve into a comprehensive traffic management solution, supporting the broader vision of intelligent urban infrastructure and smart city development.

Future Scope

- AI-based helmet detection
- Smart lane discipline detection
- Overspeeding detection using sensor fusion
- Fog/rain/night-mode enhancement
- Live dashboard for central monitoring
- Predictive analytics for accident-prone zones
- Multi-camera fusion for 360° junction coverage

References

- YOLOv8 documentation
- OpenCV official docs
- Tesseract OCR documentation
- ByteTrack research paper
- Smart traffic enforcement papers
- Dataset papers for ANPR
- Research on HSV and traffic signal detection