iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: III Month of publication: March 2025

DOI: https://doi.org/10.22214/ijraset.2025.67767

Traffi: Smart Surveillance for Roads

Nirmit Rathod[1], Prathamesh Raut[2], Preet Savliya[3], Sumit Shukla[4]
Department of Artificial Intelligence and Machine Learning, Mumbai University, India

Abstract: Vehicle detection and counting play a crucial role in intelligent transportation systems, traffic monitoring, and urban planning. Traditional methods for vehicle detection often struggle with accuracy and real-time performance, especially in dynamic environments. With the advancement of deep learning, object detection models like YOLOv4 (You Only Look Once) have significantly improved detection speed and accuracy. Coupled with OpenCV, a powerful computer vision library, YOLOv4 enables efficient vehicle detection and tracking in real-world scenarios. In this work, we implement a vehicle detection and counting system using the YOLOv4 deep learning model and OpenCV. The system processes video streams to detect vehicles, classify them, and count their movement across predefined regions. YOLOv4's convolutional neural network architecture allows for high-speed inference, while OpenCV handles image preprocessing, post-processing, and visualization. The model is trained on a dataset of various vehicle types and optimized for real-time performance on both CPU and GPU environments. Our implementation achieves high accuracy in vehicle detection and counting, even in challenging conditions such as occlusions, varying lighting, and heavy traffic. The system demonstrates real-time processing capabilities, making it suitable for smart traffic management applications. By leveraging YOLOv4 and OpenCV, we provide a robust and efficient solution for automated vehicle monitoring, contributing to improved traffic flow analysis and transportation planning.
Keywords: Vehicle Detection, Vehicle Counting, Deep Learning, OpenCv, Traffic Management, Object Detection, Smart Cities

I. INTRODUCTION

The rapid urbanization and growing traffic congestion have necessitated effective traffic monitoring solutions. Traditional methods of vehicle detection, such as magnetic sensors and inductive loops, have limitations in terms of accuracy, cost, and scalability. Deep learning, particularly OpenCv, has demon started promising results in real-time object detection tasks, including vehicle recognition. This paper explores the application of deep learning techniques in vehicle detection and counting, with the goal of improving traffic management and ensuring safer road environments. Deep learning has emerged as a transformative technology in the field of computer vision, providing powerful tools for object detection and recognition. Among the various architectures available, You Only Look Once (YOLO) has gained significant attention due to its remarkable ability to process images in real time while maintaining high accuracy. YOLO v4, the latest iteration of this architecture, incorporates several enhancements that improve detection speed and precision, making it well-suited for the challenging task of vehicle detection in real-world scenarios. By employing advanced techniques such as data augmentation, improved backbone networks, and multi-scale predictions, YOLO v4 is capable of detecting multiple vehicles in diverse conditions, from busy highways to crowded urban streets.

II. LITERATURE REVIEW

1) Artificial Intelligence (AI) Enabled Vehicle Detection and Counting Using Deep Learning • Authors: Mohana, R. Kejriwal, R. H J, and A. Arora • Conference: 2022 International Conference on Computer Communication and Informatics (ICCCI), 2022 Summary: This paper explores the application of Artificial Intelligence (AI) tools in vehicle detection and counting to enhance urban traffic management systems. The authors utilized the YOLOv3 (You Only Look Once version 3) deep learning algorithm for its efficiency and accuracy in detecting various vehicle classes. The study also discusses the role of edge devices in processing data locally, reducing latency and bandwidth usage. Additionally, the paper highlights the potential of AI tools such as Graph Neural Networks (GNN), Long Short-Term Memory (LSTM), and Quantum Convolutional Neural Networks (QCNN) in this domain.
2) Vehicle Detection and Tracking Using YOLOv8 and Deep Learning to Boost Image Processing Quality • Authors: Priyanka Ankireddy, V. Siva Krishna Reddy, and Dr. V. Lokeswara Reddy • Journal: IJFANS International Journal of Food and Nutritional Sciences, 2022 Summary: This paper proposes a system for vehicle detection and tracking utilizing the YOLOv8 architecture combined with transfer learning techniques. The authors aimed to enhance image processing quality and expedite the detection process. The study achieved a 98.48% success rate in recognizing automobiles and implemented Deep SORT as the vehicle tracker to reduce error rates.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue III Mar 2025- Available at www.ijraset.com

3) Vehicle Counting Using Deep Learning • Authors: Jenisha R, Aswin S, and Shishand A • Journal: Journal of Xi'an Shiyou University, Natural Science Edition, ISSN: 1673-064X Summary: This paper presents a vehicle counting system using the YOLOv8 model, a state-of-the-art object detection algorithm. The system aims to accurately detect and count vehicles in real-time, addressing challenges in traffic analysis and management.

III. METHODOLOGY

A. Research Methodology

This research employs a deep learning approach for vehicle detection and counting in traffic video sequences, leveraging the YOLOv4 architecture. YOLO (You Only Look Once) is a state-of-the-art object detection framework known for its real-time processing capabilities. To enhance the model's accuracy and adaptability, pre-trained models such as YOLOv3 were fine-tuned using traffic-specific datasets. The model was optimized for high detection accuracy by adjusting hyperparameters and implementing transfer learning techniques. Additionally, post-processing techniques like Non-Maximum Suppression (NMS) were applied to refine detections by reducing redundant bounding boxes.

B. Data Collection

Traffic video datasets were sourced from publicly available repositories such as the UA-DETRAC dataset, which provides annotated vehicle data across various urban settings. The dataset encompasses different weather conditions, lighting scenarios, and vehicle densities, ensuring diversity and robustness in training. Prior to training, the raw data underwent a preprocessing pipeline that included:

1) Frame Extraction: Converting video sequences into frames at a predefined frame rate.
2) Image Resizing: Standardizing image dimensions to match YOLOv4 input requirements.
3) Data Cleaning: Removing redundant or poor-quality frames to maintain dataset integrity.
4) Augmentation Techniques: Applying transformations such as rotation, flipping, brightness adjustment, and noise injection to improve generalization and mitigate overfitting.
5) Annotation Refinement: Verifying and adjusting ground truth labels to ensure consistency with model expectations.

C. Analytical Techniques

The system's performance was rigorously evaluated using multiple accuracy metrics and statistical analysis tools. Key evaluation metrics included:

1) Mean Average Precision (mAP): A widely used metric for object detection models, calculated by averaging the precision values across different Intersection over Union (IoU) thresholds.
2) Counting Accuracy: The percentage of correctly identified and counted vehicles compared to ground truth annotations.
3) Precision-Recall Analysis: Evaluating the trade-off between precision and recall to determine the optimal threshold for detection confidence.
4) Confusion Matrix: Analyzing True Positives (TP), False Positives (FP), and False Negatives (FN) to measure detection reliability.

IV. SYSTEM ARCHITECTURE

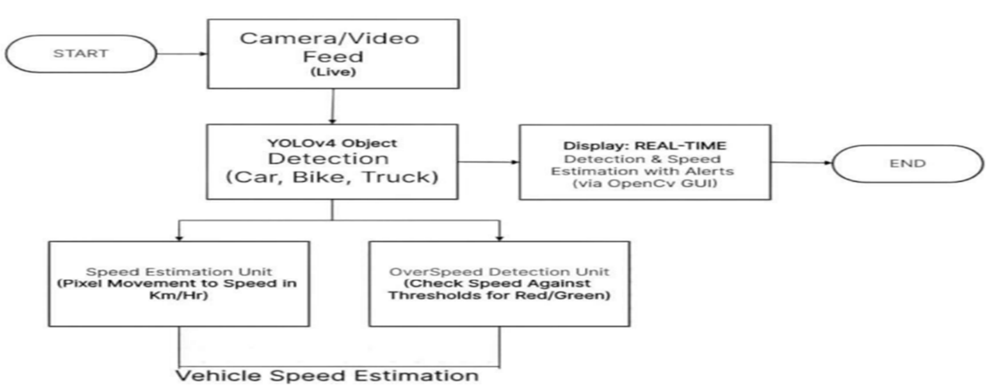


Fig. 4.1 System Architecture

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue III Mar 2025- Available at www.ijraset.com

The provided system architecture for vehicle speed estimation and overspeed detection is a comprehensive solution that leverages advanced computer vision techniques to monitor traffic and enforce speed limits. Here's a more detailed explanation of each component and its role within the system: Camera/Video Feed (Live): • The system's foundation is a continuous stream of video data from a camera. This could be a stationary camera mounted on a roadside pole, a traffic light, or a mobile device capturing footage from a moving vehicle. • The quality and resolution of the video feed are crucial for accurate object detection and speed estimation. Higher-quality video typically leads to better results. YOLOv4 Object Detection: Vehicle Detection and counting Using Deep Learning 18 • YOLOv4 (You Only Look Once) is a state-of-the-art object detection algorithm that is particularly efficient and accurate for real-time applications. • It is trained on a large dataset of images containing various objects, including cars, bikes, and trucks. During inference, the algorithm divides the input image into a grid and predicts bounding boxes and class probabilities for each grid cell. • The YOLOv4 model's ability to detect multiple objects simultaneously makes it well-suited for traffic monitoring scenarios where multiple vehicles might be present in a single frame. Speed Estimation Unit: • This unit is responsible for calculating the speed of each detected vehicle based on its motion within the video frames. • It employs a tracking algorithm to follow the bounding box of a vehicle over time. By measuring the distance traveled by the vehicle's bounding box between consecutive frames and knowing the frame rate and the scale of the image (i.e., the distance represented by a pixel), the unit can estimate the vehicle's speed. • The accuracy of the speed estimation depends on factors such as the camera's calibration, the quality of the video, and the tracking algorithm's robustness. Overspeed Detection Unit: • This unit compares the estimated speed of each vehicle against predefined speed limits. • The speed limits can be set based on road conditions, traffic regulations, and safety considerations. • If a vehicle's speed exceeds the threshold, the unit triggers an alert or signal, such as a red light, a flashing sign, or a notification to a traffic enforcement officer. • The unit can also maintain a record of overseeding vehicles for further analysis and enforcement actions. Display: REAL-TIME Detection & Speed Estimation with Alerts (via OpenCV GUI): • The system provides a visual interface to display the detected objects, their estimated speeds, and any overspeed alerts in real-time. An OpenCV GUI (Graphical User Interface) is often used for this purpose, as it offers a flexible and efficient way to visualize and interact with video data. • The display can show the detected vehicles with bounding boxes, their estimated speeds, and any relevant alerts or indicators. • This visual information is valuable for operators to monitor traffic conditions, identify potential hazards, and take appropriate actions. SORT stands for Simple Online and Realtime Tracking. **Core Idea:** SORT is a lightweight tracking algorithm that can track multiple objects (like vehicles) in real time using: Kalman Filters – Predict where the object will be in the next frame. Hungarian Algorithm – Match detected objects with existing tracks (IDs) efficiently.

Formula Used in Integration:

$$\text{Speed}_{km/h} = \frac{\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}\times\text{Pixel-to-Meter Ratio}\times 3.6}{t_2-t_1}$$

Speed Formula (in km/h):

$$\boxed{\text{Speed}_{km/h} = \frac{\text{Pixel Distance}\times\text{Pixel-to-Meter Ratio}\times 3.6}{\Delta t}}$$

Speed = Distance between two positions of the same tracked vehicle per second, converted to km/h.
x1, y1 = position at time t1
x2, y2 = position at time t2
Where:
Pixel Distance = distance between the same vehicle’s center across two frames (in pixels)
Pixel-to-Meter Ratio = estimated real-world distance per pixel (e.g., 0.05 m/pixel).
Δt = time difference between frames (in seconds).
3.6 = conversion factor from m/s to km/h.

$$\text{Speed}_{km/h} = \frac{100\times 0.05\times 3.6}{1} = 18\,\text{km/h}$$

Example: If:
Pixel Distance = 100 px
Ratio = 0.05 m/px
Δt = 1 second

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue III Mar 2025- Available at www.ijraset.com

V. EXPERIMENTATION & RESULTS

Figure 5.1 Vehicle Detection and Classification

Figure 5.1 illustrates the outcome of object detection using a pre-trained deep learning model. The detected objects are enclosed within bounding boxes, which visually highlight each identified vehicle or object. Alongside the bounding boxes, the system also provides a confidence score, representing the probability that the detected object belongs to a particular category. Additionally, a mask overlay is applied, enhancing the segmentation process by distinguishing individual objects from the background. This step is crucial for accurate detection and tracking of vehicles in real-time scenarios.

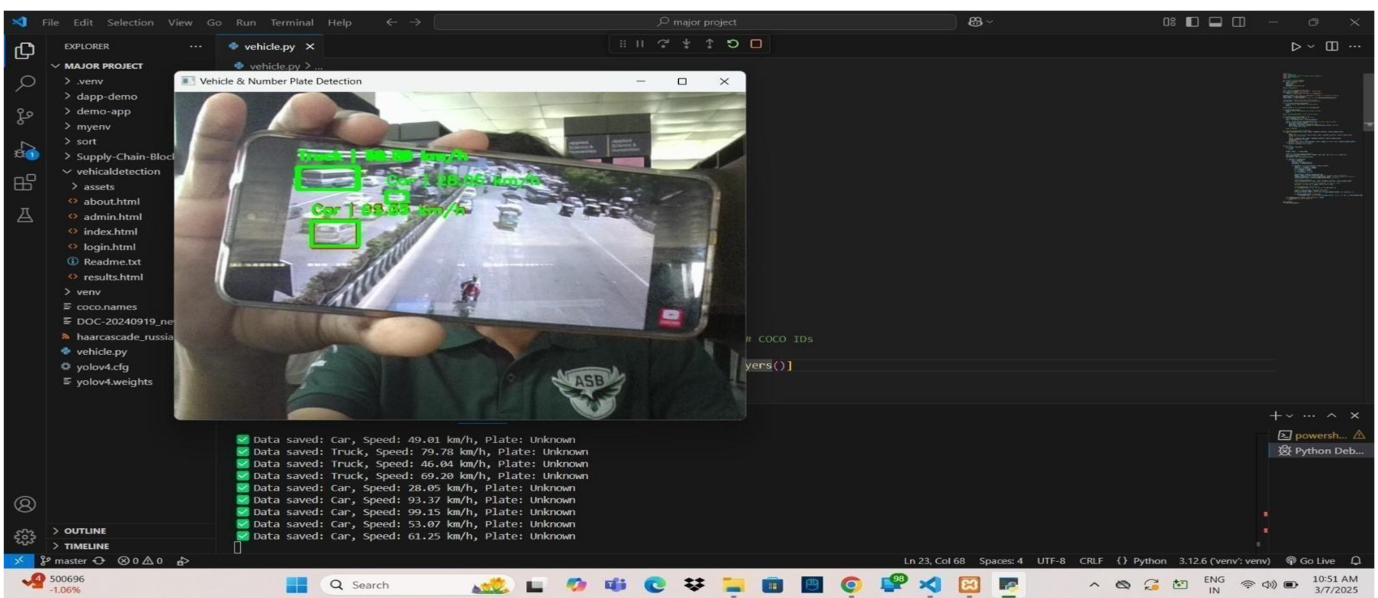


Figure 5.2 Vehicle Detection and Classification from Video

Before detecting objects in a live or recorded video, the system must extract individual frames from the video feed. Figure 5.2 demonstrates the results of the detection process on one such extracted frame. Each frame is processed through the object detection algorithm, and vehicles are identified with bounding boxes, confidence scores, and unique identifiers if tracking is involved. The ability to analyze video feeds frame-by-frame enables continuous monitoring of road traffic, making it useful for applications such as vehicle counting, speed detection, and congestion analysis.

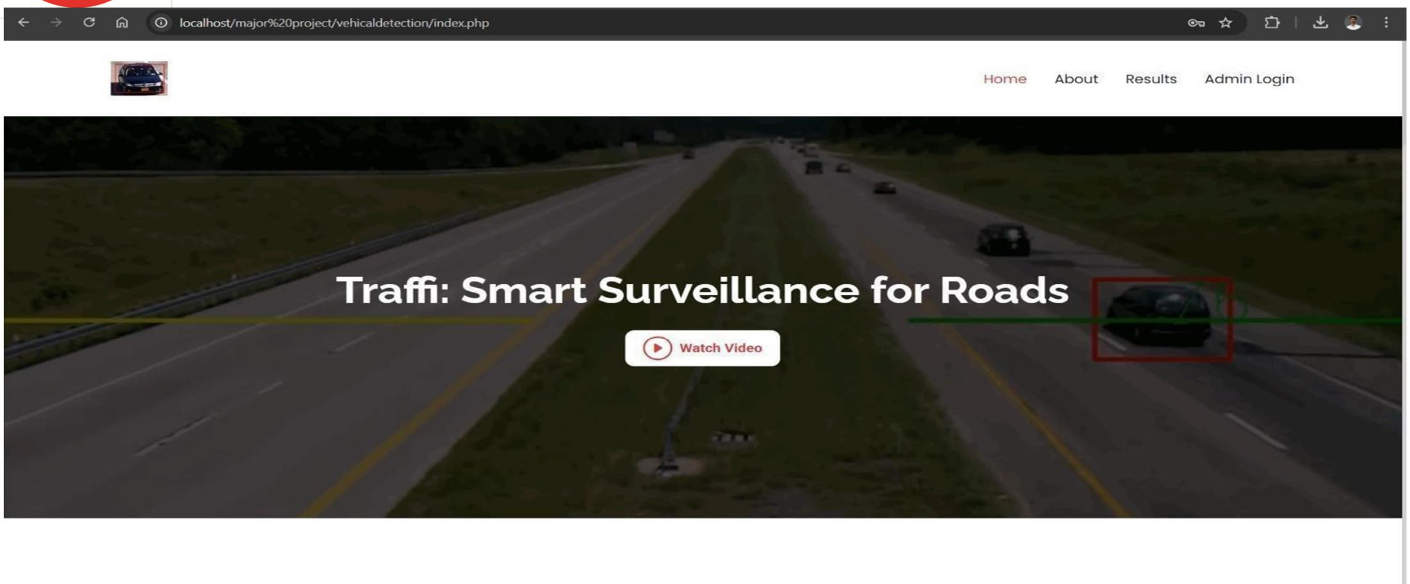


Figure 5.3 Homepage of Our Website

The homepage of the "Traffi: Smart Surveillance for Roads" system serves as the initial user interface, providing an overview of the platform's core functionalities. As depicted in Figure 5.3, the homepage features a clean and modern design, prominently displaying a "Watch Video" button, encouraging user engagement. This button likely provides an introductory demonstration of how the system functions in real-world scenarios. Additionally, the homepage contains navigation links that suggest access to other key components of the system, such as real-time traffic monitoring, historical data analysis, and administrative controls. Based on the location information inferred from the system's usage, it appears that the project has a regional focus in Pune, Maharashtra, India, making it particularly relevant for local traffic authorities and law enforcement agencies.

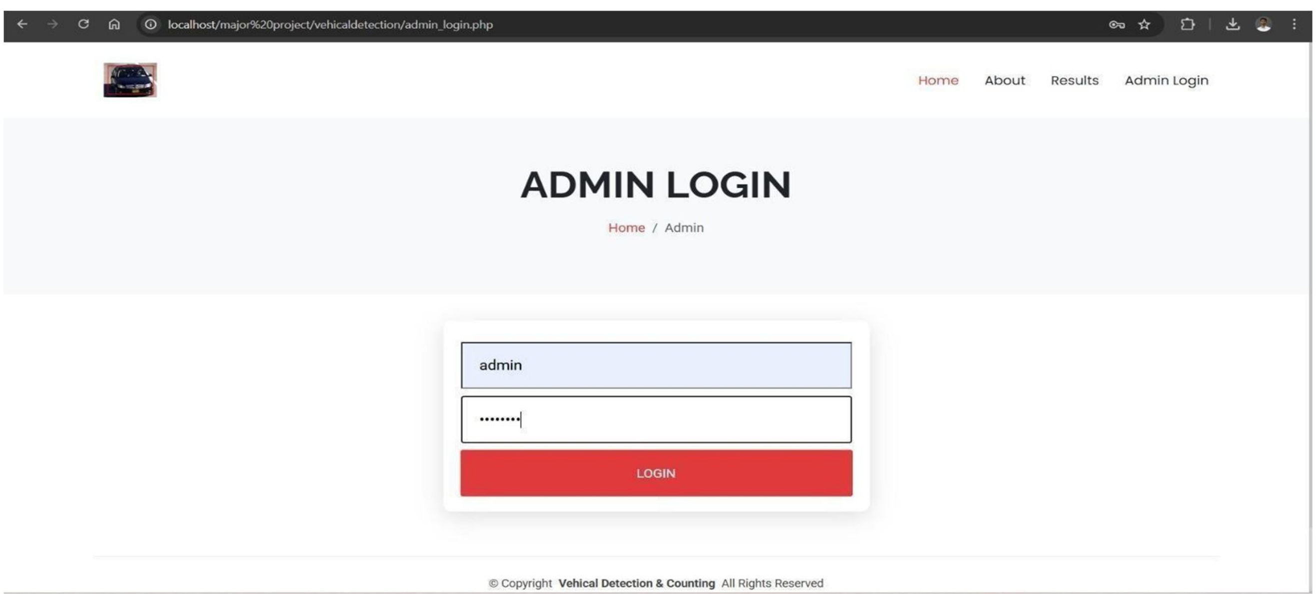
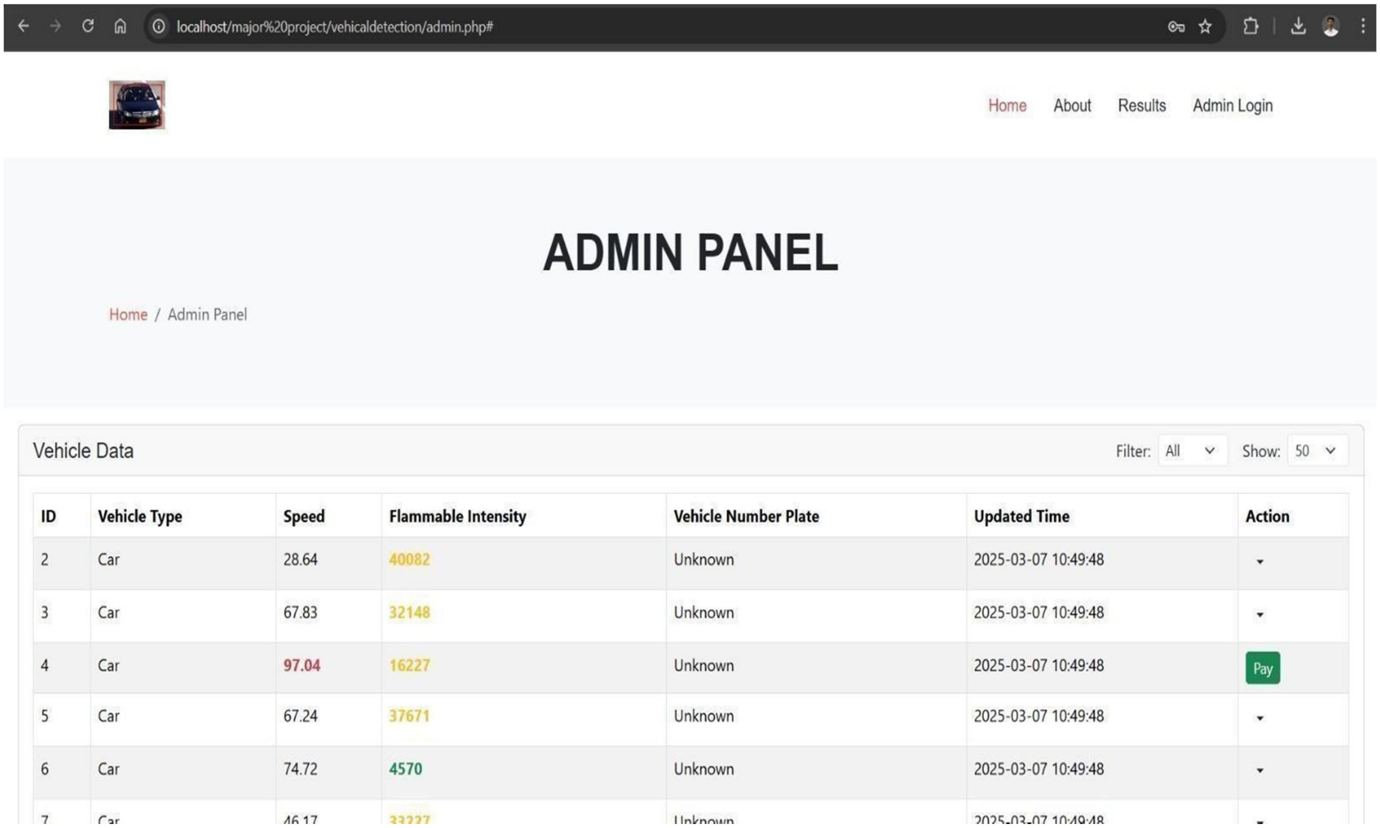


Figure 5.4 Admin Login Page

Figure 4.4 displays the admin login page of the website. This page serves as a secure gateway for administrators to access and manage the system. The interface follows a minimalistic yet functional design, prioritizing security by requiring authentication credentials (e.g., username and password). Given that the system deals with sensitive traffic data, the security measures in place are crucial to prevent unauthorized access. The location reference in the system suggests that administrators managing traffic in a specific region (possibly Pune, Maharashtra) can log in to perform tasks such as monitoring vehicle movements, generating reports, and enforcing speed regulations.



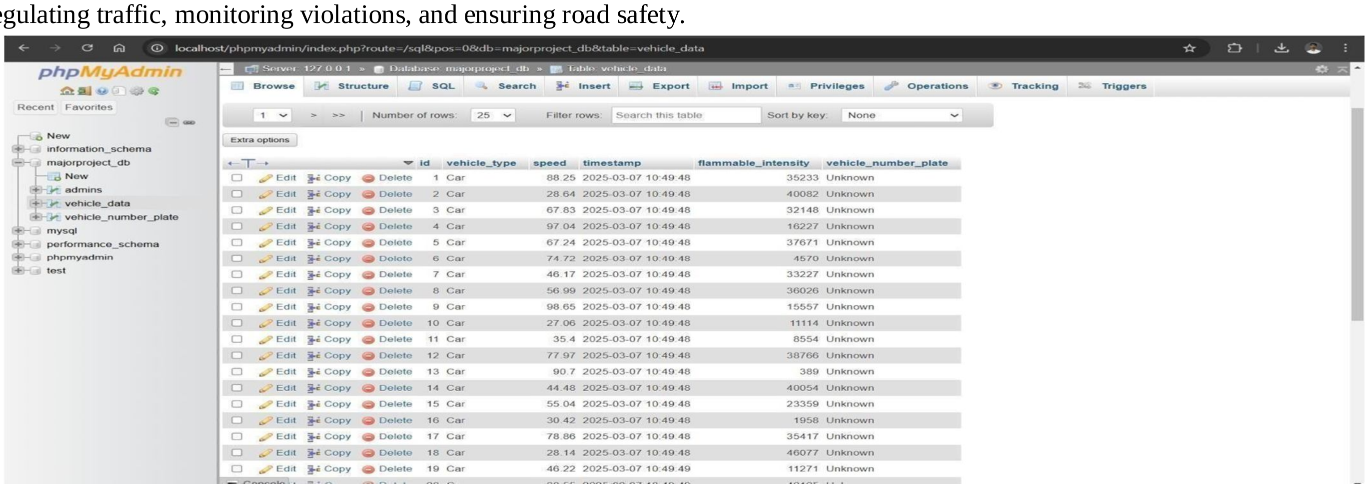
ID	Vehicle Type	Speed	Flammable Intensity	Vehicle Number Plate	Updated Time	Action
2	Car	28.64	40082	Unknown	2025-03-07 10:49:48	
3	Car	67.83	32148	Unknown	2025-03-07 10:49:48	
4	Car	97.04	16227	Unknown	2025-03-07 10:49:48	Pay
5	Car	67.24	37671	Unknown	2025-03-07 10:49:48	
6	Car	74.72	4570	Unknown	2025-03-07 10:49:48	

Figure 5.5 Admin Panel

Figure 5.5 presents the admin panel, which acts as the central hub for managing vehicle-related data collected by the "Traffi: Smart Surveillance for Roads" system. The panel provides a data-centric interface where administrators can filter, analyze, and manage vehicle detection results. Key features of this admin panel include:

- Data filtering and sorting options for easy access to specific vehicle records.
- A "Pay" button, which suggests the integration of a possible fine payment system for traffic violations.
- A "Flammable Intensity" column, indicating that the system may have advanced capabilities beyond traffic monitoring, potentially identifying hazardous vehicles carrying flammable materials.
- The inclusion of location-based information, reinforcing the system's targeted deployment in a specific geographical region.

This comprehensive vehicle management system provides law enforcement and traffic authorities with actionable insights for regulating traffic, monitoring violations, and ensuring road safety.



id	vehicle_type	speed	timestamp	flammable_intensity	vehicle_number_plate
1	Car	88.25	2025-03-07 10:49:48	35233	Unknown
2	Car	28.64	2025-03-07 10:49:48	40082	Unknown
3	Car	67.83	2025-03-07 10:49:48	32148	Unknown
4	Car	97.04	2025-03-07 10:49:48	16227	Unknown
5	Car	67.24	2025-03-07 10:49:48	37671	Unknown
6	Car	74.72	2025-03-07 10:49:48	4570	Unknown
7	Car	46.17	2025-03-07 10:49:48	33227	Unknown
8	Car	56.99	2025-03-07 10:49:48	36026	Unknown
9	Car	98.65	2025-03-07 10:49:48	15557	Unknown
10	Car	27.06	2025-03-07 10:49:48	11114	Unknown
11	Car	35.4	2025-03-07 10:49:48	8554	Unknown
12	Car	77.97	2025-03-07 10:49:48	38766	Unknown
13	Car	90.7	2025-03-07 10:49:48	389	Unknown
14	Car	44.48	2025-03-07 10:49:48	40054	Unknown
15	Car	55.04	2025-03-07 10:49:48	23359	Unknown
16	Car	30.42	2025-03-07 10:49:48	1958	Unknown
17	Car	78.86	2025-03-07 10:49:48	35417	Unknown
18	Car	28.14	2025-03-07 10:49:48	46077	Unknown
19	Car	46.22	2025-03-07 10:49:49	11271	Unknown

Figure 5.6 Localhost PHP my admin

Figure 5.6 offers a behind-the-scenes look at the database structure of the "Traffi: Smart Surveillance for Roads" system using phpMyAdmin. This tool allows for database management, query execution, and troubleshooting. The displayed table structure provides:

- A clear representation of the vehicle data schema, including columns such as vehicle ID, detection timestamp, speed, and violation status.
- Confirmation that the database supports the admin panel's real-time data updates and queries.
- A direct interface for database administrators to manage stored records, optimize performance, and ensure data integrity. Since phpMyAdmin is widely used for database administration, its integration into the system suggests a well-structured and maintainable backend, essential for handling large-scale traffic data efficiently.

VI. CONCLUSION

The Vehicle Detection and Counting project using YOLOv4 and OpenCV successfully demonstrated real-time detection and counting of vehicles with high accuracy. The YOLOv4 model was effective in identifying different vehicle types such as cars, trucks, buses, and motorcycles, and the OpenCV and HAAR Cascade enabled fast inference. This research demonstrates the effectiveness of deep learning models, specifically YOLOv4 and OPENCV, in the detection and counting of vehicles in real- time traffic monitoring systems. The results confirm that deep learning methods can provide high accuracy and efficiency in complex environments. As traffic management systems evolve, the integration of AI and machine learning will play a pivotal role in the development of smarter, safer, and more efficient urban infrastructures.

VII. ACKNOWLEDGEMENT

We take this opportunity to express our deep sense of gratitude to our project guide and project coordinator Mr. Sandeep Dubey for his continuous guidance and encouragement throughout the duration of our project work. It is especially because of his experience and wonderful knowledge; we can fulfill the requirement of completing the project within the stipulated time. We would also like to thank Mrs. Poonam Thakre, the head of Department of Artificial Intelligence and Machine Learning department, for his encouragement and support. We would also like to thank our principal Dr. J.B. Patil and the management of Universal College of Engineering, Vasai, Mumbai for providing us all the facilities and the work friendly environment. We acknowledge with thanks, the assistance provided by departmental staff, library and lab attendants for their help.

REFERENCES

[1] Mohana, R. Kejriwal, R. H J and A. Arora, “Artificial Intelligence (AI) Enabled Vehicle Detection and counting Using Deep Learning,” 2022 International Conference on Computer Communication and Informatics (ICCCI), 2022.

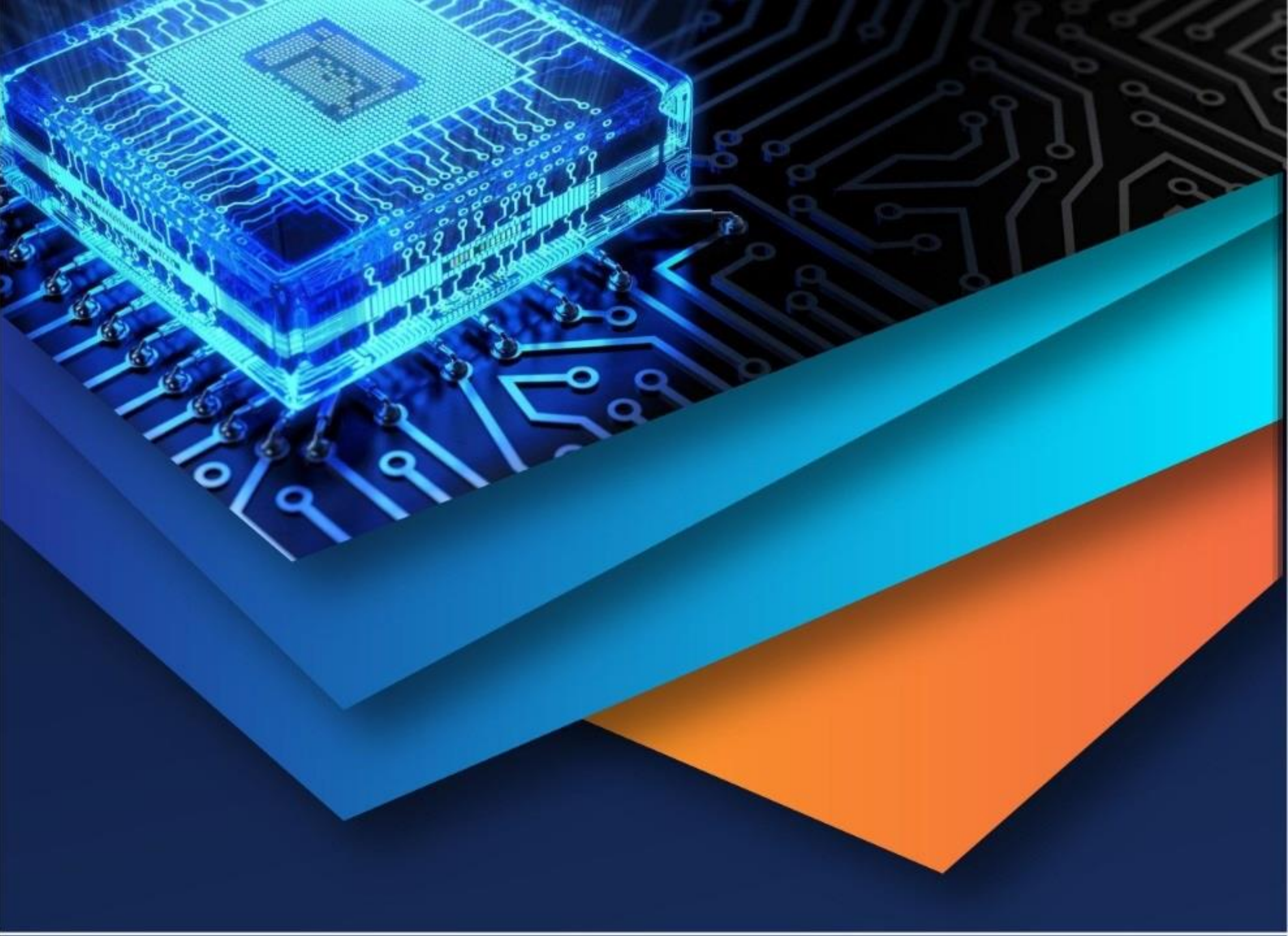
[2] Priyanka Ankireddy, V. Siva Krishna Reddy, Dr.V. Lokeswara Reddy “Vehicle Detection and Tracking Using YOLOv8 and Deep Learning to Boost Image Processing Quality” 2022 IJFANS INTERNATIONAL JOURNAL OF FOOD AND NUTRITIONAL SCIENCES, ISSN PRINT 2319 1775 Online 2320 7876.

[3] Jenisha R, Aswin S, Shishand A, “VEHICLE COUNTING USING DEEP LEARNING”, Journal of Xi’an Shiyou University, Natural Science Edition ISSN: 1673-064X VEHICLE DETECTION AND COUNTING OF A VEHICLE USING OPENCV Karthik Srivathsa D S*1, Dr. Kamalraj

[4] R21MCA, Jain University, Bengaluru, Karnataka, India. 2Guide, MCA, Jain University, Bengaluru, Karnataka, India.

[5] VEHICLE DETECTION AND COUNT USING OPENCV Vijay D Chaudhari1, Dr. Anil J Patil2 1Research scholar, Kavayitri Bahinabai Chaudhari North Maharashtra University, Jalgaon. 2Principal, Samarth College of Engineering, Belhe, Tam Junner. Pune.

[6] Vehicle Speed Estimation using Object Detection for Intelligent Traffic Management Atharva Hiwarekar, Swaroop Chavhan, Onkar Deshpande, Vedant Joshi Pune Institute of Computer Technology.



10.22214/IJRASET

INDEX COPERNICUS

45.98

ISRA JIF

IMPACT FACTOR: 7.129

SJIFactor

TOGETHER WE REACH THE GOAL

IMPACT FACTOR: 7.429

ISI

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY