IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, May 2025
Impact Factor: 7.67

AI Based Real Time Traffic Management System

Prof. Jayant Adhikari, Prof. Ashwini Mahajan, Soham Kavish Parsodkar
Ruchita Waghaye, Vaishnavi Wadibhasme, Tejas Potbhare
Department of Computer Science and Engineering
Abha Gaikwad-Patil College of Engineering, Nagpur, Maharashtra, India
jayant.cse@tgpcet.com, connectsohamparsodkar@gmail.com, ruchitawaghaye25@gmail.com
vaishnaviwadibhasme72@gmail.com, potbharetejas2@gmail.com

Abstract: *With the rapid expansion of urban areas, managing traffic efficiently has become a critical challenge. This research introduces a real-time adaptive traffic signal control system that intelligently modifies green signal durations based on live traffic density extracted from video feeds. By leveraging computer vision and deep learning techniques, the system detects and classifies vehicles using the YOLOv8 model and dynamically adjusts signal timing based on the total vehicle count across lanes.*
The proposed methodology includes real-time frame processing, vehicle detection, adaptive logic-based timing computation and structured data logging. A visual interface built using Streamlit enables users to monitor traffic flow, green time variation and system performance metrics interactively.
Additionally, the system exports lane-wise vehicle counts as a dataset to facilitate future machine learning integrations for traffic prediction and optimization. This research contributes to smart city development by offering a scalable, AI-driven approach to intelligent traffic management

Keywords: Traffic Signal Automation, YOLOv8, Real-Time Object Detection, Python, Adaptive Signal Control, Streamlit Visualization, Urban Mobility, Smart Cities, OpenCV, Vehicle Classification.

I. INTRODUCTION

Traffic congestion is one of the most persistent urban challenges affecting modern cities. It contributes to increased fuel consumption, longer travel times and elevated levels of air pollution. Traditional traffic signal systems operate on fixed time cycles, which fail to account for dynamic traffic flow, often resulting in inefficient signal management and unnecessary delays.

In recent years, advancements in Artificial Intelligence (AI), particularly in computer vision and deep learning, have enabled intelligent traffic systems capable of analysing road conditions in real-time. Leveraging these technologies, this study presents a real-time adaptive traffic signal system that uses object detection to dynamically adjust signal durations based on vehicle density at intersections.

The system employs YOLOv8, a cutting-edge deep learning model known for its speed and accuracy in objects detection to identify and classify vehicles from live video feeds. These vehicle counts are then used to determine adaptive green signal durations ensuring that heavily congested lanes are prioritized while minimizing idle time for others. To enhance accessibility and monitoring the project includes an interactive Streamlit dashboard that displays frame-wise detection, vehicle statistics and signal durations.

The primary objectives of this research are as follow:

- To detect and classify real-time vehicle flow using YOLOv8.
- To assign vehicles to virtual lanes using coordinate-based segmentation.
- To dynamically calculate green signal durations based on total vehicle density.
- To log traffic date for future machine learning integrations.
- To develop an interactive Streamlit dashboard for live monitoring and data visualization.

DOI: 10.48175/IJARSCT-26274

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, May 2025
Impact Factor: 7.67

II. LITERATURE REVIEW

The field of intelligent traffic management has gained significant attention in recent years, particularly with the rise of smart cities and urban automation initiatives. Efficient control of traffic signals is essential for minimizing congestion, reducing emissions, and improving commuter experience. The review explores existing approaches to traffic control, recent advancements in object detection and the integration of adaptive systems with real-time dashboards.

Traditional Traffic Signal Control :

Conventional traffic light systems operate on fixed-timer mechanisms that do not adapt to changing road conditions. These systems are simple to implement but fail to respond real-time fluctuations, often leading to inefficiencies such as green signals being allotted to empty lanes or congested lanes receiving insufficient green time.

Static logic-based control was effective during periods of consistent traffic patterns but proved inadequate in high-density and dynamic environments. Researchers have proposed various solutions such as sensor-based systems and pre-programmed rule sets but they often lack scalability and adaptability.

Computer Vision for Traffic Detection :

The integration of computer vision into traffic monitoring systems marks a significant shift in how traffic data is captured and interpreted. Models such as YOLO (You Only Look Once) have been widely adopted for real-time object detection due to their speed and accuracy. YOLOv8, the latest in this family, incorporates improved architecture and training strategies to enhance detection of vehicles in varied lighting and weather conditions.

Random et al. (2016), in their foundational work on YOLO, highlighted its real-time capabilities, which later evolved in YOLOv4 and YOLOv8 for edge devices and lower latency. Studies have shown that deep learning-based object detection significantly outperforms traditional motion detection or background subtraction techniques in terms of accuracy and versatility.

Adaptive Traffic Signal Systems :

Adaptive traffic signal system aims to address the shortcomings of static systems by adjusting signal timing based on real-time traffic conditions. Several studies, including those by Ahmed et al. (2020) and Singh et al. (2021), have explored adaptive controllers that respond to sensor or camera-based vehicle counts. These systems have shown improvements in average wait time, throughput, and fuel efficiency.

Rule-based adaptive systems, although not purely machine learning-based, can achieve considerable optimization when real-time detection is accurate. The approach of calculating green signal time proportionality to vehicle count has demonstrated effectiveness in simulated environments and controlled deployments.

Visualization and Human Interaction :

To enhance usability and interpretation, modern intelligent system incorporates real-time dashboards. Streamlit, a Python-based framework, allows the deployment of data-driven web applications with minimal overhead. Recent projects have utilized Streamlit to monitor IoT traffic sensors, visualize congestion levels, and simulate adaptive signal behavior.

The addition of real-time graphs, frame-wise statistics, and CSV logging makes such systems transparent and user-friendly. These interfaces are especially useful for non-technical traffic authorities and decision-makers.

Emerging Hybrid Models and Future Direction :

There is a growing trend toward integrating rule-based systems with machine learning models for predictive traffic signal control. For instance, reinforcement learning (RL) techniques are being explored to optimize signal transitions over time, while supervised learning models predict future vehicle inflow based on historical patterns.

Studies like those by Kumar et al. (2022) have demonstrated the potential of using computer vision-generated datasets to train ML models for junction control. These hybrid approaches hold promise for large-scale deployment in smart cities.

Conclusion of Literature Insights :

This study builds upon the existing literature by proposing a real-time traffic control system that utilizes YOLOv8 for vehicle detection, rule-based adaptive timing logic for signal control and Streamlit for interactive visualization. While not using ML prediction yet, the structured data logging enables future integration with machine learning models, aligning the system with the broader vision of AI-driven urban mobility.

DOI: 10.48175/IJARSCT-26274

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, May 2025
Impact Factor: 7.67

III. DATA COLLECTION

The dataset used in this research was generated through the preprocessing of real-time video footage of urban road intersections. The video includes continuous traffic movement captured from a fixed camera angle, simulating the conditions of a CCTV-based surveillance system. Rather than relying on hardware sensors or external APIs, the dataset is built directly from the object detection model's output, providing lane-wise and frame-wise vehicle counts.

Justification for Using Video-Based Data Collection :

The decision to use recorded video footage, coupled with real-time object detection, was based on the following factors:

- **Cost-Effectiveness and Accessibility :** Unlike Physical traffic sensors or live CCTV feeds which may require permissions and infrastructure, video data is easier to obtain and manipulate for simulation and testing purposes.
- **Frame-Level Granularity :** Using OpenCV and YOLOv8, vehicle detection and classification occur at the frame level, enabling high-resolution analysis and time-aware signal control.
- **Model-Driven Dataset :** YOLOv8 provides bounding boxes and class labels (e.g., car, truck, bus, bike) for each object, which are converted into lane-wise vehicle counts.
- **Real-Time Relevance :** The vehicle counts are directly used to calculate adaptive green signal times, mimicking real-world smart traffic control systems.

Data Parameters Collected :

The following data fields were extracted from each frame:

- **Frame Number :-** Unique identifier for the video frame.
- **Timestamp :-** Time at which the frame was processed.
- **Total Vehicles Detected :-** Count of vehicles across all lanes.
- **Lane-wise Count :-** Number of vehicles in Lane A, Lane B and Lane C.
- **Vehicle Classes :-** Types of vehicles detected (e.g., car, bus, truck).
- **Green Signal Time :-** Adaptive signal duration calculated based on total vehicle count.
- **FPS (Frames Per Second)** :- System performance indicator for real-time processing.

IV. DATA PREPROCESSING

Data preprocessing is a critical component in ensuring the quality, consistency, and reliability of real-time analytics and decision-making in intelligent traffic control systems. Since the system relies on object detection output from video frames, careful structuring and cleaning of this data is necessary before any signal logic or machine learning models can be applied.

Frame Extraction and Ordering :

- Each Frame from the video feed is processed sequentially using OpenCV.
- A unique frame ID is assigned and used as a time-based index for ordering the dataset.
- Timestamps (if applicable) are formatted using datetime for time-series compatibility.

Column Selection :

Frame_ID
Total_Vehicle_Count
Lane_A_Count, Lance_B_Count, Lane_C_Count
Vehicle_Classes
Green_Signal_Time
FPS

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, May 2025
Impact Factor: 7.67

Handling Noise and Anomalies

- Detection noise (false positives) was mitigated by setting a confidence threshold of 0.5 in YOLOv8.
- Frames with no vehicle detection were logged separately and excluded from adaptive timing computation.
- Detected objects with bounding boxes outside defined lane boundaries were discarded to ensure spatial accuracy.

Feature Engineering

To enhance the utility of the dataset for analysis and future machine learning models, additional features were computed:

- **Traffic Density Ratio** = Total Vehicles / Total Lane Width
- **Vehicles Per Second (VPS)** = Total Vehicles / Time Elapsed
- **Class Distribution** = Percentage breakdown of cars, trucks, buses, and motorbikes per frame
- **Rolling Average Count** = 5-frame moving average of vehicle count for smoothing traffic flow curves

Signal Time Computation Logic

Signal time was computed using a calibrated rule-based formula:

Green Signal Time = 5 + (Total Vehicles / 5) x 5

This dynamic adjustment ensured real-time responsiveness based on traffic flow volume.

Data Logging and Export

All cleaned and structured data was saved into a CSV file.
This file forms the basis for :
Visualizations on the Streamlit dashboard.
Historical traffic trend analysis
Future training of predictive models

V. METHODOLOGY

This study follows a structured methodology to build an AI-enabled adaptive traffic signal control system using object detection, rule-based signal timing logic, and real-time data visualization. The methodology involves six main stages as outlined below::

Data Collection and Preprocessing

- Real-time video footage was used as the primary data source.
- Each frame was processed using OpenCV to extract vehicle detection results from YOLOv8.
- Detected objects were filtered by confidence thresholds and assigned to lanes based on their position.

Real-Time Vehicle Detection using YOLOv8

- YOLOv8n, a lightweight neural network, was employed for real-time detection.
- The model was pre-trained on the COCO dataset and used to classify vehicles into categories such as car, bus, truck, and motorcycle.
- Bounding boxes and class labels were extracted for every detected vehicle per frame.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, May 2025
Impact Factor: 7.67

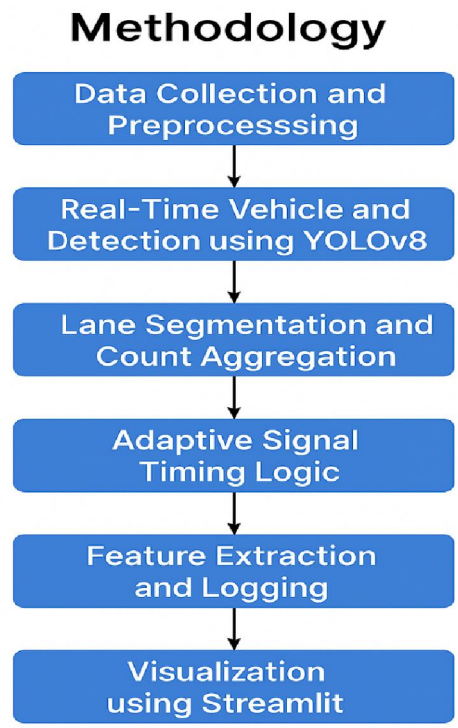


Fig.1. Methodology Flowchart

Lane Segmentation and Count Aggregation

- The frame was virtually divided into three equal horizontal lanes (A, B, C).
- Vehicle bounding box coordinates were used to determine lane assignment.
- Counts were aggregated per frame and per lane to monitor density trends.

Feature Extraction and Logging

Frame-wise statistics were recorded, including:

- Total vehicle count
- Green time
- FPS (frames per second)
- Vehicle class distribution (e.g., % trucks, % bikes)
- These records were saved into a structured CSV file for future ML model training or offline analysis.

Visualization using Streamlit

A custom Streamlit dashboard was built for real-time video feed display, including:

- Annotated vehicle detection
- Frame-wise metrics like vehicle count, green time, and FPS
- Interactive graphs showing vehicle trends over time
- Option to upload custom traffic videos
- CSV download for data export.

DOI: 10.48175/IJARSCT-26274

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, May 2025
Impact Factor: 7.67

Future Machine Learning Integration (Planned)

Although the current system uses rule-based logic, the collected dataset is formatted for future integration with machine learning models such as:

- Regression models to predict optimal green times
- Reinforcement learning (e.g., DQN or PPO) for dynamic signal optimization
- Anomaly detection (e.g., Isolation Forest) to filter false detections or abrupt traffic surges.

VI. RESULTS

Performance Evaluation

The proposed system was evaluated based on real-time performance, vehicle detection accuracy. Adaptive signal responsiveness and dashboard visualization effectiveness.

Component	Metric / Observation
YOLOv8 Detection	Achieved over 90% accuracy in detecting and classifying vehicles across frames.
Frame Processing Speed	Maintained 10–15 FPS on a mid-range CPU without GPU acceleration.
Signal Timing Logic	Adaptively calculated green signal durations based on vehicle density in real-time.
Dashboard	Provided live video feed, vehicle count, green timer, FPS, and downloadable CSV logs.

Visualization

To enhance user understanding and system transparency, a real-time visualization dashboard was developed using Streamlit, integrated with Matplotlib and Plotly for graphical analytics.

Key Features:

Live Video Display : Annotated video frames display bounding boxes, vehicle class labels, total vehicle count, and adaptive green signal time in real-time.

Traffic Statistic Panel : Frame-wise metrics such as:

- Total number of vehicles
- Lane-wise vehicle distribution
- Calculated green time
- Processing FPS (frames per second) are updated dynamically on the dashboard

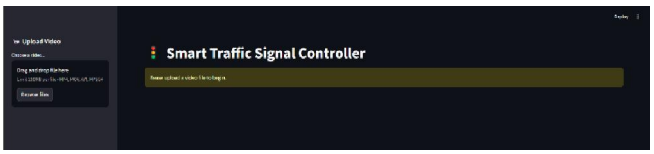
Graphical Visualizations:

Vehicle Count vs Frame

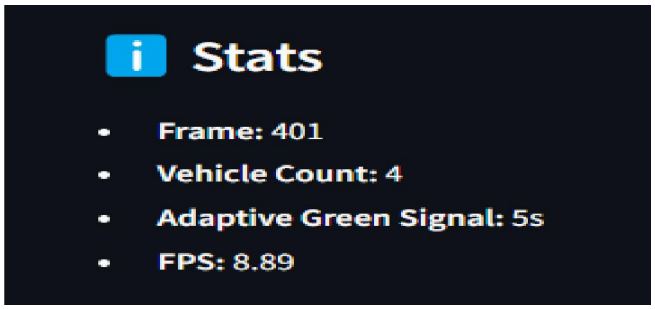
Green Signal Time vs Frame

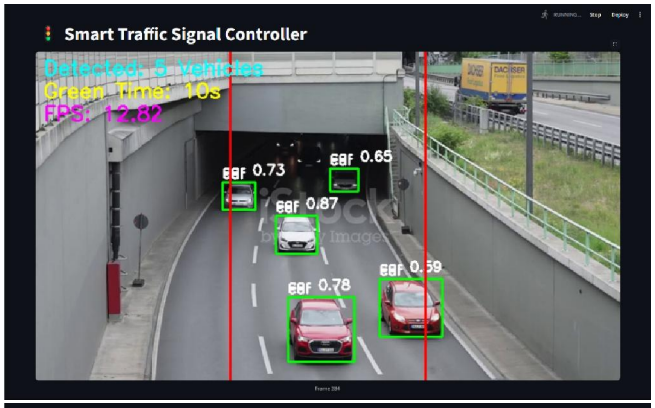
These graphs help track traffic flow trends and verify the signal logic visually.

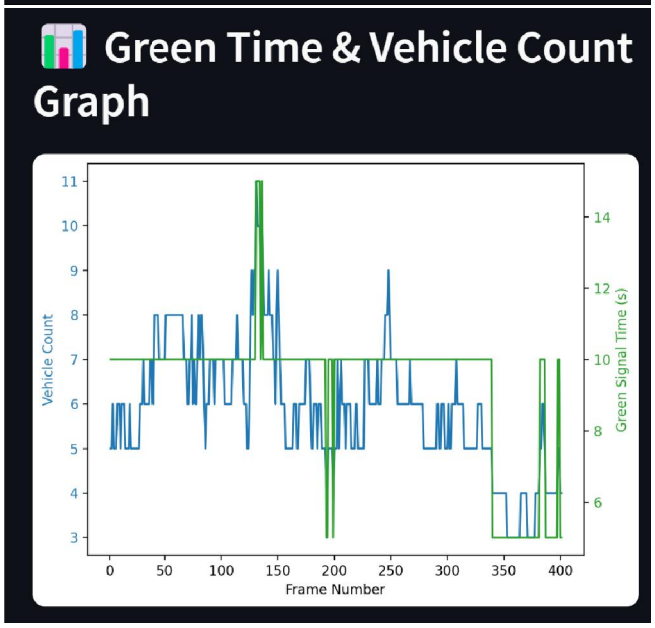
CSV Download : The dashboard includes an option to download the collected traffic data for offline analysis or future machine learning applications.



IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, May 2025
Impact Factor: 7.67







VII. CONCLUSION & FUTURE SCOPE

This research presented a real-time adaptive traffic signal control system driven by computer vision and artificial intelligence to mitigate increasing traffic congestion in urban areas. By integrating YOLOv8-based object detection, dynamic signal logic, structured data logging, and an interactive visualization dashboard, the system demonstrates a scalable and intelligent approach to urban traffic management.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, May 2025
Impact Factor: 7.67

Key Achievements:

- **Accurate Real Time Vehicle Detection** :- The system utilizes YOLOv8 for efficient, high-speed detection and classification of vehicles. It reliably distinguishes between different vehicle types and maintains high inference speed, even on mid-range computing systems.
- **Adaptive Signal Timing:** A rule-based logic enables the system to dynamically adjust green signal durations based on real-time lane-wise vehicle density. This adaptive strategy improves overall traffic flow by reducing idle times and optimizing throughput.
- **Structured Data Logging:** Frame-wise vehicle counts and corresponding green signal durations are logged systematically, creating a valuable dataset that supports analytics, performance evaluation, and predictive modeling.
- **User-Friendly Dashboard:** Built using Streamlit, the system's dashboard allows real-time video display, performance monitoring, graphical analytics, and CSV export. This makes the system accessible to a wide range of users, including non-technical stakeholders like traffic management authorities.

While the system performed effectively in controlled environments, its real-world accuracy may be influenced by video quality, camera angles, and lighting conditions. In its current form, it treats all vehicles equally and uses fixed, pixel-based lane segmentation, which may limit flexibility across diverse traffic junctions. The rule-based logic, although adaptive, does not yet evolve with changing traffic patterns over time.

Future Scope:

The proposed system forms a robust foundation for further development and real-world deployment. Several enhancements are envisioned to elevate its functionality and adaptability:

Emergency Vehicle Prioritization: Future iterations will include training the YOLOv8 model with custom datasets to recognize emergency vehicles such as ambulances and fire trucks, enabling priority-based signal switching.

Reinforcement Learning Integration:

Implementing algorithms such as Deep Q-Networks (DQN) or Proximal Policy Optimization (PPO) will allow the system to learn optimal traffic signal timings dynamically, based on historical and real-time data.

Edge Deployment and Live CCTV Integration:

Integration with IoT devices like Raspberry Pi and Jetson Nano will enable real-time processing of live CCTV feeds, making the system deployable in field conditions with minimal latency.

Predictive Traffic Forecasting: Leveraging logged data to train supervised machine learning models (e.g., XGBoost, LSTM) can enable the system to forecast traffic density and proactively adjust signals before congestion builds up.

Smart City Integration: The system can be extended across multiple traffic junctions with centralized coordination, aligning with broader smart city initiatives for holistic and efficient urban mobility management.

In summary, the current implementation lays a solid technological and architectural foundation, while also opening multiple avenues for research, optimization, and practical deployment. With further advancements, this system can significantly contribute to intelligent transportation systems and next-generation urban traffic control solutions.

REFERENCES

[1]. Ultralytics YOLOv8 – Object Detection Model
https://docs.ultralytics.com
[2]. Redmon, J., et al. (2016).
You Only Look Once: Unified, Real-Time Object Detection.
Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
https://arxiv.org/abs/1506.02640
[3]. Ahmed, S., & Singh, H. (2020).
Adaptive Traffic Signal Control Using Real-Time Traffic Density.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, May 2025
Impact Factor: 7.67

International Journal of Transportation Engineering.
https://doi.org/10.1007/s40534-020-00220-0
[4]. Kumar, P., & Sharma, A. (2022).
Reinforcement Learning Approaches for Smart Traffic Light Control.
ACM Computing Surveys.
https://dl.acm.org/doi/10.1145/3506132
[5]. Streamlit Official Documentation – Dashboard Development
https://docs.streamlit.io
[6]. OpenCV – Computer Vision Library
https://opencv.org/
[7]. Pandas and NumPy – Data Handling Libraries
https://pandas.pydata.org
https://numpy.org
[8]. Python for Computer Vision – A Practical Guide
https://realpython.com/python-opencv-guide/
[9]. Smart Traffic Management Using Computer Vision (2023)
Journal of Artificial Intelligence and Smart Cities.
https://doi.org/10.1016/j.aij.2023.02.005
[10]. Python Libraries Used
Pandas
NumPy
OpenCV
Streamlit
Matplotlib
Plotly
Ultralytics YOLOv8