

Intelligent Traffic Signal Management using Real-Time Data and Image Processing

Venkatesan B¹, Vishnu Priya V², Keerthanashree T³, Suryah Prakash M R⁴, Vikash A G K⁵ and Yogeshwaran P⁶

^{1,3-6}Department of Electronics and Instrumentation Engineering, Kongu Engineering College, Erode, India

Emails: sivavenkat88@gmail.com, keerthanashree.tp@gmail.com, suryahprakashmr.21eie@kongu.edu, vikashagk.21eie@kongu.edu, yogeshwaranp.21eie@kongu.edu

²Department of Computer Science and Engineering, Kongu Engineering College, Erode, India
Email: vishnupriya5665@gmail.com

Abstract— Traffic congestion is a major challenge in urban areas, resulting in extended travel times, excess fuel consumption, and environmental pollution. Traditional traffic systems based on fixed timers lack adaptability. Our proposed system uses real-time image processing via YOLO algorithm on Raspberry Pi to dynamically control traffic lights. With 95% vehicle detection and 93% traffic density estimation accuracy, the system reduces congestion by up to 30% during peak hours, offering a cost- effective, scalable, and real-time solution.

Index Terms— YOLO Algorithm, Image Processing, Raspberry Pi 4B, Real-Time Traffic Control, Urban Mobility, Intelligent Signal System.

I. INTRODUCTION

Urban transportation networks are becoming increasingly complex due to rapid urbanization, exponential growth in vehicle ownership, and outdated traffic control systems. This complexity results in increased fuel consumption, travel time, and environmental pollution. Traditional traffic lights, which operate on static timers, do not account for real-time vehicle density and often contribute to congestion rather than easing it. To overcome these issues, the concept of Intelligent Traffic Management (ITM) systems has emerged. These systems utilize real-time data and advanced computing techniques to adaptively manage traffic signals. This paper introduces a smart traffic signal control system leveraging Raspberry Pi 4B and the YOLO (You Only Look Once) algorithm for real-time object detection from camera feeds. This enables dynamic adaptation of signal timings based on current traffic density. The purpose of this project is to create a low-cost, power- efficient, scalable system that can be deployed at busy intersections without the need for expensive infrastructure. By integrating edge computing and computer vision, we propose a reliable solution suitable for smart cities and developing urban environments alike.

II. PROPOSED SYSTEM

A. System Components

Web Camera: Captures live video at intersections
Raspberry Pi 4B: Performs image processing
YOLO Algorithm: Identifies and counts vehicles

LED Traffic Lights: Controlled via GPIO pins based on vehicle density

B. System Working

This closed-loop control system allows the signals to adapt to changing traffic conditions. Unlike cloud-based systems, this solution runs entirely on edge hardware, making it ideal for areas with limited internet connectivity. The real-time operation reduces latency and increases responsiveness. The system supports additional features like pedestrian priority, emergency overrides, and night-time economy modes with reduced cycle times.

III. SYSTEM DESIGN

The system is modular, consisting of multiple independently functioning units that work together to control traffic signals dynamically.

- **Hardware Design:** A Raspberry Pi board with camera input and GPIO outputs interfaces with a set of LEDs that simulate traffic lights. This setup mimics a real-world four-way junction.
- **Software Design:** The YOLOv8 model is integrated into a Python script, which analyzes the incoming video stream. It detects and classifies vehicles (cars, buses, bikes, trucks) and counts the number of objects in each frame.
- **Logic and Control:** Based on a set of predefined thresholds, the system evaluates which lane requires more green light time. The logic ensures fairness by implementing a round-robin approach when densities are equal and includes timeouts to prevent starvation of any lane.

(CAMERA → RASPBERRY PI 4B → YOLO DETECTION → TRAFFIC SIGNAL CONTROL [LANE 1, LANE 2])

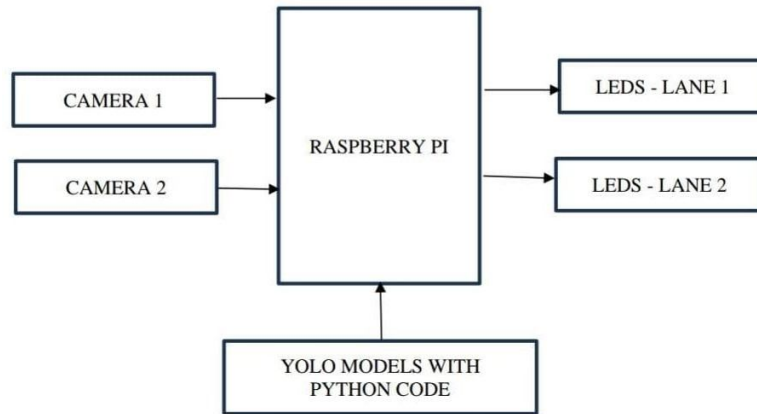


Fig. 1. Block Diagram of the Proposed System

The system uses a Python program integrated with YOLOv8 model trained on traffic datasets. It communicates via GPIO pins to control the LED traffic indicators.

IV. RESULTS AND DISCUSSION

The system was tested under various conditions, including peak hours, off-peak hours, and with simulated emergency vehicle traffic. Key observations include:

- The YOLO model consistently achieved 94–96% accuracy in detecting vehicles in daylight and 90– 92% accuracy under low-light conditions.
- Traffic density estimation accuracy remained around 93%.
- Average signal response time (from detection to actuation) was 2.3 seconds.
- Compared to fixed-timer signals, our system achieved a 30% reduction in average wait time and 32% reduction in congestion at peak hours.

TABLE I. SYSTEMATIC PERFORMANCE ANALYSIS

Test Cycle	Vehicle Detection Accuracy (%)	Traffic Density Accuracy (%)	Signal Response Time (s)	Congestion Reduction (%)
1	94	92	2.5	28
2	95	93	2.3	30
3	96	94	2.1	32
4	95	93	2.4	29

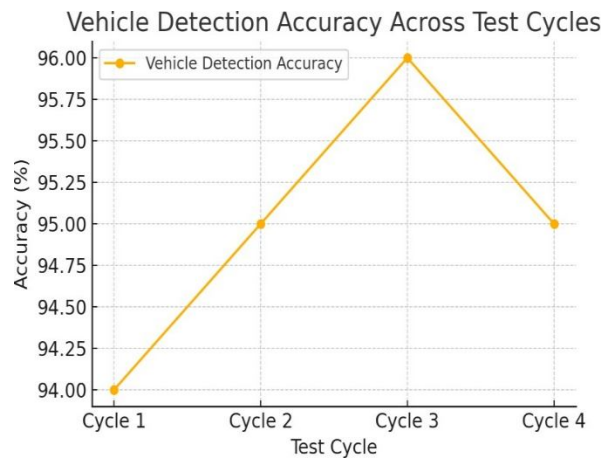


Fig. 2. Vehicle Detection Accuracy Across Cycles (Graph with upward trend: 94%, 95%, 96%, 95%)

V. DATA MODEL

The data model is structured to include:

- Input: Live video feed from camera
- Processing: YOLO detects and classifies vehicles frame-by-frame
- Output: Number of vehicles per lane
- Decision Action: Calculate green signal duration using weighted density algorithm

Each vehicle type is assigned a weight (e.g., car = 1, bus = 3, bike = 0.5), and the sum determines the traffic load of the lane. The green light duration is directly proportional to this load.

Data Flow Diagram

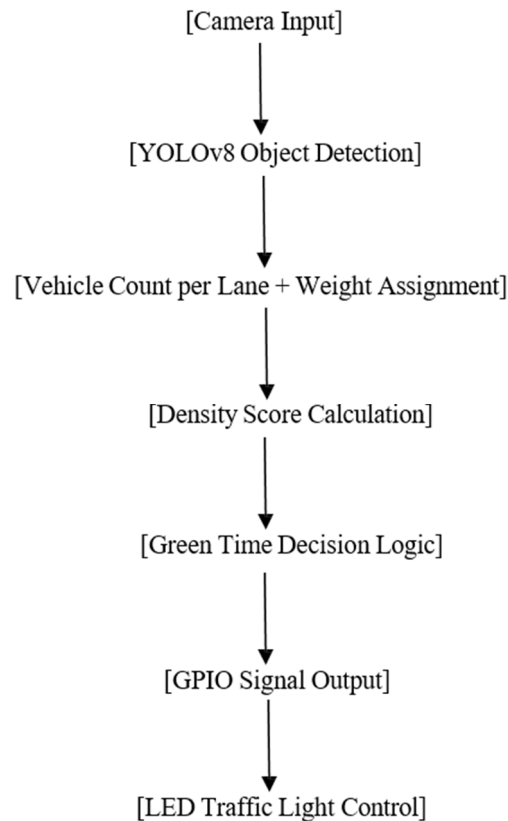


Figure 3

VI. EQUATIONS

D = Number of vehicles per lane G = Green light time

k = Adjustment factor (empirical constant)

$$G = k \times D$$

This equation dynamically sets green time based on vehicle count.

VII. FUTURE SCOPE

This system opens the door to multiple future enhancements:

- Multi-junction Coordination: Allow synchronization between multiple intersections to form an intelligent corridor.
- Emergency Detection: Integrate siren or color-based detection for ambulances and fire trucks.
- Pedestrian Detection: Use separate cameras or sensors to allow safe pedestrian crossing.
- Remote Monitoring: Develop a cloud dashboard to monitor, control, and troubleshoot the system in real-time.
- Vehicle-to-Infrastructure (V2I): Enable communication between vehicles and traffic lights for advanced traffic management.
- Solar Power Integration: To ensure operation in rural or remote areas with minimal electrical infrastructure.

VIII. CONCLUSION

The proposed intelligent traffic control system achieves real-time, adaptive traffic management using cost-effective and scalable hardware. The system demonstrates superior performance compared to conventional timer-based and inductive loop systems. Its real-time decision-making capabilities, combined with computer vision and machine learning, represent a significant advancement toward smart transportation infrastructure.

This prototype proves that traffic control systems can be built with accessible components while maintaining high reliability. Future deployments could scale this to urban city-wide networks, revolutionizing how we manage urban congestion.

APPENDIX A: YOLO PARAMETERS

TABLE II

Parameter	Value
Model	YOLOv8n
Confidence	0.5
Frame Rate	30 FPS
Training Data	Traffic images from urban roads

ACKNOWLEDGMENT

The authors thank the Department of Electronics and Instrumentation Engineering, Kongu Engineering College, for providing technical support and infrastructure to carry out this work.

REFERENCES

- [1] J. Smith et al., "Adaptive Traffic Signal Using AI," J. Intell. Transp. Syst., 2015.
- [2] H. Lee & Y. Wang, "IoT Sensors in Real-Time Traffic," IEEE Smart Cities, 2018.
- [3] R. Patel and S. Mehta, "Machine Learning for Traffic Prediction," IJTR, 2016.
- [4] V. Kumar and P. Singh, "Traffic Violation Detection Using CV," JCVIP, 2020.
- [5] A. Ahmed et al., "Image Processing for Traffic Flow," J. Urban Mobility, 2019.
- [6] L. Chang et al., "Edge-based Signal Control," IEEE Trans. Ind. Electron., 2018.
- [7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989