IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁙ Peer-reviewed & Refereed journal ⁙ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154172

Smart Traffic Management System Using Machine Learning for Adaptive Signal Control and Emergency Vehicle Prioritization

Aditya Nanote[1], Srujal Ogale[2], Vikrant Palvi[3], Lokesh Patil[4], Prof. Rakesh C. Suryawanshi[5]

Student, Department of Computer Engineering, A.C. Patil College of Engineering, Kharghar, Navi Mumbai, India[1,2,3,4]

Professor, Department of Computer Engineering, A.C. Patil College of Engineering, Kharghar, Navi Mumbai, India[5]

Abstract: The rapid growth of urban populations and vehicle density has resulted in severe traffic congestion, increased travel time, and environmental pollution in modern cities. Conventional traffic control systems rely on fixed signal timings or reactive approaches, which are inefficient in handling dynamic and unpredictable traffic conditions. This paper proposes a Smart Traffic Management System (STMS) that leverages machine learning and data-driven techniques to optimize traffic flow. The system employs a Random Forest Regression model to predict future traffic density based on historical datasets and simulated scenarios, analysing parameters such as vehicle count, time intervals, and lane-wise traffic patterns. An intelligent decision-making module dynamically adjusts signal timings based on these predictions. The system additionally incorporates an emergency vehicle prioritization mechanism and a visualization module providing analytical insights into traffic trends and prediction accuracy. By eliminating costly IoT infrastructure and focusing on predictive analytics, the proposed system offers a cost-effective, scalable, and flexible solution for modern traffic management, achieving over 96% packet delivery reliability and end-to-end latency as low as 200-350 ms under Wi-Fi conditions.

Keywords: Adaptive Signal Control, Emergency Vehicle Prioritization, IoT, Machine Learning, Predictive Analytics, Random Forest Regression, Smart Traffic Management.

I. INTRODUCTION

With the rapid increase in urbanization and vehicle ownership, traffic congestion has become a major challenge in modern cities. Traditional traffic control systems operate on fixed time intervals and often fail to respond to real-time traffic conditions, leading to delays, fuel wastage, and increased pollution. The emergence of Artificial Intelligence (AI) and the Internet of Things (IoT) offers innovative solutions to these issues by enabling the collection and analysis of real-time data from interconnected sensors and devices.

Conventional traffic management systems rely on pre-programmed signal timings that operate in fixed cycles, without considering actual road conditions. As a result, vehicles often experience unnecessary delays at intersections, contributing to increased air pollution, driver frustration, and economic losses. An IoT-based Smart Traffic Management System (STMS) uses sensors, microcontrollers, and communication networks to collect real-time traffic flow data. The data is transmitted to a central processing unit or cloud server where algorithms make intelligent decisions about signal control, dynamically adjusting light durations and providing green corridors for emergency vehicles.

A. Motivation

The motivation behind this project stems from the need to address critical challenges: the increasing number of vehicles causing severe congestion; static signal systems that do not adapt to real-time conditions; excessive fuel consumption and air pollution due to prolonged idling; and the absence of automated priority handling for emergency vehicles such as ambulances and fire trucks.

B. Problem Statement

Urban areas face severe traffic congestion due to the rapid increase in vehicle numbers and the inefficiency of traditional traffic control systems. Conventional traffic signals operate on fixed time intervals, failing to adapt to real-time road conditions. An IoT-based intelligent traffic management system is therefore needed to dynamically monitor vehicle density, adjust signal timings, prioritize emergency vehicles, and generate real-time insights with minimal human intervention.

C. Objectives

The primary objectives of this project are to:

1. Design an IoT-based STMS capable of real-time monitoring and intelligent control.
2. Deploy a machine learning model (Random Forest Regression) for traffic density prediction.

3. Dynamically adjust signal timings according to predicted traffic conditions.
4. Implement emergency vehicle prioritization with automatic green corridors.
5. Provide a centralized cloud-based dashboard for remote monitoring.
6. Contribute to smart city infrastructure by reducing fuel consumption and emissions.

II. LITERATURE SURVEY

A. Survey of Existing Systems

Existing traffic management systems are primarily based on conventional fixed-timer signal control technology that operates independently of actual traffic conditions. Early responses to these limitations include actuated traffic signals and advanced systems such as SCOOT (Split Cycle Offset Optimization Technique) and SCATS (Sydney Coordinated Adaptive Traffic System). However, these require significant infrastructure investment and centralized control, making them difficult to scale in developing urban areas.

The emergence of IoT and Machine Learning technologies has opened new avenues for intelligent traffic management. Researchers have explored combinations of sensors, edge computing, cloud analytics, and AI-driven signal control. Despite these advances, many implementations suffer from high deployment costs, dependency on proprietary hardware, lack of interoperability, and limited scalability.

B. Limitations of Existing Systems

- Fixed signal timings do not adapt to real-time traffic variations.
- Lack of real-time data collection prevents proactive management.
- No provision for automatic emergency vehicle prioritization.
- High infrastructure and maintenance costs limit scalability.
- Poor integration with modern IoT, cloud, and AI platforms.
- Insufficient support for sustainable urban mobility goals.

C. Research Gap

Despite significant advancements, existing solutions remain largely fragmented, focusing on isolated components rather than providing a fully integrated end-to-end architecture. Many implementations rely heavily on cloud-centric processing, leading to increased latency and reduced reliability. Additionally, most systems depend on proprietary ecosystems such as AWS, Azure, and GCP, resulting in vendor lock-in and high operational costs. Security also remains a critical concern, as lightweight IoT systems frequently implement weak authentication and unencrypted protocols.

D. Related Work

Table I summarizes key related works in Smart Traffic Management Systems.

TABLE I SUMMARY OF RELATED WORK IN SMART TRAFFIC MANAGEMENT

Paper Title	Authors	Methodology	Key Findings / Limitations
IoT-Based Adaptive Traffic Management	Lilhore U.K. (2022)	IoT sensors, DBSCAN clustering, adaptive signals	Effective real-time adaptation; complex setup
ML and YOLO for Adaptive Traffic	Sakhare et al. (2024)	Camera + YOLO vehicle detection	Lane-level detection; high edge cost
Fog-IoT for Traffic Surveillance	Chen et al.	Fog nodes + iFogSim	Reduced latency; simulation-only
IoT WSN Traffic Congestion	Vadivel et al. (2023)	IoT WSN + cloud analysis	Effective detection; limited edge AI
Smart City Traffic Survey	Anaswara et al. (2016)	IoT review	Broad overview; lacks evaluation

III. PROPOSED SYSTEM

A. System Overview

The proposed Smart Traffic Management System aims to overcome the limitations of traditional traffic control methods by leveraging machine learning, data-driven analytics, and intelligent decision-making. Unlike traditional approaches, the system introduces a predictive and adaptive framework that dynamically adjusts traffic signal timings based on anticipated traffic conditions.

The system operates entirely in a software environment using historical traffic datasets and simulated traffic inputs, eliminating the need for costly physical sensor deployment. The core is built around a Random Forest Regression model that predicts future traffic density based on vehicle count, time intervals, and lane-specific data.

B. Key Features

- Predictive Traffic Analysis: ML algorithms forecast future density enabling proactive management.
- Adaptive Signal Optimization: Green durations are dynamically adjusted per predicted load.
- Emergency Vehicle Prioritization: Priority signal overrides ensure faster green corridors for critical vehicles.
- Visualization and Analytics: Graphical dashboards display waiting time comparisons, traffic trends, and prediction accuracy.
- Modular and Scalable: Supports multiple intersections with minimal structural changes.
- Cost-Effective: Purely software-based; eliminates hardware deployment costs.

C. System Architecture

The architecture follows a multi-layered modular design. Four layers constitute the system:

Perception Layer: Road-side IoT sensor nodes equipped with ultrasonic/infrared sensors, camera modules, and ESP32 microcontrollers continuously monitor vehicle presence and density.

Network and Communication Layer: MQTT protocol with JSON payloads provides lightweight transmission over Wi-Fi or cellular networks. An optional Raspberry Pi gateway handles edge-level preprocessing.

Data Processing and Application Layer: The cloud server receives sensor data, performs real-time traffic analysis, runs signal optimization algorithms, and triggers emergency vehicle preemption.

User Interface Layer: A web-based dashboard enables traffic authorities to monitor live junction statuses, view density charts, and receive incident alerts.

D. System Components

Traffic Data Input Module: Utilizes structured datasets or simulated traffic inputs containing vehicle count, timestamp, lane identification, and traffic density to model real-world scenarios.

Data Preprocessing Module: Handles missing values, removes noise, and applies normalization to ensure consistency for reliable machine learning operations.

Feature Engineering Module: Extracts meaningful attributes such as traffic density trends, peak-hour patterns, and lane-wise variations. Partial correlation analysis identifies significant relationships between variables and eliminates redundant features.

Machine Learning Prediction Module: A Random Forest Regression model forms the core predictive engine. It captures non-linear relationships within the data and produces robust predictions for dynamic traffic conditions.

Decision-Making Engine: Utilizes predicted traffic values to determine optimal signal timing per lane, dynamically allocating green durations based on traffic demand.

Emergency Handling Module: Simulates emergency vehicle presence and overrides normal signal operation to provide immediate lane clearance, improving response time for critical services.

Visualization and Analytics Module: Generates charts including waiting time comparison, predicted vs. actual traffic density, and overall traffic trends for effective system evaluation.

IV. METHODOLOGY

A. Data Collection and Preprocessing

The system processes structured CSV-based traffic datasets containing vehicle count, timestamps, and lane-specific data. Preprocessing includes removal of missing/inconsistent values and noise; Min-Max normalization of numerical features; temporal feature extraction (hour-of-day, day-of-week); and partial correlation analysis to identify statistically significant predictors.

B. Random Forest Regression Model

Random Forest Regression was selected due to its ability to handle non-linear relationships, robustness to overfitting through ensemble averaging, and high performance on tabular traffic datasets. Given n decision trees, the prediction for traffic density y-hat is the average of all tree predictions Ti(x) for input feature vector x. The model was trained on 80% of the dataset and validated on the remaining 20%.

C. Adaptive Signal Timing Algorithm

The Decision-Making Engine computes green time Gj for lane j proportionally to its predicted density. The green time is calculated as: Gj = Gmin + (Gmax - Gmin) * (predicted density of lane j / sum of predicted densities of all lanes), where Gmin and Gmax are minimum and maximum green durations respectively, and L is the total number of lanes.

D. Emergency Vehicle Prioritization

Emergency scenarios are triggered via predefined simulation inputs. Upon detection, the system executes the overrideSignal() function in the Emergency Handler module, assigning maximum green time to the affected lane and logging the priority event with duration and vehicle ID.

E. MQTT Message Schema

IoT nodes publish traffic data using a standardized JSON payload containing: device_id (e.g., "junction01"), timestamp, sensor readings (vehicle_count, avg_speed, lane_status), and signal_state fields. This lightweight schema ensures efficient transmission over constrained IoT networks.

V. SYSTEM REQUIREMENTS

A. Hardware Requirements

- Processor: Intel i5 / AMD Ryzen 5 or higher (quad-core)
- RAM: Minimum 8 GB (16 GB recommended for training)
- Storage: 256 GB SSD
- GPU: NVIDIA GPU (optional, for accelerating ML tasks)
- IoT Nodes: ESP32 microcontrollers with ultrasonic/IR sensors
- Gateway: Raspberry Pi 4 (edge processing)

B. Software Requirements

- Programming Language: Python 3.x
- Libraries: NumPy, Pandas, Scikit-learn, Matplotlib, Seaborn
- ML Model: Random Forest Regression (Scikit-learn)
- Communication: Mosquitto MQTT Broker
- Database: MySQL / Firebase
- Dashboard: Streamlit / Flask (optional)
- IDE: Jupyter Notebook / VS Code / PyCharm

VI. RESULTS AND DISCUSSION

A. Experimental Setup

The prototype was deployed using three ESP32 nodes functioning as junction sensors and one Raspberry Pi acting as a local MQTT gateway. Tests were conducted to measure latency, packet delivery ratio, power consumption, and signal optimization effectiveness under simulated real-world traffic conditions.

B. Performance Results

Table II summarizes the key experimental results. The system demonstrated reliable data collection, efficient MQTT communication, and responsive signal control.

TABLE II EXPERIMENTAL TEST RESULT SUMMARY

Test Parameter	Result
Latency (Wi-Fi)	200-350 ms typical
Latency (Cellular Hotspot)	500-800 ms typical
Packet Delivery Ratio (QoS 1)	>96%
Battery Life Estimate	60-80 hours

Under Wi-Fi conditions, end-to-end latency between vehicle detection and signal actuation was measured at 200-350 ms, well within acceptable limits for real-time traffic management. Cellular connections yielded 500-800 ms latency due to higher network overhead but remained functionally effective. MQTT message delivery with QoS Level 1 exceeded 96%, confirming robust communication layer reliability.

C. Adaptive Signal Control Performance

DOI: 10.17148/IJARCCE.2026.154172

Adaptive signal control demonstrated a measurable reduction in average vehicle waiting time compared to fixed-timing scenarios during simulated high-density traffic periods. Emergency vehicle preemption was successfully triggered in all test cases, confirming the reliability of the green corridor mechanism.

D. Discussion

The experimental results confirm that the proposed IoT-based Smart Traffic Management System effectively addresses the limitations of conventional fixed-timer traffic signals. The modular architecture proved flexible and easy to extend, with new IoT nodes integrating into the network with minimal configuration effort. MQTT with JSON payloads provided a lightweight and scalable communication framework suitable for IoT deployments with constrained bandwidth. Edge processing via Raspberry Pi demonstrated measurable improvements in response latency and provided resilience against temporary cloud connectivity interruptions.

Key areas for improvement include: increased sensor accuracy under adverse weather conditions; reduced cellular latency through enhanced edge computing; and more sophisticated ML-based signal optimization for multi-junction coordination.

VII. CONCLUSION

The Smart Traffic Management System represents a significant step forward in addressing urban traffic congestion through intelligent automation. By combining IoT sensor networks, real-time data processing, a Random Forest Regression predictive model, adaptive signal control, and cloud-based monitoring, the system provides a cost-effective and scalable solution that overcomes fundamental limitations of traditional fixed-timer systems.

The system successfully demonstrated reliable vehicle detection, low-latency signal control, effective emergency vehicle prioritization, and robust data transmission in prototype testing. Its modular architecture ensures ease of deployment, maintenance, and future expansion without complete system redesign. The project highlights how affordable IoT hardware, open-source software tools, and lightweight communication protocols can deliver intelligent infrastructure solutions suitable for smart city environments.

Key outcomes include: adaptive traffic signal optimization reducing average vehicle waiting time; reliable communication with more than 96% MQTT packet delivery ratio; fully automated emergency vehicle prioritization with green corridor creation; scalable, modular deployment supporting multiple urban junctions; and a foundation for future AI-enhanced smart city transportation systems.

Future work will focus on integrating deep learning and reinforcement learning models for enhanced prediction, incorporating GPS-based emergency vehicle tracking across multiple junctions, developing mobile application interfaces for commuter route guidance, and deploying federated cloud architectures for city-scale traffic optimization.

ACKNOWLEDGMENT

The authors sincerely thank Prof. R. C. Suryawanshi for constant guidance and invaluable support throughout the development of this project. The authors also express gratitude to Dr. M. M. Deshpande, Head of Department, and all faculty members of the Department of Computer Engineering, A.C. Patil College of Engineering, Kharghar, Navi Mumbai, for providing the necessary facilities, resources, and academic environment.

REFERENCES

[1] Ministry of Road Transport and Highways, "Road Transport Yearbook 2021-22," Government of India, New Delhi, 2022.

[2] Transport Corporation of India, "TCI-IIMC Logistics Report 2023," TCI Foundation, Gurugram, 2023.

[3] A. Singhal, R. Mehta, and S. Patel, "Emergency response time analysis in Indian metropolitan areas," Journal of Urban Health, vol. 98, no. 3, pp. 412-421, 2021.

[4] M. Papageorgiou, C. Diakaki, V. Dinopoulou, A. Kotsialos, and Y. Wang, "Review of road traffic control strategies," Proceedings of the IEEE, vol. 91, no. 12, pp. 2043-2067, 2003.

[5] F. V. Webster, "Traffic signal settings," Road Research Technical Paper No. 39, HMSO, London, 1958.

[6] D. I. Robertson and R. D. Bretherton, "Optimising networks of traffic signals in real time - the SCOOT method," IEEE Transactions on Vehicular Technology, vol. 40, no. 1, pp. 11-15, 1991.

[7] P. R. Lowrie, "The Sydney co-ordinated adaptive traffic system - principles, methodology, algorithms," in Proc. IEE Int. Conf. on Road Traffic Signalling, London, 1982, pp. 67-70.

[8] S. Bharti, A. Tiwari, and N. Joshi, "IoT based adaptive traffic signal control using IR sensors," in Proc. IEEE ICPC2T, Raipur, 2020, pp. 1-5.

[9] R. Patil and A. Kulkarni, "ESP8266-based smart traffic light system with density detection," Int. J. Engineering Research and Technology, vol. 10, no. 4, pp. 234-238, 2021.

[10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in Proc. IEEE CVPR, Las Vegas, 2016, pp. 779-788.

DOI: 10.17148/IJARCCE.2026.154172

[11] X. Zhang, W. Yang, Z. Tang, and J. Liu, "A fast learning method for accurate and robust lane detection using two-stage feature extraction with YOLO," Sensors, vol. 18, no. 12, p. 4308, 2019.

[12] G. Jocher et al., "ultralytics/yolov5: v6.0 - YOLOv5n 'Nano' models," Zenodo, 2020.

[13] R. Nelson and S. Bullock, "Impact of GPS based priority on transit and auto traffic: simulation study," Transportation Research Record, no. 1634, pp. 86-93, 1998.

[14] C. Wu, D. Chen, and J. Wu, "A siren detector for emergency vehicle preemption," in Proc. IEEE ISCAS, Melbourne, 2014, pp. 2662-2665.

[15] V. Sharma, A. Gupta, and S. Mishra, "Emergency vehicle detection at traffic intersections using image processing," in Proc. IEEE ICRITO, Noida, 2019, pp. 1-5.

[16] Y. Liu, J. Guo, and L. Li, "Cooperative signal control at arterials using vehicle infrastructure integration," Transportation Research Part C, vol. 98, pp. 27-44, 2019.