International Journal of Innovative Information Systems & Technology Research 19(4):1-9, Oct-Dec, 2026

© SEAHI PUBLICATIONS, 2026 **www.seahipublications.org ISSN: 2467-8562**

DOI:10.5281/zenodo.23143010

Real-Time Smart City Vehicular Traffic Monitoring System In Port Harcourt

G. K. David, E. O. Bennett, D. J. S. Sako

Department of Computer Science, Rivers State University, Port Harcourt, Nigeria

Corresponding Author: David, Greatman Kio Corresponding Author's Email: bennett.okoni@ust.edu.ng, david.greatman@rsu.edu.ng

ABSTRACT

Rapid urbanization, increasing vehicle ownership, and inadequate access to continuous traffic information have intensified vehicular congestion and transportation-management challenges in Port Harcourt. This study presents a Real-Time Smart City Vehicular Traffic Monitoring System (RT-SCVTMS) that integrates Internet of Things (IoT), computer vision, artificial intelligence, edge computing, MQTT communication, MySQL storage, and a web-based monitoring dashboard. Live traffic video is acquired through an IP camera and processed at the edge using the YOLOv8 object detection model. The system detects, classifies, tracks, and counts vehicles, estimates traffic density, identifies congestion conditions, transmits processed metadata through MQTT, stores traffic records, and provides real-time and historical visualization. The updated vehicle-counting result recorded 720 vehicles comprising 214 cars, 187 commercial buses, 284 mini buses (keke buses), 19 trucks/heavy vehicles, and 16 motorcycles. Rumuokoro Junction records were maintained at 15-minute intervals, with vehicle counts ranging from 547 to 693. Under the simulated testing conditions, the system achieved 97.8% detection accuracy, 98.2% precision, 97.3% recall, 97.7% F1-score, 41 ms processing latency, 24 FPS throughput, and 99.8% MQTT delivery reliability. The findings indicate that the proposed edge-oriented framework can provide near-real-time traffic intelligence while reducing dependence on manual observation and continuous transmission of raw video.

Keywords: Smart City; Vehicular Traffic Monitoring; IoT; YOLOv8; Edge Computing; MQTT; Traffic Analytics.

1. INTRODUCTION

Rapid urbanization and population growth have increased transportation demand and vehicular movement in major cities. The United Nations Department of Economic and Social Affairs reported continued global urbanization, creating pressure on transport infrastructure and increasing the need for intelligent traffic-management approaches [1]. In developing cities such as Port Harcourt, increasing vehicle ownership and commercial activity have contributed to congestion, travel delays, environmental pollution, and reduced transportation efficiency.

Traditional traffic monitoring practices, including manual traffic counting, fixed-time traffic signals, and isolated surveillance, provide useful observations but have limited capacity for continuous automated

analysis. In the Port Harcourt context, manual traffic counting has been used for traffic-flow assessment [2], while traffic-management studies have documented the importance of traffic wardens and conventional signals [3]. These approaches can make it difficult to obtain continuous vehicle-level information and respond quickly to changing traffic conditions.

Recent advances in IoT, computer vision, artificial intelligence, and edge computing provide opportunities for more responsive traffic monitoring. IoT sensing enables continuous collection of transportation information [4], while computer-vision techniques can automatically detect, classify, count, and track vehicles from traffic video [5]. Edge computing further allows computational processing to be performed close to the point of data generation, thereby reducing communication latency and bandwidth requirements for time-sensitive applications [6].

Several recent systems have demonstrated the potential of edge AI for transportation. [7] investigated edge-enabled video analytics for smart transportation, while [8] developed Edge YOLO for real-time object detection through edge-cloud cooperation. [9] applied YOLO-based detection and density estimation to adaptive traffic-light control. These studies establish the technical feasibility of edge-based traffic intelligence, but the present study focuses on an integrated monitoring platform for Port Harcourt that combines live detection, vehicle counting, congestion analysis, IoT communication, database storage, and dashboard visualization.

The objective of RT-SCVTMS is therefore to provide a practical real-time traffic-monitoring framework that converts traffic-camera observations into structured traffic information. The system is designed around edge processing so that traffic video can be analysed locally before selected metadata are communicated to the monitoring platform.

2. RELATED WORKS

Intelligent transportation research has progressed from manual surveys and fixed-time control toward sensor-based monitoring, computer vision, IoT communication, and artificial intelligence. The evolution has increased the ability of traffic systems to acquire and process transportation information continuously. However, the usefulness of an intelligent traffic platform depends not only on detection accuracy but also on processing latency, communication reliability, data storage, and accessibility of the resulting information.

The demonstration of the relevance of edge computing for video segmentation and object detection in smart transportation was done by [7]. Their work supports processing traffic information near the source to reduce the amount of information that must be transferred. [8] reported an Edge YOLO system for autonomous-vehicle object detection, showing that real-time object detection can be supported through edge-cloud cooperation. These systems are technically related to the present work, but their primary focus is object detection rather than an end-to-end urban traffic-monitoring dashboard.

The development of an edge-deployed adaptive traffic-light control system using YOLO-based vehicle detection and PCE-aware density estimation was done by [9]. The study reported 90% mAP and 74 FPS and focused on adaptive signal control. RT-SCVTMS differs in its primary functional objective: it provides real-time vehicular monitoring, classification, counting, congestion estimation, historical storage, MQTT communication, and web-based visualization rather than concentrating mainly on traffic-signal control.

Investigations on AI-based traffic-congestion prediction in Port Harcourt using historical traffic records and reported high predictive accuracy was carried out by [10]. The present study extends the local traffic-monitoring perspective by using live camera input, edge AI processing, automated vehicle detection, and structured real-time traffic reporting. The distinction is important because historical prediction and live observation address related but different stages of traffic intelligence.

3. METHODOLOGY

The study adopted a Design and Development Research approach and used a prototype-oriented development process. The work involved requirements identification, system architecture design,

functional design, implementation, testing, and performance evaluation. The design integrates camera-based traffic acquisition, edge AI processing, traffic analytics, IoT communication, database storage, and web visualization.

The proposed architecture is divided into functional layers that support the movement of traffic information from data acquisition to analysis and visualization. Figure 1 presents the exact architecture adopted from the project work, showing the relationship among the communication, edge processing, artificial-intelligence, analytics, database, and visualization components.

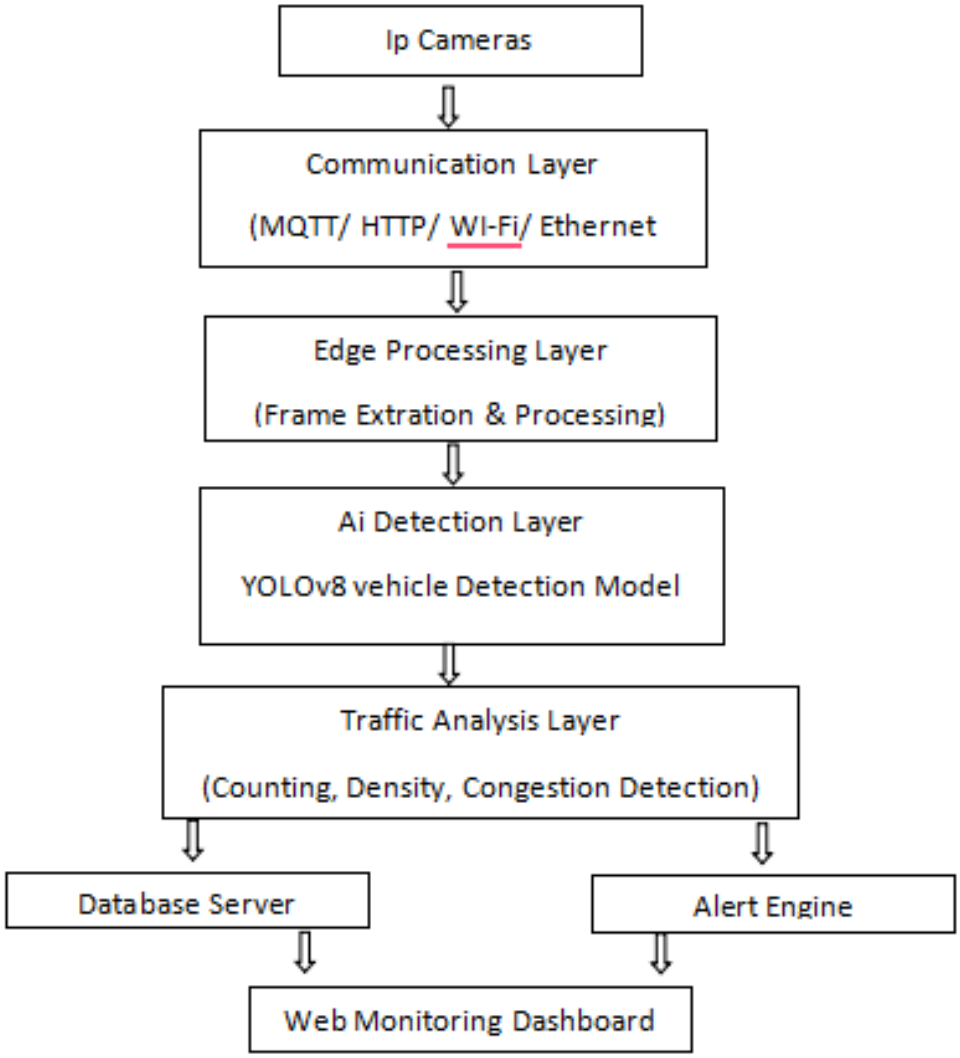

Figure 1: Architecture of the Proposed System

The architecture places computationally intensive traffic-video processing at the edge. After acquisition, frames are preprocessed and analysed by the YOLOv8 detection module. The resulting traffic metadata are then used by the analytics module for vehicle counting, density estimation, and congestion assessment. MQTT provides lightweight message communication, while the database and dashboard support storage, retrieval, visualization, and historical reporting.

3.1 Functional Design

The functional design defines what the proposed RT-SCVTMS does and how its major functions interact. The design begins with traffic-data acquisition and proceeds through preprocessing, AI-based detection and classification, tracking and counting, traffic analytics, communication, storage, and visualization. This functional sequence ensures that each subsystem has a clearly defined responsibility and that the output of one function becomes the input required by the next.

S/N	Functional Module	Main Function	Output
1	Traffic Data Acquisition	Capture live traffic video from the monitoring camera.	Video frames
2	Frame Preprocessing	Resize, enhance, normalize, and prepare frames for inference.	Processed frames
3	Vehicle Detection	Apply YOLOv8 to locate vehicles in each frame.	Detected vehicle objects
4	Vehicle Classification	Assign detected objects to vehicle categories.	Vehicle classes and confidence values
5	Tracking and Counting	Track objects and count each vehicle once within the counting region.	Vehicle counts
6	Traffic Analytics	Estimate traffic density and congestion condition.	Traffic status and alerts
7	IoT Communication	Transmit processed traffic metadata through MQTT.	Traffic messages
8	Database Management	Store counts, density, locations, timestamps, and alerts.	Historical traffic records
9	Dashboard Visualization	Display current traffic status and historical reports.	Monitoring information

Table 1: Functional Design of the Proposed RT-SCVTMS

The component hierarchy used in the project further clarifies how the functional modules are decomposed into smaller system components. Figure 2 presents the exact component hierarchy from the project and shows the relationship among the principal subsystems of the proposed monitoring platform.

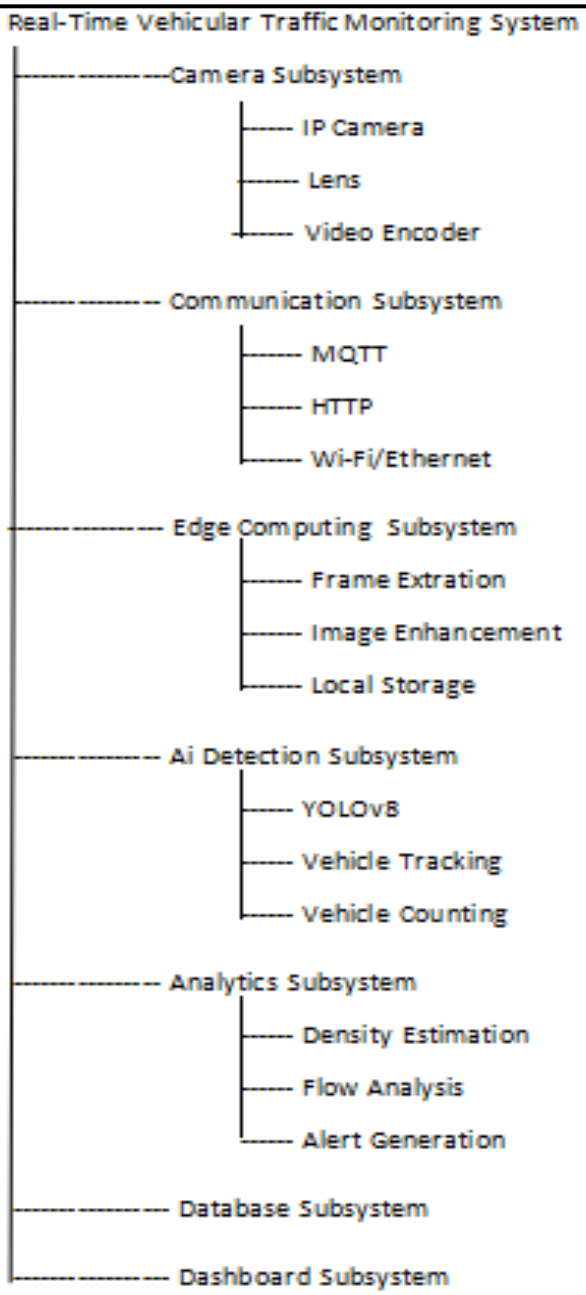

Figure 2: Component Hierarchy of the Proposed System

3.2 Data Acquisition and Processing

The system uses traffic video captured from an HD surveillance camera. The project specifies 1280 × 720-pixel input at approximately 30 FPS, with frame-by-frame processing using OpenCV and YOLOv8. Captured frames undergo preprocessing before being passed to the AI detection stage. The AI module detects and classifies vehicles, after which the tracking and counting functions prevent duplicate counting. Traffic analytics subsequently determine vehicle volume, density, and congestion condition.

Table 2 presents the principal characteristics of the sample traffic input used for system testing.

Parameter	Description
Input Source	Live traffic video captured from an HD surveillance camera
Input Type	Video stream (image frames)
Image Format	JPEG/RGB frames
Resolution	1280 × 720 pixels (HD)
Frame Rate	30 frames per second (FPS)
Camera Position	Roadside-mounted camera overlooking the roadway
Processing Method	Frame-by-frame analysis using OpenCV
Detection Model	YOLOv8
Vehicle Categories	Car, Bus, Truck, Motorcycle, Van
Traffic Condition	Normal daytime traffic
Processing Platform	Edge computing device running Python and OpenCV

Table 2: Characteristics of the Sample Traffic Input

4. RESULTS AND DISCUSSION

This section presents the principal results generated by the completed RT-SCVTMS prototype. The system successfully performed traffic-data acquisition, vehicle detection, classification, counting, density estimation, congestion monitoring, MQTT communication, database storage, and historical visualization.

4.1 Vehicle Detection and Classification

The vehicle-detection module automatically identified vehicles within the monitored traffic scene and assigned vehicle categories. The project recorded 720 vehicles in the sample classification result. Table 4 presents the category-level counts obtained from the system.

Table 3: Sample Vehicle Detection and Counting Result

Vehicle Category	Number Detected
Cars	214
Commercial Buses	187
Mini Bus (keke bus)	284
Trucks / Heavy	19
Motorcycles	16
Total Vehicles	720

The graphical representation corresponding to Table 4 is presented in Figure 3. The figure provides a visual comparison of the frequency of the detected vehicle categories.

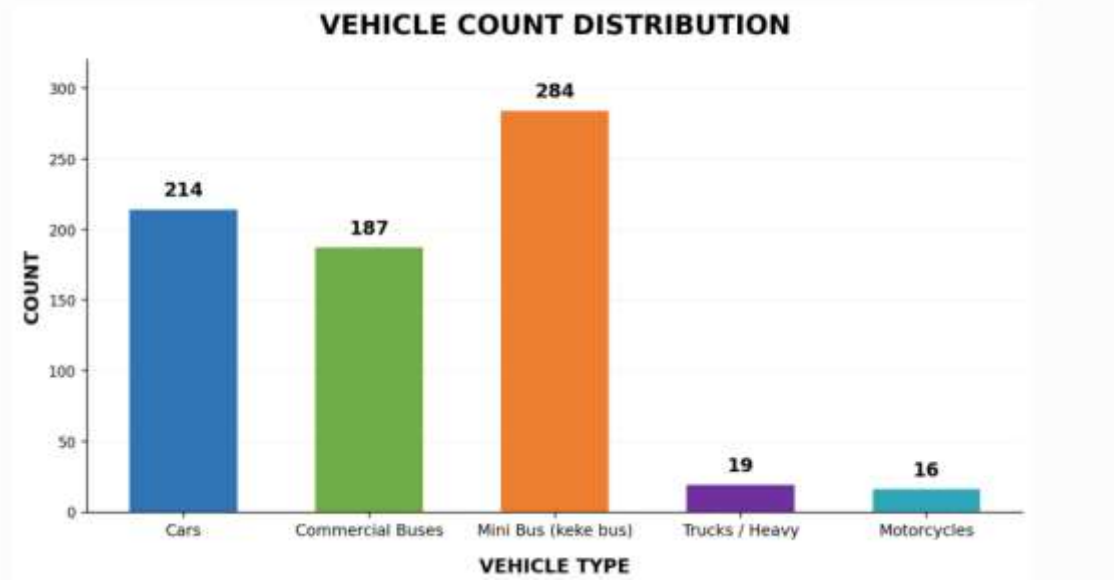

Figure 3: Vehicle Count Distribution

4.2 Traffic Records and Historical Monitoring

The historical monitoring function stores traffic observations with their location, vehicle count, density condition, and timestamp. This enables traffic operators to review changes in traffic volume over time rather than relying only on a single real-time observation. Table 5 presents the updated Rumuokoro Junction observation records used for the historical analysis.

S/N	Location	Timestamp	Vehicle Count
1	Rumuokoro Junction	2026-07-01 15:00	584
2	Rumuokoro Junction	2026-07-01 15:15	621
3	Rumuokoro Junction	2026-07-01 15:30	547
4	Rumuokoro Junction	2026-07-01 15:45	693
5	Rumuokoro Junction	2026-07-01 16:00	638

Table 4: Updated Historical Traffic Records for Rumuokoro Junction

The trend represented by these records is illustrated in Figure 4. The figure shows the variation in vehicle volume across the five 15-minute observation intervals.

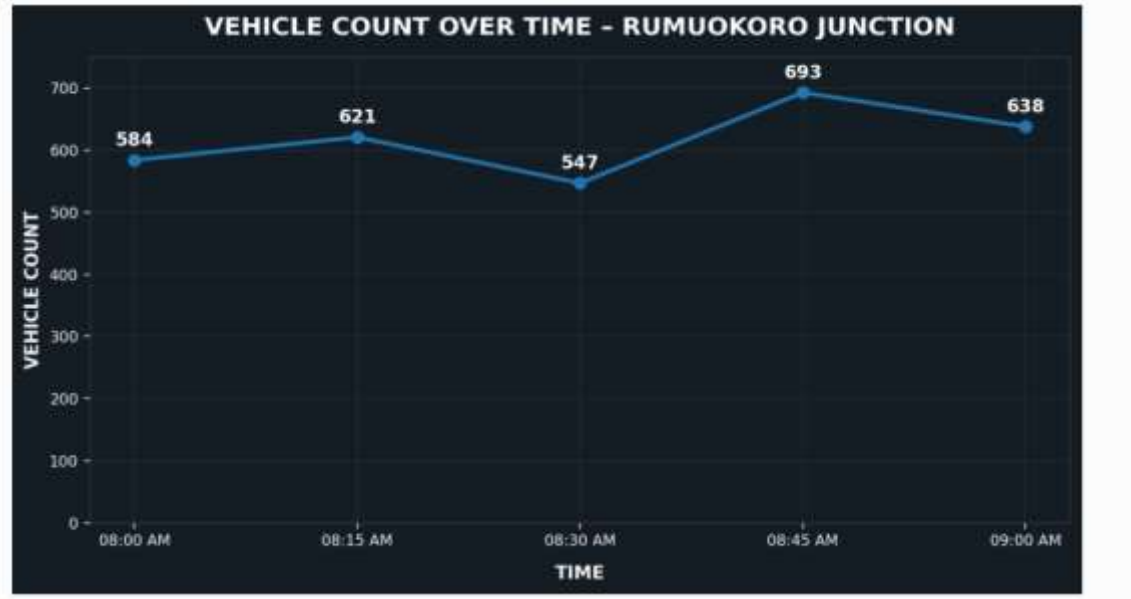

Figure 4: Historical Traffic Report Chart

4.3 Performance Evaluation

The performance results obtained during simulated testing are summarized in Table 6. The table is introduced before presentation to make the relationship between the reported metrics and the subsequent discussion explicit.

Performance Metric	Result
Detection Accuracy	97.8%
Precision	98.2%
Recall	97.3%
F1-score	97.7%
Latency	41 ms
FPS	24
MQTT Delivery	99.8%
CPU Usage	54%
Memory Usage	2.8 GB

Table 5: Performance Evaluation of the Proposed RT-SCVTMS

The results show that the prototype achieved balanced detection performance, with precision and recall values of 98.2% and 97.3%, respectively, producing an F1-score of 97.7%. The 41 ms processing latency and 24 FPS throughput indicate near-real-time operation under the tested conditions. The 99.8% MQTT delivery rate indicates reliable transmission of processed traffic metadata, while the reported CPU and memory values describe the resource conditions of the prototype during evaluation.

5. DISCUSSION

The findings demonstrate the practical value of integrating computer vision, IoT communication, and edge computing in a traffic-monitoring workflow. The system does not merely detect vehicles; it transforms detections into traffic counts, density information, congestion conditions, stored records, and dashboard reports. This integrated workflow directly addresses the research gap identified in the project, where previous local approaches were characterized by manual 15-minute counting, dependence on traffic wardens and conventional signals, or prediction based primarily on historical data.

The reported detection results are consistent with the objective of deploying AI-based vehicle recognition in a real-time traffic-monitoring context. However, the 97.8% detection accuracy in this study should not be treated as directly equivalent to mAP values reported by related studies because the evaluation measures, datasets, and testing conditions differ. This distinction is particularly important when comparing the proposed system with Edge YOLO and other object-detection studies.

The edge-oriented architecture represents an important system-level advantage. [6] highlighted the ability of edge computing to reduce communication latency and bandwidth requirements. In RT-SCVTMS, raw video is processed locally and summarized traffic information is transmitted, allowing the communication layer to carry traffic metadata rather than continuously forwarding full video streams. The project also incorporates MQTT, which is suitable for lightweight IoT messaging, together with centralized MySQL storage and a web dashboard. The historical reporting capability provides an additional advantage over isolated real-time detection demonstrations. By storing vehicle counts, density, congestion conditions, locations, and timestamps, the system creates a structured traffic history that can support future traffic analysis and planning. This directly supports the project's contribution to data-driven traffic intelligence for Port Harcourt.

6. CONCLUSION

This study developed a Real-Time Smart City Vehicular Traffic Monitoring System for Port Harcourt by integrating IoT, computer vision, artificial intelligence, edge computing, MQTT communication, database management, and web-based visualization. The system converts live traffic video into structured traffic information through local edge processing and subsequently provides traffic counts, density information, congestion status, and historical reports.

The evaluation produced 97.8% detection accuracy, 98.2% precision, 97.3% recall, 97.7% F1-score, 41 ms latency, 24 FPS throughput, and 99.8% MQTT delivery reliability. The vehicle-counting result further demonstrated the system's ability to distinguish major vehicle categories. The integrated architecture provides a practical foundation for future multi-junction deployment, adaptive traffic-signal control, incident detection, connected-vehicle integration, and broader smart-city transportation services.

REFERENCES

[1] United Nations, Department of Economic and Social Affairs. (2018). World Urbanization Prospects: 2018 Revision. United Nations.

[2] Iwo, A., & Egop, O. (2025). Average Flow in Some Major Roads in Port Harcourt.

[3] Ikezam, I., Obinna, C., & Abibo, E. (2018). Evaluation of Traffic Management Techniques on Major Roads in Port Harcourt, Rivers State, Nigeria.

[4] Li, P., Wang, H., & Xu, Y. (2020). High-resolution traffic state estimation using multi-sensor visual analytics. Transportation Research Part C: Emerging Technologies, 115, 102621.

[5] Kumar, S., Mehta, Y., & Roy, R. (2021). Computer-vision techniques for real-time vehicle detection and traffic monitoring: A survey. Journal of Transportation Technologies, 12(3), 45–63.

[6] Nguyen, T., Pham, C., & Bui, T. (2022). Edge computing for real-time traffic video analytics in smart cities. IEEE Access, 10, 88012–88025.

[7] Wan, J., Li, D., Imran, M., & Guizani, M. (2022). Edge computing-enabled video segmentation and object detection for smart transportation. IEEE Transactions on Intelligent Transportation Systems, 23(12), 20245–20258.

[8] Liang, S., Wu, H., Zhen, L., et al. (2022). Edge YOLO: Real-time intelligent object detection system based on edge-cloud cooperation in autonomous vehicles. IEEE Transactions on Intelligent Transportation Systems.

[9] Raza, M., Kazmi, M., Kidwai, H. M., Khan, H. R., Qazi, S. A., Arshad, K., & Assaleh, K. (2025). An edge-deployed real-time adaptive traffic light control system using YOLO-based vehicle detection and PCE-aware density estimation. IEEE Access.

[10] Didigwu, A. (2024). An AI-driven Approach to Traffic Congestion Prediction in Port Harcourt City Using Naïve Bayes.