Journal of Electronics and Informatics (ISSN: 2582-3825)
www.irojournals.com/iroei/

GiveWay ATES: An Adaptive Traffic Equity System Integrating PCE-Weighted Computer Vision, RFID Emergency Preemption, and Ghost Lane Detection

Vishnu Balaji C.[1], Siddharth S.[2], Hari Haran T.[3], Arockia Rubi S.[4]

[1,2,3]Student,[4]Assistant Professor, Department of Information Technology, NPR College of Engineering and Technology, Natham, Dindigul, India.

E-mail: [1]vishnubalajiit@gmail.com

Abstract

The increasing complexities of urban traffic management results in the need to develop an adaptive solution. The proposed solution integrates the following components: vehicle composition, emergency services priority, safety, and resilience. This research study proposes GiveWay ATES, an adaptive traffic equity system. The proposed algorithm is designed to integrate weighted traffic demands along with waiting time escalation in order to avoid lane starvation, while the use of RFID at hardware level makes it possible to provide pre-emptive emergency measures without depending on the backend processing. Ghost lane detection, weather-based clearance duration control and multi-source backup are also integrated in the proposed system. The experimental analysis has shown that traffic simulation, hardware-in-the-loop and field-based observation have resulted in an average delay of 28.4 s, maximal queue length of 18.9 cars, and a throughput of 2,563 PCE/h. Preemption delay was less than 200 ms, while ghost lane detection precision was around 97.3%.

Keywords: Adaptive Traffic Signal Control, Intelligent Transportation Systems, Passenger Car Equivalent (PCE), YOLOv8, Emergency Vehicle Preemption, Heterogeneous Traffic.

1. Introduction

The increasing urbanization and number of on-road vehicles have worsened the problem of traffic congestion, causing delays with increased fuel usage, pollution, and other economic

DOI: https://doi.org/10.36548/jei.2026.3.008
Extended version of article received from **International Conference on Smart Innovations in Engineering Technology (ICSIET 2026)**
Received: 20.06.2026, received in revised form: 23.07.2026, accepted: 08.08.2026, published: 19.08.2026

consequences. Texas Transportation Institute has shown that the urban congestion causes considerable economic impacts in terms of productivity loss and excessive fuel usage [1]. The problem is especially challenging in the context of Indian urban setting, where the traffic flow is diverse, consisting of motorcycles, auto-rickshaws, cars, buses, and trucks moving within one road network. The diversity of the traffic makes the traffic light management challenging due to the varying demand of each type of vehicle for road space.

The fixed time traffic signals operation does not have the capability to adapt to the changing traffic conditions due to its pre-set time phases. In terms of the vehicle actuation and vision adaptive systems, the system remains responsive to real-time traffic conditions by using the traffic sensors; however, the traffic control is only based on the number of vehicles present and it might not reflect the true nature of the traffic demand. Different vehicles would provide a wrong estimation of the traffic demand and hence it results in the inefficient utilization of green lights time.

In addition to congestion control, an adaptive traffic control system should have considerations for emergencies, safety, and availability of the system. Priority allocation to emergency vehicles needs more attention, and techniques that solely rely on a central server or communication through the network might cause unnecessary delays or become susceptible to communication failure. Moreover, the unusual events, including stalled cars and uncleared approaches, need to be identified in a timely manner, while bad weather could also result in increased signal clearance times.

Existing research studies have demonstrated the advances in vision-based vehicle detection, PCE-aware density estimation, predictive traffic modelling, and adaptive signal control [2]-[5]. However, these capabilities are frequently investigated as separate functions or evaluated predominantly within simulation-oriented environments.

To address these challenges, we have developed GiveWay ATES, an integrated and adaptive traffic management system for efficiently handling the heterogeneous intersections in urban environments. The main contributions of this research include: (1) a PCE-weighted traffic demand model that takes into consideration vehicle-class distinctions in adaptive signal scheduling; (2) a priority strategy that integrates traffic demand with waiting time escalation to avoid long-term lane starvation; (3) hardware-based RFID emergency pre-emption that runs independently from the regular backend scheduling path; (4) ghost lane detection through post-green traffic density equilibrium; and (5) weather-related clearance timing and multiple-fall-back functionality.

2. Related Works

Research on the intelligent traffic management system that uses computer vision, traffic flow prediction, machine learning, and edge IoT have made progress towards efficient vehicle detection and congestion assessment, as well as adaptive traffic light control. While much progress has been made through these developments, it still remains as a significant research problem to integrate all of these aspects into one deployable solution.

The traffic analysis system was proposed by Narayanan et al. [2], which is based on YOLOv8 and BoT-SORT [8]. The proposed system uses traffic videos collected from various junctions in Bangalore to detect and track vehicles, count turn-patterns, and forecast short-term traffic. This work has proven the capability of vision-based traffic analysis methods to be used in heterogeneous traffic conditions in India. Yet, the system did not incorporate the concept of PCE-based traffic weighting.

The RDAMVGN model framework was introduced by Nigam et al. [3], which includes an improved YOLO detection architecture and the fusion of Faster R-CNN, Mask R-CNN, and LSTM-based predictive signal timings. The researchers have concluded that spatial road occupancy is a better representation of traffic demand than vehicle counts. Thus, there is a requirement for class-aware traffic analysis. Nevertheless, the testing was conducted mostly in simulations and did not include a full physical implementation.

Raza et al. [4] discussed about an adaptive traffic light control system, which used YOLO vehicle detection, in addition to density estimation based on Passenger Car Equivalent (PCE). Raza et al. also proposed Green Denial Counter (GDC), which helps to avoid lane starvation by increasing priority of those lanes that are denied a green light phase repeatedly. However, this model serves as a good starting point for the development of PCE-weighted traffic management and lane starvation prevention. It does not use some additional components, such as hardware-based emergency pre-emption.

Wu et al. [5] developed a traffic-flow prediction framework using YOLOv8 to extract traffic-flow, density, and speed information from roadside video. The extracted parameters were subsequently processed using a hybrid machine-learning model for traffic prediction. Their work demonstrated the feasibility of combining computer vision with predictive analytics for intelligent traffic monitoring. However, the study primarily focused on traffic-flow prediction rather than closed-loop adaptive intersection control, where real-time observations directly influence the signal-phase selection and physical traffic-light operation.

In summary, the current literature reveals significant advancements in terms of vision-based vehicle detection, PCE-aware traffic estimation, predictive modeling, and adaptive traffic signal control. Nonetheless, these techniques are either considered separately or within simulation-based settings. The simultaneous combination of PCE-driven traffic analysis, emergency pre-emption at the hardware level, prevention of starvation of lanes, detection of ghost lanes, weather-dependent signal timing, and robust fallbacks has yet to be fully explored. To resolve the identified issue, GiveWay ATES proposes a new approach involving the aforementioned functionalities in a single architecture.

3. System Architecture

GiveWay ATES is designed with three key functional layers, namely, the edge perception layer for obtaining the traffic and environmental data input, traffic intelligence layer for interpreting the input data and decision making, and finally actuation layer for controlling signal at the intersection. There is also a special hardware layer used for RFID emergency pre-emption, which overrides the signals without the need for the traffic intelligence layer.

3.1 Edge Perception Layer

The ESP32-CAM is installed in Wi-Fi station mode at all the lanes. The camera captures frames in JPEG format and sends them to the Intelligence Layer using HTTP POST to /api/edge-data, which carries a JSON object containing the lane ID, base-64 encoded frame, and the timestamp. Vision inference is offloaded from the edge node to the server in such a way that the edge node does not need any GPU, which was intentionally done for reducing costs and enabling field upgrade without altering the firmware.

At the same time, RFID readers for EM-18 tags are installed in dedicated lanes designed for emergency vehicles. The readers are connected to the Arduino Mega 2560 via SoftwareSerial on analogue ports A4 (Lane 1) and A3 (Lane 2). In the case of detection of an emergency transponder, the reader sends HW_RFID:[lane]:[tagID] messages through USB serial at a baud rate of 115,200. This data flow does not pass through the Node.js server because the local firmware of the Arduino triggers ALL_R (red stop all lights) and [n]G (green lane-n light) before the server even gets an RFID event.

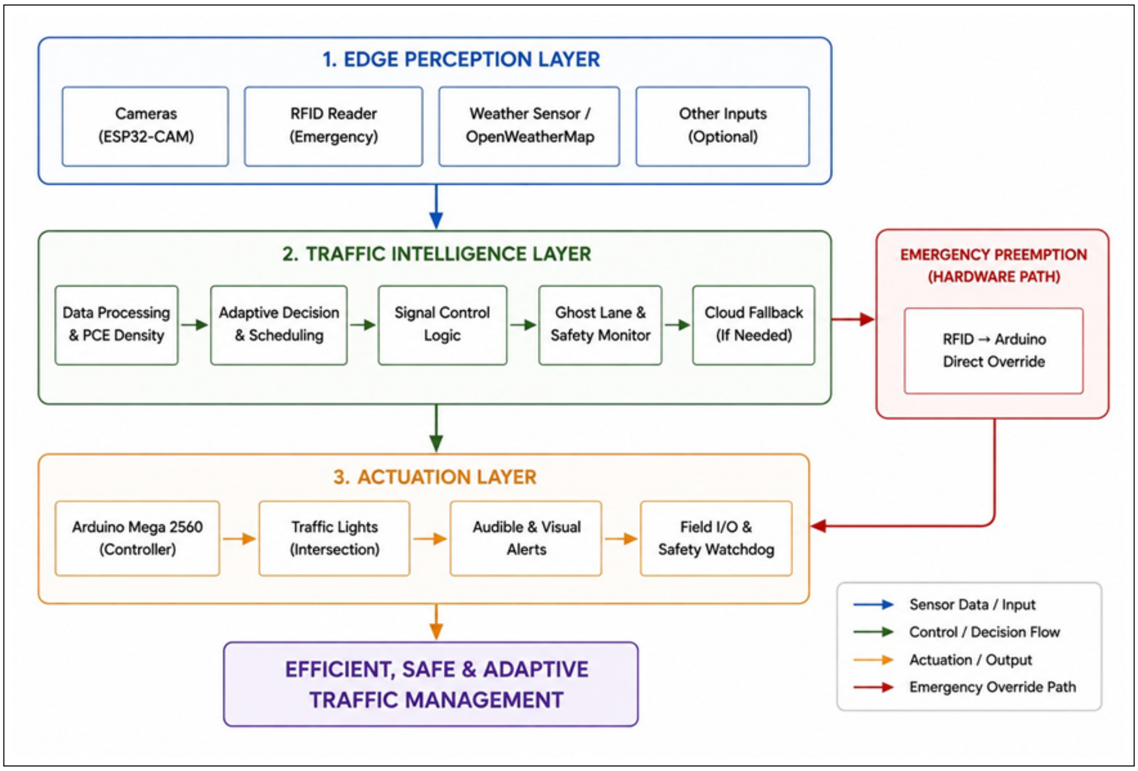

Figure 1. System architecture

3.2 Intelligence Layer — Traffic Engine

The traffic intelligence layer analyses the observations obtained from the edge devices and makes adaptive decisions regarding the traffic control. The Traffic Engine is based on Node.js 18 Express server and works in cycles of 100 ms duration. The vision pipeline sends the frames obtained by ESP32-CAM to the Python-based YOLOv8 inference engine created with the help of the Ultralytics library [6]. The resulting PCE-weighted observations are accumulated to obtain a per-lane traffic-density value.

The density information is provided to the GiveWay scheduling algorithm, which combines traffic demand, emergency priority, and waiting-time escalation to determine the next lane to receive a green phase. The calculated decision is then transmitted to the intersection controller for signal actuation.

The traffic intelligence layer also contains the Ghost Lane Detector, the Weather Adaptation Module, and the Cloud Fallback Module. The Ghost Lane Detector senses the changes in traffic density during the operation of the green signal phase and detects any abnormality if the traffic density does not change significantly from the threshold value. The Weather Adaptation Module adjusts the time for clearing the signal during adverse weather conditions.

3.3 Emergency Pre-emption Hardware Path

Emergency vehicle pre-emption works using a specialized hardware interface that is separate from the regular traffic intelligence processing system. In case the EM-18 RFID reader identifies a valid emergency transponder, the RFID event is sent directly to the Arduino Mega 2560 device. The local controller firmware will then automatically activate the emergency signals, starting with red and then proceeding with a green signal for the specific emergency vehicle.

The RFID to Arduino interface makes the system less dependent on the higher-level processing on the server. Thus, the emergency pre-emption process will continue to work even if the traffic intelligence system is not available.

4. Mathematical Model: The Giveway Algorithm

The GiveWay scheduling algorithm is evaluated at intervals of $\Delta t = 100\ ms$. For each approach lane i, the algorithm computes a PCE-weighted traffic density, incorporates emergency priority and waiting-time starvation prevention, selects the lane with maximum priority, and determines its green-phase duration.

4.1 Step 1 — PCE-Weighted Traffic Density

Let $N_{i,j}(t)$ denote the number of detected vehicles belonging to class j in lane i at time t, and let α_j denote the corresponding Passenger Car Equivalent (PCE) factor. The PCE-weighted density of lane i is

$$D_i(t) = \sum_{j=1}^{m} N_{i,j}(t)\,\alpha_j \tag{1}$$

where

$$\begin{aligned} D_i(t) &= \text{PCE-weighted traffic density of lane } i, \\ N_{i,j}(t) &= \text{number of vehicles of class } j \text{ in lane } i, \\ \alpha_j &= \text{PCE factor assigned to vehicle class } j, \\ m &= \text{total number of vehicle classes.} \end{aligned}$$

The PCE factors used in the proposed traffic-density calculation are based on the Highway Capacity Manual (HCM 2016) [7] and adapted to the heterogeneous vehicle categories considered in this study with reference to the classification approach reported by Raza et al. [4].

$$\alpha_j = \begin{cases} 0.25, & \text{motorcycle/scooter}, \\ 0.50, & \text{auto-rickshaw}, \\ 1.00, & \text{car/jeep/van/taxi}, \\ 1.50, & \text{pickup}, \\ 2.00, & \text{mini bus}, \\ 2.50, & \text{large bus}, \\ 3.00, & \text{truck/trailer}. \end{cases} \tag{2}$$

Thus, the scheduling algorithm considers the effective road-space occupancy of heterogeneous traffic rather than treating every detected vehicle equally.

For example, two large buses produce

$$D_i = 2(2.50) = 5.00PCE,$$

whereas ten motorcycles produce

$$D_i = 10(0.25) = 2.50PCE.$$

Hence, despite having fewer vehicles, the bus-dominated lane receives a higher density score.

4.2 Step 2 — Total Scheduling Priority

For every lane i, the total scheduling weight is defined as

$$W_i(t) = D_i(t) + E_i(t) + P_i(t) \tag{3}$$

where $D_i(t)$ is the PCE-weighted density, $E_i(t)$ is the emergency-preemption bonus, and $P_i(t)$ is the waiting-time penalty.

The emergency priority term is

$$E_i(t) = \begin{cases} 10000, & \text{if a valid emergency RFID tag is detected,} \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Since the emergency bonus is substantially larger than normal traffic-density and waiting-time scores, a lane containing a validated emergency vehicle receives immediate scheduling priority.

To prevent indefinite starvation of lanes with relatively low traffic density, a waiting-time penalty is introduced. Let $\tau_i(t)$ denote the continuous red-phase waiting time of lane i, measured in seconds. Then

$$P_i(t) = \begin{cases} 0, & \tau_i(t) \leq 90, \\ 1.2\tau_i(t) - 90, & \tau_i(t) > 90. \end{cases} \tag{5}$$

The exponential term progressively increases the priority of a lane after it has remained unserved for more than 90 s.

The lane selected for the next green phase is therefore

$$i^* = \arg\max_{i \in L} W_i(t) \tag{6}$$

where L denotes the set of candidate approach lanes and i^* is the selected lane.

4.3 Step 3 — Adaptive Green-Phase Duration

After selecting lane i^*, its green duration is determined from its scheduling weight using a bounded linear scaling function:

$$T_{\text{green},i^*} = \text{clamp}\left(T_0 + \frac{W_{i^*}(t)}{M}, T_{\min}, T_{\max}\right) \tag{7}$$

with

$$T_0 = 10\ s,\ T_{\min} = 10\ s,\ T_{\max} = 30\ s,$$

where $M > 0$ is a junction-specific calibration parameter.
Equivalently, the clamp operation can be written explicitly as

$$T_{\text{green},i^*} = \min\left[30,\ \max\left(10,\ 10 + \frac{W_{i^*}(t)}{M}\right)\right] \tag{8}$$

The lower bound prevents excessively short green phases, while the upper bound prevents a high-priority lane from monopolising the intersection.

4.4 Step 4 — Weather-Adaptive Clearance Timing

Let $A(t) \in \{0, 1\}$ denote the adverse-weather indicator:

$$A(t) = \begin{cases} 1, & \text{rain or snow detected,} \\ 0, & \text{normal weather.} \end{cases} \tag{9}$$

The amber duration can then be expressed compactly as

$$T_{\text{amber}}(t) = 3 + 2A(t) \tag{10}$$

and the all-red clearance duration as

$$T_{\text{all\_red}}(t) = 2 + 2A(t) \tag{11}$$

both in seconds. Therefore,

Condition	*Amber Time (s)*	*All-Red Time (s)*
Normal	*3s*	*2s*
Rain/Snow	*5s*	*4s*

5. Implementation

5.1 Hardware Configuration

GiveWay ATES uses the Arduino Mega 2560 as the hardware controller to control the signals at intersections and provide emergency responses. This controller works in coordination with the three traffic-signal groups, EM-18 RFID readers, and an alert device. The Arduino Mega 2560 represents the actuation component of the system that controls the traffic signal outputs, emergency RFID inputs, and local safety alerts. The detailed hardware connections and pin assignments are summarized in Table I.

Table 1. Arduino mega 2560 pin allocation

Component	**Pin**	**Designation**	**Mode**
Lane 1 (South)	8, 9, 10	D8, D9, D10	R/Y/G
Lane 2 (East)	11, 12, 13	D11, D12, D13	R/Y/G
Lane 3 (West)	14, 15, 16	A0, A1, A2	R/Y/G
RFID Reader 1	A4	SoftSerial RX	EM-18
RFID Reader 2	A3	SoftSerial RX	EM-18
Piezo Buzzer	22	Digital OUT	Active

Each signal cluster includes the outputs for red, yellow, and green signals, using relay or drive circuits appropriate to interface between the low-voltage microcontroller and the traffic lights themselves. RFID readers placed at specified emergency approaches act as direct inputs for the controller to allow pre-emption of traffic signals without relying on server-based processing. The audio alarm unit aids the ghost lane safety protocol with a local alarm when abnormal post-green traffic conditions occur.

5.2 Technology Stack

The technologies used for each layer of the proposed system and its deployment roles are summarized in Table 2.

Table 2. Technology stack and deployment details

Layer	Technology	Deploy	Purpose
Edge Camera	ESP32-CAM	Intersection	Image Acquisition
Emergency Detection	EM-18 RFID	Intersection	Emergency Detection
AI Vision	YOLOv8 + Ultralytics	Python	Vehicle Detection
Backend	Node.js + Express	Server	Traffic Engine
Database	MongoDB Atlas	Cloud	Audit Logs
Hardware Control	Arduino Mega 2560	Local	Signal Control
Weather	OpenWeatherMap	Cloud API	Weather Input
Traffic Fallback	TomTom Traffic Flow API	Cloud API	Fallback Traffic Input

Vehicle observations received from the edge cameras are processed through the vision pipeline and converted into PCE-weighted lane-density information for adaptive signal scheduling. Real-time communication ensures that the web dashboard and mobile applications are able to get junction state, signal state, and generate safety alerts. Persistent storage ensures retention of operations and logging information for analysis and auditing purposes. Environmental services provide external traffic and weather data to assist in environmental adaptability as well as fall-back operations when primary sensing fails.

This design ensures that updates to individual software and hardware modules can be made without making any significant changes to the entire system. The proposed system has been deployed in a cloud-based architecture, where the backend traffic engine has been deployed in Render and MongoDB Atlas to ensure the persistent storage of operations and audit logs. Source code and implementation materials have been provided publicly in the GiveWay ATES repository [10].

6. Experimental Results and Evaluation

The proposed system was tested using complementary traffic simulation, hardware-in-the-loop testing, and field observation. The areas that were evaluated include traffic efficiency, emergency response capability, ghost lane detection, weather-based operation, and impacts of system enhancement.

6.1 Traffic Control Performance

Two base-case traffic signal control algorithms were also considered for comparative analysis. Fixed Time Control (FTC), where pre-set time intervals are employed, and Vehicle Actuated Control (VAC), where the traffic signals operate depending on detected traffic volume. This base case scenario was tested in identical experimental traffic scenarios to those employed for the proposed GiveWay ATES system.

As shown in Table 3, the proposed strategy outperforms the base-case fixed-time and vehicle count-based traffic signal control algorithms in terms of traffic management performance. This is mainly due to the application of PCE-based traffic evaluation instead of mere vehicle counts.

Table 3. Performance comparison across signal control strategies

Metric	**Fixed-Time Control (FTC)**	**Vehicle-Actuated Control (VAC)**	**GiveWay**
Mean Delay (s)	47.2	38.6	28.4
Max Queue (veh.)	34.1	27.3	18.9
PCE Throughput (PCE/hr)	1,824	2,107	2,563
Emergency Latency (ms)	N/A	N/A	< 200
Ghost Lane Acc. (%)	—	—	97.3
Amber Extension	Fixed 3 s	Fixed 3 s	5 s (rain/snow)

In heterogeneous traffic environment, vehicle counts do not really indicate the effective demand on the intersection, as each class of vehicle uses road space in its own way. As a result, GiveWay ATES makes use of PCE criteria while scheduling the traffic signals in order to prioritize based on weighted demand. It enables the controller to differentiate, say, between a lane with predominantly two-wheelers from another with bigger buses/freight trucks.

The resulting reduction in queue formation and delay indicates that vehicle-class-aware density estimation can provide a more representative basis for adaptive signal allocation in mixed urban traffic.

Table 4. Per-version performance benchmark (GiveWay ATES)

Metric	**v4.0**	**v4.5**	**v4.7**
Mean Delay (s)	36.1	30.7	28.4
Ghost Lane Detect.	—	94.1%	97.3%
Preempt. Latency	< 200 ms	< 200 ms	< 200 ms
WS Throughput (msg/s)	42	48	67
Fallback Active	RFID only	RFID + Weather	Full 3-tier

Table 4 includes an internal version-based assessment of the GiveWay ATES prototype system. The versions v4.0, v4.5, and v4.7 are development stages of the same proposed system rather than different external models. Version v4.0 refers to the first implementation with adaptive control and hardware integration, version v4.5 has been developed with additional safety features and weather adaptation, while v4.7 refers to the latest implementation with improved communication support from various sources.

The observed progression suggests that the system benefits not only from improvements to the core scheduling algorithm but also from the integration of supporting mechanisms for safety, communication, and operational resilience. This modular evolution also enables individual features to be evaluated before their incorporation into the complete architecture.

6.2 Emergency Pre-emption

Hardware-in-the-loop testing was conducted to verify the reliability and responsiveness of RFID-based emergency pre-emption. It has been shown that the emergency path does not depend on any normal backend traffic processing activity. Such independence is achieved by implementing the pre-emption decision logic at the microcontroller level. After detecting an RFID emergency tag, the controller can perform the necessary switching operation without waiting for a full cycle of sensing, communication, and scheduling on the backend side. Thus, the proposed architecture decreases dependence on the availability of the network and backend processing in emergency cases. Moreover, stress tests have shown that the hardware pre-emption algorithm works correctly when there is an overload of the backend.

6.3 Ghost Lane Detection

The ghost-lane mechanism was analysed in several obstructive situations such as presence of parked cars and improperly cleared lanes after green time. The results, presented in Table III, show that the density-stasis detection is a useful lightweight method to detect the presence of the ghost-lane. This technique utilizes the traffic density information already present in the perception pipeline and does not require the accident detection module. If there is no change in the density during active green time and the density value exceeds a predefined threshold, it can be considered as an obstruction. It should be noted that there is a trade-off in setting the threshold since shorter periods increase the ability to detect short obstructions, however, at the same time, they may trigger false alarms due to slow clearing of the queue.

6.4 Weather-Adaptive Signal Operation

The concept of weather-sensitive timing was explored to check whether the intervals of signal clearance need modification in reduced road friction. It was found that adverse weather leads to increased stopping requirements for the approaching vehicles. In light of this finding, the system changes the duration of amber and all red intervals in case of presence of adverse weather. Instead of changing the lane priority calculation based on the PCE value, this method is applied directly to transition intervals. With this separation between lane priority and transition time, traffic requirement determines the priority of lanes whereas weather conditions determine the safety time.

6.5 Limitations

Even though GiveWay ATES system is performing well, there are certain limitations of the present study which need to be noted:

- The system has been evaluated under limited conditions of traffic and intersections; thus, the performance of the system can differ depending upon the nature of intersection, traffic density, and driving patterns.
- Vehicle detection and PCE estimation can get affected due to occlusion, poor lighting, weather conditions, heavy traffic, and camera positions.

- Estimated values of PCE might not accurately define road-space occupation due to variable traffic and roadway conditions.
- Wireless communication of ESP32-CAM nodes may include delays and packet losses.
- Ghost-lane detection depends upon traffic-density stasis and it might fail to differentiate between the incident and clearing traffic queues.
- The current study focuses only on single-junction operations.

7. Conclusion

This research study has designed and implemented GiveWay ATES, an adaptive traffic management framework to handle with heterogeneous traffic demands, prioritize emergencies, ensure operational safety, and resilience. The proposed architecture implemented PCE-based weighted traffic evaluation to model vehicle class-specific road space requirements and employed adaptive scheduling along with waiting time escalation for minimizing lane starvation. An independent method of pre-emption through hardware-level RFID ensured the prioritization of emergency vehicles whereas ghost lane monitoring, weather-based clearance time, and multi-channel fallback made the architecture to go beyond traditional traffic management. This study has also experimentally evaluated the feasibility and advantages of the integrated architecture and compared it with traditional signal control techniques. Nevertheless, a more generalized validation is needed for testing the model under different intersection geometry and traffic situations. Future research will focus on enabling multi-intersection coordination, scheduling parameter optimization, improving multimodal sensing, and V2I communication for traffic management.

References

[1] Texas A&M Transportation Institute, "2023 Urban Mobility Report," Texas A&M University, College Station, TX, USA, Tech. Rep., 2023. [Online]. Available: https://mobility.tamu.edu/umr/

[2] Narayanan, Sundarakrishnan, Sohan Varier, Tarun Bhupathi, Manaswini Simhadri Kavali, P. Ramakanth Kumar, and K. Sreelakshmi. "Vehicle Turn Pattern Counting and Short Term Forecasting using Deep Learning for Urban Traffic Management System." IEEE Access 2025, vol. 13, 8585-8593.

[3] Nigam, Nikhil, Dhirendra Pratap Singh, Jaytrilok Choudhary, and Surendra Solanki. "An Efficient Model for Real-Time Traffic Density Analysis and Management using Visual Graph Networks." IEEE Access 2025, vol. 13, 140413-140439.

[4] Raza, Mehwish, Majida Kazmi, Hamza Munir Kidwai, Hashim Raza Khan, Saad Ahmed Qazi, Kamran Arshad, and Khaled Assaleh. "An Edge-Deployed Real-Time Adaptive Traffic Light Control System using YOLO-based Vehicle Detection and PCE-Aware Density Estimation." IEEE Access 2025, vol. 13, 153586-153613.

[5] Wu, Fukui, Hanzhong Tan, Linfeng Zhang, Shuangbing Wen, and Tao Hu. "Multivariate Machine Learning Model based on YOLOv8 for Traffic Flow Prediction in Intelligent Transportation Systems." IEEE Access 2025, vol. 13, 105091-105100.

[6] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLO (Version 8.0.0)," Ultralytics, 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[7] Transportation Research Board, Highway Capacity Manual: A Guide for Multimodal Mobility Analysis (HCM 2016), 6th edition Washington, DC, USA: National Academies of Sciences, Engineering, and Medicine, 2016.

[8] N. Aharon, R. Orfaig, and B.-Z. Bobrovsky, "BoT-SORT: Robust Associations Multi-Pedestrian Tracking," 2022, arXiv: 2206.14651.

[9] OpenWeatherMap, "Current Weather Data API," OpenWeather Ltd., 2024. [Online]. Available: https://openweathermap.org/current

[10] Vishnubalaji-C, "GiveWay ATES Source Repository (v4.7)," GitHub, 2026. [Online]. Available: https://github.com/Vishnu