International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 05 May 2025
Page No: 2234 - 2240
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0328

Lane-Wise Traffic Intelligence Using Deep Vision Systems for Signal Optimization

E Pavan Kumar[1], Harshitha C[2], A Bhavishya [3], Gattu Pranava Kumar Reddy[4], Mr. Saida[5]
[1,2,3,4]UG Scholar, Dept. of CSE-AIML, Sphoorthy Engineering College, Hyderabad, Telangana, India.
[5]Assistant professor, Dept. of CSE-AIML, Sphoorthy Engineering College, Hyderabad, Telangana, India.
Emails: *pavanpavan56108@gmail.com[1], harshithachavan620@gmail.com[2], annepallybhavishya@gmail.com[3], gpranavareddy@gmail.com[4], bhukyasaidanaik@gmail.com[5]*

Abstract
Urban traffic congestion remains a critical challenge affecting commute times, fuel efficiency, and air quality. This project presents a data-driven approach to traffic flow optimization by dynamically adjusting traffic signal timings based on real-time vehicle density across multiple lanes. Utilizing computer vision techniques, such as YOLO-based vehicle detection, the system captures live video feeds from intersections to estimate vehicle count per lane. The signal timings are then optimized to prioritize lanes with higher traffic density, thereby reducing overall waiting times and improving traffic throughput. Experimental results from a simulated environment demonstrate significant improvements in traffic flow efficiency and reduced signal idle times. The proposed solution offers a scalable and adaptive framework for smart city traffic management systems.

Keywords: *Deep learning; Dynamic traffic signal control; Lane-wise vehicle counting; Vehicle detection; YOLOv4.*

1. Introduction
Urban traffic congestion has become a significant problem in modern cities, with implications for both commuters and the environment. Traditional traffic signal systems, often based on fixed time intervals, struggle to respond to fluctuating traffic conditions, leading to inefficiencies such as long waiting times and increased fuel consumption. The growing urban population and an increasing number of vehicles on the road exacerbate these challenges, making the need for more adaptive traffic management systems even more pressing. Recent advances in computer vision and machine learning have provided innovative solutions to these challenges. Technologies such as real-time vehicle detection using deep learning models are proving effective in monitoring traffic conditions dynamically. By leveraging computer vision, it is possible to identify vehicle density at intersections and adjust signal timings accordingly. This adaptive approach promises to not only improve traffic flow but also reduce congestion and the environmental impact of urban transportation systems. This work aims to develop a system that dynamically adjusts traffic signal timings based on the real-time vehicle count in each lane using YOLO-based vehicle detection. The proposed system is designed to optimize traffic flow by giving priority to lanes with higher vehicle density, thus minimizing waiting times and improving overall intersection throughput. The integration of computer vision into traffic signal control offers a significant step forward in the development of smart city infrastructure, enhancing the efficiency and sustainability of urban transportation.[1]

2. Methods
The objective of this project is to optimize traffic flow at intersections by dynamically adjusting traffic signal timings based on real-time vehicle detection. This approach uses computer vision and web-based technologies to evaluate vehicle density in each lane, adjusting signal timings accordingly. The following describes the methods used to implement this system.

2.1. Vehicle Detection Using YOLOv4
YOLOv4 (You Only Look Once version 4) is

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 05 May 2025
Page No: 2234 - 2240
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0328

employed for object detection to identify vehicles in real-time. The pre-trained YOLOv4 model is loaded using OpenCV's DNN module, which allows for seamless execution of the model. Each video frame captured from the intersection is resized to 416×416 pixels and converted into a format compatible with the YOLOv4 model. The model identifies vehicle classes such as cars, motorcycles, buses, and trucks. A confidence threshold of 50% is set to ensure that only accurate detections are considered. Non-Maximum Suppression (NMS) is applied to remove redundant bounding boxes, ensuring that each vehicle is counted once.

2.2.Lane-Specific Frame Analysis

The system processes video frames captured from the intersection, assigning each frame to a specific lane for analysis. The vehicle count for each lane is updated based on the number of vehicles detected in the frame. Each vehicle detected is highlighted with a bounding box, and the annotated frame is saved locally for display on the web interface. This lane-specific analysis forms the foundation for adjusting the signal timing dynamically based on vehicle density. [3]

2.3.Dynamic Traffic Signal Timing Adjustment

The signal timings for each lane are adjusted based on the number of vehicles detected in that lane. A predefined total cycle time (e.g., 60 seconds) is distributed proportionally to the lanes based on vehicle count. If a lane has no detected vehicles, it receives a default green time of 10 seconds. Lanes with higher vehicle counts are allocated longer green light durations to prioritize traffic flow in busy lanes. This ensures that each lane receives an appropriate signal duration based on real-time traffic conditions.

2.4.Signal Control Logic for Cyclical Operation

The traffic signal operates in a cyclic manner. Each lane is assigned the green signal for a duration based on its vehicle count, while the other lanes remain red. A countdown timer is displayed for each lane during its green signal phase to mimic real-world signal behaviour. Once the green signal for a lane expires, the signal switches to the next lane in the cycle, ensuring continuous traffic flow across all lanes. This cycle repeats, adapting dynamically to traffic changes.

2.5.Web Interface for User Interaction

A Flask-based web interface is created to allow users to control and monitor the system. The interface includes:

- **Start/Stop Detection:** Users can initiate or halt the vehicle detection process using the /start detection and /stop detection API endpoints.
- **Real-Time Lane Data:** The interface displays the number of vehicles detected in each lane, visual representations of the lanes with vehicles, and the current signal timings.
- This simple yet effective interface enables real-time monitoring and control of the traffic signal system, providing users with valuable insights into traffic flow.

2.6.Multithreading for Efficient Execution

To ensure smooth operation, the system uses Python's threading module to handle multiple tasks concurrently. Vehicle detection and traffic signal control processes are executed in parallel, which prevents delays and ensures that the web interface remains responsive. Multithreading enables the system to continuously detect vehicles and adjust signal timings in real time, without interruptions. (Table 1)

Table 1 System Parameters

Parameter	Value
YOLO Confidence Threshold	0.5
Maximum Green Time	60
Input Image Resolution	416 x 416
YOLO Model Used	YOLOv4

Table 1. provides key parameters used in the traffic signal optimization system based on the YOLOv4 model. The maximum green time is capped at 60 seconds to prevent excessive delays for other lanes. A confidence threshold of 0.5 ensures only reliable vehicle detections are considered for decision-making. The input image resolution is set to 416 × 416, balancing processing speed and detection accuracy. YOLOv4 is selected as the detection model due to its proven real-time performance and high accuracy in identifying various vehicle types across traffic surveillance scenarios. [2]

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 05 May 2025
Page No: 2234 - 2240
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0328

Table 2 Vehicle Count per Lane (Sample Run)

Lane Number	Vehicle Count	Captured Image Name
Lane 1	7	Lane_1_Output.jpg
Lane 2	8	Lane_2_Output.jpg
Lane 3	7	Lane_3_Output.jpg
Lane 4	6	Lane_4_Output.jpg

(Table 2) illustrates a snapshot of vehicle detection results across multiple lanes during system operation. The entries indicate the number of vehicles identified per lane using YOLOv4, alongside the classification information based on object type. This real-time data is crucial for determining how long each traffic signal should remain green or red, optimizing flow according to congestion in each lane. (Figure 1) [4]

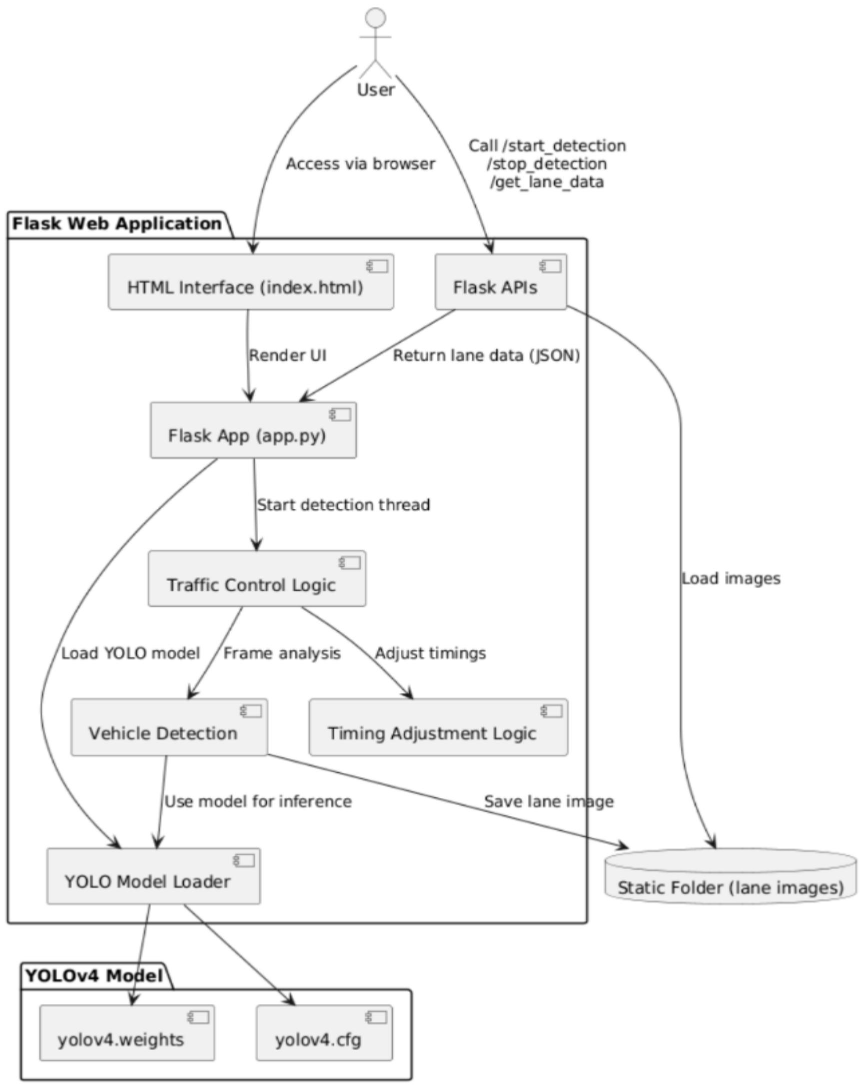

Figure 1 System Architecture Diagram

The system architecture illustrates the overall workflow of a dynamic traffic signal control application integrated with vehicle detection using YOLOv4 and a Flask-based web interface. The process begins with the user interacting with the system through a browser. The HTML interface (index.html) serves as the user interface, while a set of RESTful Flask APIs handle commands such as starting or stopping detection and fetching lane-specific data. (Figure 1) At the core of the application is the main Flask script (app.py), which serves as the control centre. Upon receiving user requests, it initiates a detection thread that invokes the traffic control logic. This logic is responsible for analysing input frames, managing the vehicle detection module, and dynamically adjusting traffic signal timings based on vehicle density. The vehicle detection module uses a YOLOv4 model for identifying and classifying vehicles in each lane. The model configuration and weights (yolov4.cfg and yolov4.weights) are loaded using a dedicated loader module. The detection results are then used to calculate traffic density per lane, which informs the timing adjustment logic. This logic computes the optimal duration for green lights based on real-time vehicle counts. Processed images, along with updated lane information, are saved to a static folder, making them accessible for live display on the web interface. This enables real-time monitoring of both lane images and corresponding traffic signal durations, delivering a seamless and interactive experience for users. In addition to real-time traffic regulation, the modular architecture of the system allows for easy scalability and future enhancements. Since the vehicle detection and timing adjustment components are decoupled from the web interface, improvements such as integrating advanced predictive analytics, incorporating different detection models, or adding support for more traffic parameters can be implemented with minimal changes to the existing codebase. This separation of concerns also aids in testing and maintenance, ensuring that each module can be updated or debugged independently without affecting the overall system functionality. Figure 2 iillustrates the output generated by the YOLOv4-based vehicle detection system for a four-lane traffic setup. The lanes are clearly identified with labels A, B, C, and D, allowing for straightforward differentiation of traffic direction and vehicle distribution across the monitored road section. Although detected vehicles are enclosed within green bounding boxes to indicate recognition by the model, the labels shown in the figure are associated with the respective lanes, not with individual vehicles. (Figure 2) [5] Internally, each detected object is assigned a class ID based on the COCO dataset, but these IDs are not visually displayed to maintain simplicity. This lane-wise identification approach enables precise monitoring of traffic in each lane, which is crucial for accurate vehicle counting and assessing congestion.

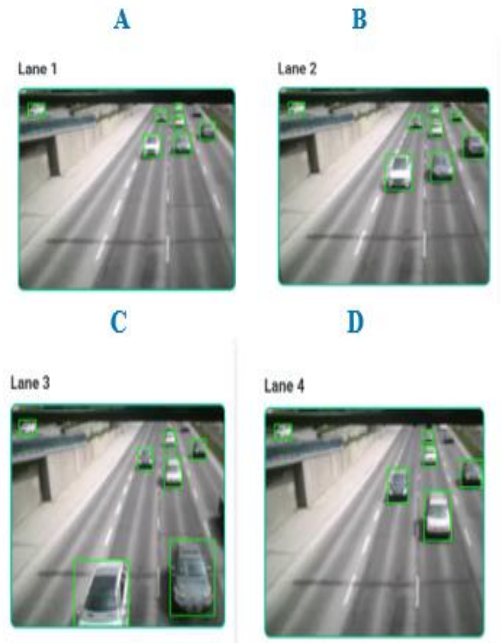

Figure 2 Sample Vehicle Detection Output

By linking each lane to its computed vehicle count, the system can adapt traffic signal durations in real time, ensuring smoother flow and minimizing delays at intersections. This output serves as vital input for the traffic control logic, which uses detection results to optimize signal switching and reduce congestion under varying traffic conditions. (Figure 3) [6] The process begins with the capturing of a frame from a live video feed. This frame is then passed to a decision point where the system checks whether any vehicles are detected. If no vehicles are detected in the frame, the system simply waits for the next frame to continue checking, ensuring unnecessary processing is avoided. If vehicles are detected, the system proceeds to process the image using a vehicle detection model (like YOLO). The image is analysed to identify and classify vehicles such as cars, buses, or trucks. After the image is processed, the system moves to the next step: counting the number of vehicles present in the detected regions. Once vehicle counts are determined, the system then adjusts the signal timings based on traffic density. Lanes with more vehicles are allocated longer green light durations to ease congestion, while less crowded lanes receive shorter durations After adjusting the

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 05 May 2025
Page No: 2234 - 2240
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0328

timings, the system reaches a decision point where it evaluates whether the new signal timings are ready to be implemented. If they are, the system proceeds to update the traffic signal timings accordingly, reflecting the changes in the real-world traffic lights.

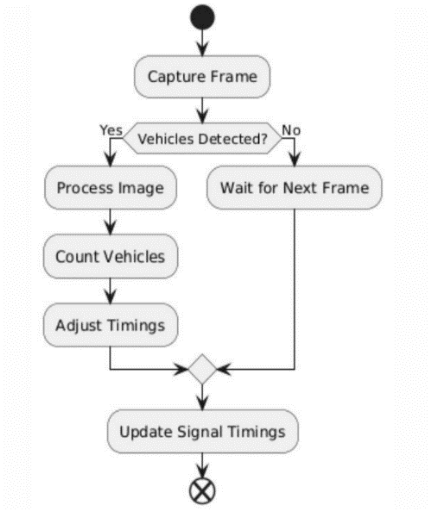

Figure 3 Traffic Signal Control Flow

This control flow then loops back, allowing the system to continuously capture new frames, analyse traffic conditions, and update signal timings in real time, thereby enabling an adaptive and intelligent traffic control mechanism. [7-8]

3. Results and Discussion

3.1. Results

The final output of this project presents a highly effective vehicle detection and dynamic traffic signal control system, designed to optimize traffic flow by adjusting traffic signal timings based on real-time vehicle detection. The system uses YOLOv4, a state-of-the-art deep learning model, to detect and classify vehicles such as cars, motorcycles, trucks, and buses. Each detected vehicle is counted in real-time and the traffic light timings are adjusted dynamically based on the number of vehicles in each lane. In the system, each lane is monitored using a video feed, and the YOLOv4 model processes every frame to detect vehicles. A confidence threshold of 0.5 is applied to filter out false positives, ensuring that only confident detections contribute to the vehicle count. The system counts the number of vehicles in each lane, which then informs how long the green light for that lane should remain active. If a lane has more vehicles, the green light duration increases to accommodate the traffic load. Conversely, lanes with fewer vehicles receive shorter green light durations, optimizing the intersection's throughput and reducing overall waiting times. The dynamic signal control works by allocating green light times proportionally to the vehicle count in each lane. For instance, if Lane 1 detects a higher vehicle count than Lane 2, Lane 1 will receive a longer green signal. The maximum green time allocated for any lane is capped at 60 seconds, ensuring efficient traffic flow. Additionally, if a lane detects no vehicles, the system defaults to a base green time of 10 seconds, maintaining minimal delay for all lanes. The system also includes a visual component where the output frames, showing vehicle detection and bounding boxes, are saved and available for review. This not only provides confirmation of the system's functionality but also helps in tracking the vehicle flow and signal timing. The integration of this output with a web interface allows for real-time monitoring, showing lane statuses, vehicle counts, and dynamic signal timings.

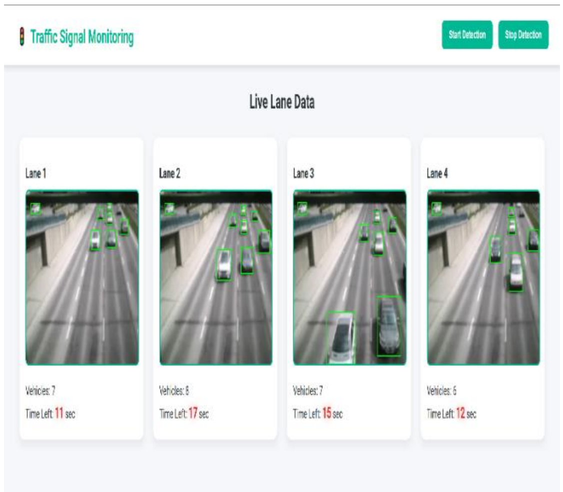

Figure 4 Traffic Signal Control Flow

In essence, the final output showcases a dynamic, real-time vehicle detection system that effectively adjusts traffic signal timings to improve traffic flow at intersections. Through its integration of advanced vehicle detection techniques, smart traffic light control, and automated response mechanisms, the system ensures optimal traffic management, reducing congestion and improving the overall efficiency of traffic flow. [9]

IRJAEH
International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 05 May 2025
Page No: 2234 - 2240
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0328

3.2.Discussion

This system's approach to dynamic traffic signal control demonstrates how real-time vehicle detection can substantially optimize traffic flow. By using YOLOv4 for vehicle detection, the system efficiently counts vehicles in each lane, and dynamically adjusts the green signal durations based on these counts. This flexibility allows for green light durations to be extended for lanes with heavier traffic, reducing waiting times and enhancing the throughput of vehicles. Conversely, lanes with fewer vehicles receive shorter green signals, preventing delays caused by unnecessarily long signal times. A significant advantage of this system is its adaptability. Unlike traditional traffic signal systems, which rely on fixed signal timings, this dynamic system responds to changing traffic conditions. By continuously monitoring the number of vehicles in each lane, the system is capable of adjusting signal timings in real time. This not only reduces congestion but also ensures that green lights are allocated proportionally to the traffic load, preventing underutilization of the green signal and minimizing delays at intersections. Moreover, the system's ability to revert to a base green time of 10 seconds when no vehicles are detected ensures that idle time is minimized. This minimizes unnecessary delays for all lanes, contributing to a more efficient intersection. The maximum green time of 60 seconds prevents one lane from monopolizing the signal and guarantees fair distribution of the green light across all lanes, even in situations where one lane experiences prolonged vehicle congestion. The system also provides valuable visual feedback. The output frames, including detected vehicles and bounding boxes, are saved for further review. This output not only verifies the system's functionality but also provides an important tool for tracking traffic patterns and signal timings over time. The integration of this output with a web interface enables real-time monitoring, showing live lane statuses, vehicle counts, and signal timings, which enhances both operational transparency and traffic management efficiency. This system represents a significant step forward in smart traffic control, with its ability to dynamically adjust signals based on real-time traffic conditions. While the system is already efficient in reducing congestion and improving traffic flow, there is potential for further improvements. Future enhancements could involve integrating predictive models based on historical traffic data to proactively manage traffic signals. Additionally, expanding the system to monitor and control more complex traffic scenarios, such as intersections with more lanes or multiple directions of traffic, would further enhance its functionality. [10]

Conclusion

This project demonstrates the effective deployment of an adaptive traffic signal control mechanism that utilizes the YOLOv4 object detection framework for real-time vehicle recognition. The system is designed to dynamically adjust traffic signal durations by analysing vehicle density across multiple lanes, thereby enhancing intersection throughput. As observed from the results, signal timings are automatically calibrated according to the number of vehicles detected in each lane, ensuring that heavily congested lanes receive longer green light durations while less crowded ones are allocated shorter intervals. Leveraging YOLOv4's high detection precision, the system accurately identifies a wide range of vehicle classes—including cars, buses, motorcycles, and trucks—which aids in a more comprehensive traffic analysis. By continuously monitoring vehicle flow, the solution intelligently adapts signal phases to actual traffic behaviour, significantly improving intersection performance and minimizing idle time for drivers. The observed performance indicates a promising outlook for deploying such systems in urban environments where dynamic traffic management is essential. This intelligent control model provides a scalable, cost-effective alternative to traditional fixed-time traffic signals, particularly beneficial during rush hours or at high-traffic intersections. Overall, the project introduces a forward-thinking approach to smart city traffic infrastructure. Furthermore, the integration of this system with a web-based interface enhances usability by providing real-time visualization of lane-wise traffic data, signal timings, and detection outputs. This feature not only facilitates seamless monitoring and control but also supports future expansion by allowing remote access and centralized traffic coordination. With its modular design and reliance on open-source tools, the system offers strong potential for deployment in various smart city frameworks and can be extended to incorporate additional features such as emergency vehicle detection or pedestrian prioritization.

Acknowledgements

We, the authors, would like to express our sincere gratitude to the developers and maintainers of the open-source frameworks and datasets that were pivotal in this work. The YOLOv4 model and COCO dataset played an essential role in enabling accurate vehicle detection for our traffic control system. We are also deeply thankful for the ongoing support and guidance from our faculty throughout the development of this project. This work relied solely on open-source tools such as Python and OpenCV for its implementation, testing, and analysis.

References

[1]. Kachroo, P., & Samuel, M. K. (2006). Traffic Signal Control: A Review. IEEE Transactions on Intelligent Transportation Systems, 7(3), 310-321
[2]. Ye, X., Ren, S., & Sun, J. (2018). Real-time Traffic Light Control based on Efficient Object Detection. arXiv preprint arXiv:1803.08109.
[3]. Zeadally, S., Younis, M. F., & Alomainy, A. (2018). Traffic Signal Control in Smart Cities: Challenges and Opportunities. IEEE Access, 6, 44479-44488.
[4]. Gustafsson, C., & Andreasson, H. (2012). Traffic Light State Estimation at Intersections for Traffic Flow Optimization. IEEE Transactions on Intelligent Transportation Systems, 13(3), 1154-1166.
[5]. Lu, S., & Cham, W. K. (2014). A Survey of Computer Vision for Traffic Scene Understanding. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 44(7), 917-935.
[6]. Ding, K., Wang, K., Zhang, W., et al. (2016). Real-time Traffic Light Recognition and Detection in Urban Environment. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA).
[7]. Ding, K., Wang, K., Zhang, W., et al. (2016). Real-time Traffic Light Recognition and Detection in Urban Environment. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA).
[8]. El-Tantawy, S. A., Elhenawy, M. I., & Abbas, M. M. (2017). Reinforcement Learning-Based Adaptive Traffic Signal Control: State of the Art and Future Directions. Transportation Research Part C: Emerging Technologies, 79, 276-292.
[9]. Liu, H., Zheng, J., & Shladover, S. E. (2017). Vehicle-to-Infrastructure (V2I) Communication for Traffic Signal Control: A Review. IEEE Transactions on Intelligent Transportation Systems, 18(11), 2977-2998.
[10]. Li, C., Wei, Y., & Wang, H. (2019). Decentralized Traffic Signal Control System Using Blockchain. In Proceedings of the IEEE International Conference on Blockchain (Blockchain), 278283.