engineering proceedings

MDPI

Proceeding Paper

DroneDeep RL (DDR): A Traffic Congestion Control Strategy Using Prioritization LLM Agent and Circular Deep Q-Network †

Md. Mujahid Hasan [1], Afsana Siddika [1], Maria Akter Khushi [1], Salman Md Sultan [2], Tahira Alam [1,*] and Shajedul Hasan Arman [3]

[1] Department of Computer Science and Engineering, University of Asia Pacific, Dhaka 1605, Bangladesh; 21201126@uap-bd.edu (M.M.H.); 21201122@uap-bd.edu (A.S.); 21201107@uap-bd.edu (M.A.K.)
[2] Innovative Skills Ltd., Dhaka 5042, Bangladesh; salmanmdsultan92@gmail.com
[3] Department of CSE, University of Central Florida, Orlando, FL 32816, USA; sh400274@ufc.edu
* Correspondence: tahira.cse@uap-bd.edu; Tel.: +880-1914876034
† Presented at the 7th Eurasia Conference on Biomedical Engineering, Healthcare and Sustainability 2025 (ECBIOS 2025), Kaohsiung, Taiwan, 23–25 October 2025.

Abstract

Traffic congestion is a problem in urban traffic that needs to be monitored and managed intelligently. In this study, a hybrid traffic management system is designed based on a combination of drone vision, large language model (LLM) inferences, and deep reinforcement learning (DRL). Using drones videos of real-time traffic, the lightweight You Only Look Once v11 model detects vehicles, and after, traffic flow levels are identified by the proposed LLM agent. A Circular-Deep Q-Networks-based DRL controller is proposed to reduce the average waiting time of vehicles. Simulation experiments validate improved congestion detection, reduced delay, and more effective communication for smart city traffic control.

Keywords: traffic management; large language model (LLM); deep reinforcement learning; smart cities

1. Introduction

Traffic congestion (TC) is a common problem of modern urban life, leading to delays, wasted fuel, and pollution. Many researchers are utilizing various technologies (i.e., IoT-based sensor systems, vision-based technology, and so forth) to reduce TC in urban areas [1,2]. Moreover, accurately predicting traffic flow in high-density areas is crucial for reducing vehicle delay time [3]. IoT-based sensor systems provide a new way to estimate the real-time traffic flow. Despite their effectiveness, non-image IoT sensor systems face accuracy-based challenges during traffic flow estimation [4,5].

To overcome these challenges, drone-based vision systems are used for complementary solutions. Drones with high-resolution cameras and computer vision programs detect and calculate real-time traffic data without needing a lot of ground infrastructure [6]. However, drones have limited processing power and battery life. Therefore, a light version of You Look Only Once (YOLO) v11 is needed to achieve real-time detection without draining hardware resources [7,8]. Furthermore, dynamic traffic signal control is one way to prevent TC in uncertain urban environments, as it can adaptively adjust timing based on real-time traffic conditions [9–12].

Deep reinforcement learning (DRL) enables an agent to learn optimal signal policies through continuous interaction with the environment, thereby reducing waiting time and improving overall traffic throughput [10]. However, existing centralized DRL-based traffic

Check for updates

Academic Editors: Teen-Hang Meen, Cheng-Yi Chen and Cheng-Fu Yang

Published: 16 April 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

control systems do not explicitly consider prioritized traffic flow, which usually leads to higher waiting times and ultimately worsens TC [13,14]. Furthermore, the existing Deep Q-Network (DQN)-based deep RL method [15] uses deque-based experience replay to memorize experiences. However, deque-based memory takes more time to store experiences, which causes higher waiting times in the traffic control system.

To overcome the issue of prioritizing traffic flow and deque memory, we introduce a prioritization large language model (LLM) agent and circular memory in DQN, in this paper. The developed LLM is responsible for identifying the level (i.e., high, mid, low) of traffic flow based on traffic data by Yolo11 [8,16]. The system does not allow low-level traffic flow to pass as an input to the proposed circular-DQN layer. As a result, the latency of the overall system is reduced significantly compared to existing methods. Moreover, the circular memory stores experiences faster than the existing deque memory in DQN. After employing circular-DQN, the overall waiting time is reduced significantly compared to existing Deque-DQN methods.

The remainder of this article is organized as follows. Sections 2 and 3 describe our proposed methodology and simulation results, respectively. Finally, Section 4 presents the conclusion of this research work.

2. Proposed Methodology

2.1. System Overview

Figure 1 illustrates the complete architecture of the drone-based traffic management system. In this proposed system, we use drones to capture aerial traffic data over intersections along roads. These captured aerial traffic data are transmitted to a lightweight YOLOv11 model, which detect and count vehicles to estimate the traffic flow of each area, which is the purpose of the developed system. Then, the value of traffic flow is subsequently fed into the proposed LLM agent to rank the traffic, listing the most congested locations and identifying these as being in a high-priority state of need. In the final layer, we use a circular-DQN-based DRL controller that decides which traffic lights should turn green based on collaborative training. Overall, the system aims to minimize the mean waiting time of cars, reduce TC at intersections, and enable smart, prioritized traffic management.

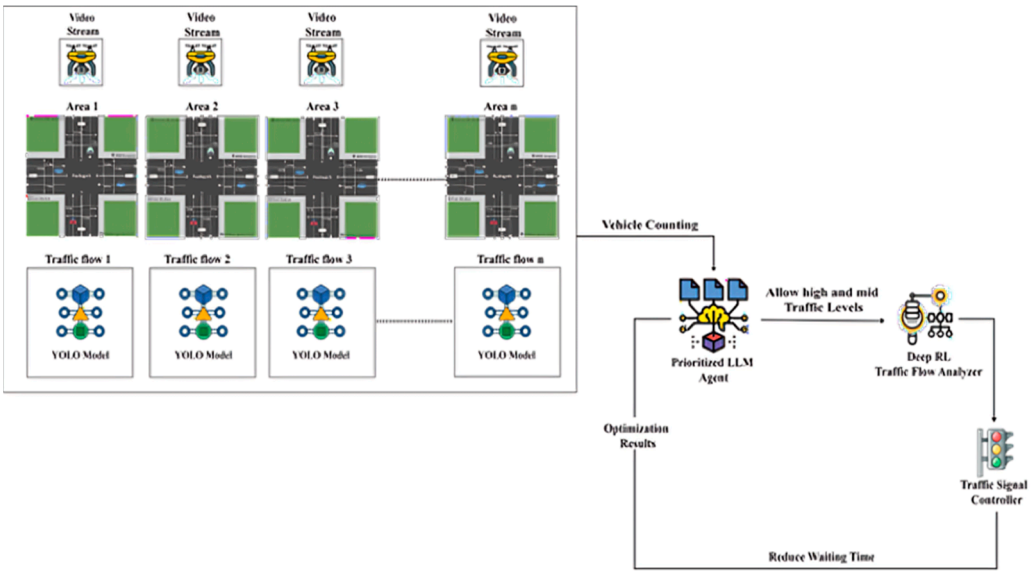


Figure 1. System architecture.

2.2. LLM Agent-Based Congestion-Aware Prioritization

Figure 2 illustrates the computation performed by the LLM agent in the proposed system for prioritizing congested areas. In this proposed system, we have used Gemini version "gemini-2.0-flash" as the LLM agent. The LLM agent receives (1) area numbers (e.g., area 1, area 2, . . ., area N), (2) lane-wise vehicle counts for each area (i.e., the number of vehicles passing through each lane), and (3) average vehicle counts per minute for each area. Then, these inputs are combined into a single prompt and sent to Gemini. Using its internal buffer management system, Gemini classifies each area into congestion levels: high, medium, or low. The areas that are classified as low are then filtered out, leaving only high- and medium-priority areas for subsequent traffic management decisions.

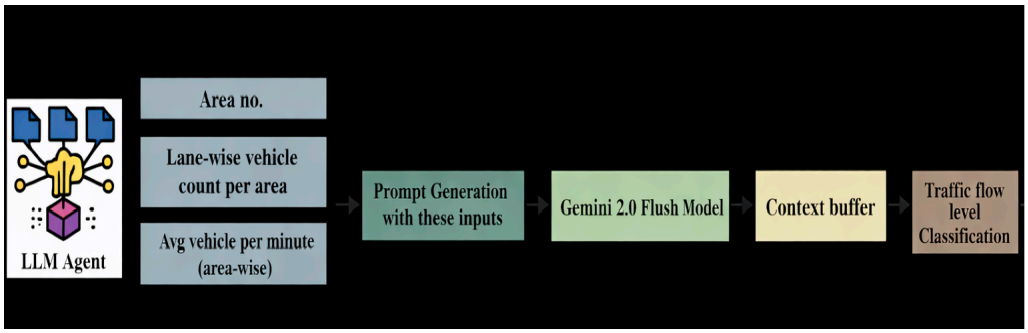


Figure 2. Proposed Prioritization LLM agent.

2.3. Proposed DRL Controller

We use DQN to optimize traffic signal control in a selected area, which is shown in Figure 3. The current state is represented by the vehicle counts of each lane at an intersection. This lane-wise traffic information helps the model understand real-time congestion levels and make decisions based on data. To capture the sequence of traffic flow, we add a Long Short-Term Memory (LSTM) layer to the DQN architecture. LSTM allows the agent to learn time-based connections, as traffic patterns at any moment are often affected by earlier conditions. Each action in the network selects one of the lanes to receive the green signal, aiming to maximize overall traffic throughput.

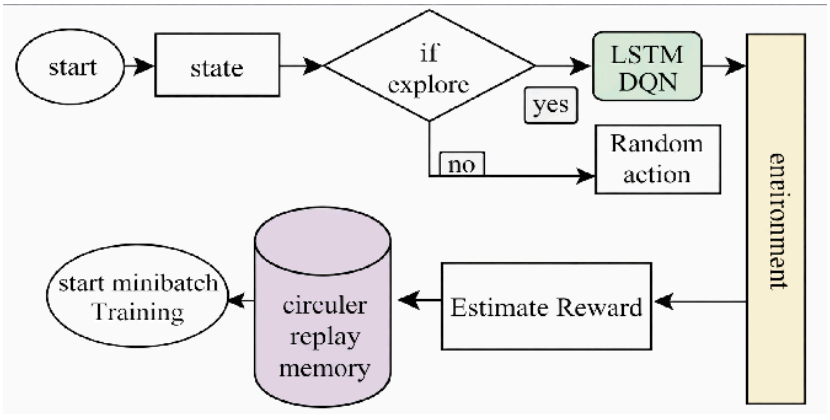


Figure 3. Proposed circular-DQN architecture.

To improve the efficiency and stability of training, we employ an experience replay memory mechanism. Rather than using a traditional deque-based buffer, the developed system uses a circular memory structure. This design allows for faster memory updates and sampling since it avoids the computational overhead of resizing dynamic memory in deque implementations. The reward mechanism supports effective decision-making. In our setup, the agent receives a reward of 1 when it successfully selects the best lane for the

green signal. This lane allows released vehicles to improve traffic flow. If the chosen lane does not reduce traffic or cause congestion, the agent receives a reward of 0. This simple reward system makes learning easier and helps the DQN agent make better lane selection choices. Figure 3 illustrates our proposed system.

3. Simulation Result

3.1. Average Waiting Time Analysis While Prioritizing Areas

Figure 4 shows a comparison of the average waiting time per area. After applying the proposed LLM agent while prioritizing areas based on traffic flow level, the average queue latency time significantly decreases for each area. Note that we use the traditional M/M/1 queue model [17] to generate the queue latency time for each area, as shown in Figure 4.

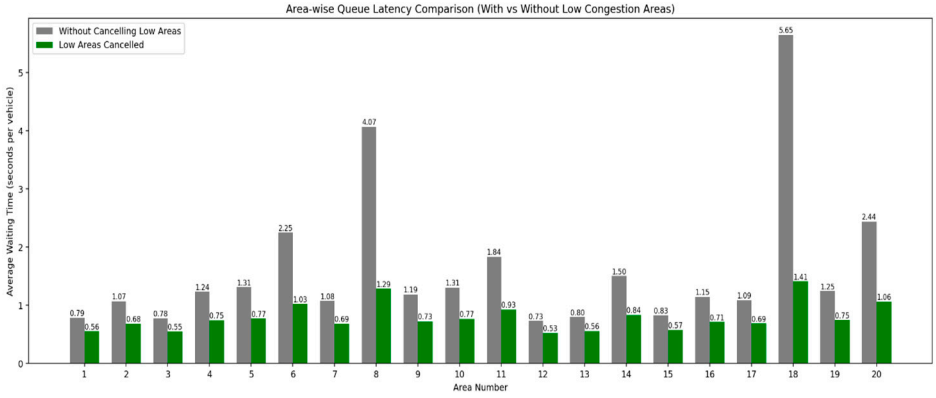


Figure 4. Average waiting time analysis while prioritizing areas.

3.2. Cumulative Rewards per Episode

As illustrated in Figure 5, the cumulative rewards initially start at a lower level due to the agent's exploratory behavior. At the initial stage, the agent performs exploratory actions to learn the traffic environment, resulting in relatively low cumulative rewards. Figure 5 further illustrates the reward convergence profile for both DRL models over 1000 episodes. The proposed circular-DQN demonstrates quick stabilization to the maximum expected cumulative reward (approximately 60) after the initial exploration phase. When comparing the proposed model with the existing Dequen-DQN approach, it is evident that the proposed model achieves higher cumulative rewards consistently throughout most of the episodes.

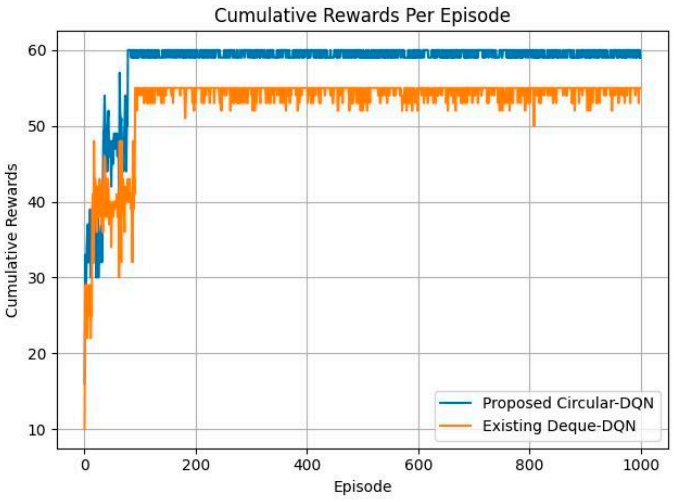


Figure 5. Cumulative rewards per episode.

3.3. Average Waiting Time Delay

The average waiting time delay ($Delay_{avg}$) is calculated using the webstar method. Figure 6 shows the cumulative distribution function (CDF) of $Delay_{avg}$ per vehicle for the developed and existing DQN. The circular-DQN significantly outperforms the existing Deque-DQN in reducing vehicle delays. The curve corresponding to the proposed method is noticeably shifted to the left, indicating that a larger proportion of vehicles experience a shorter average waiting time due to utilizing circular memory instead of deque memory.

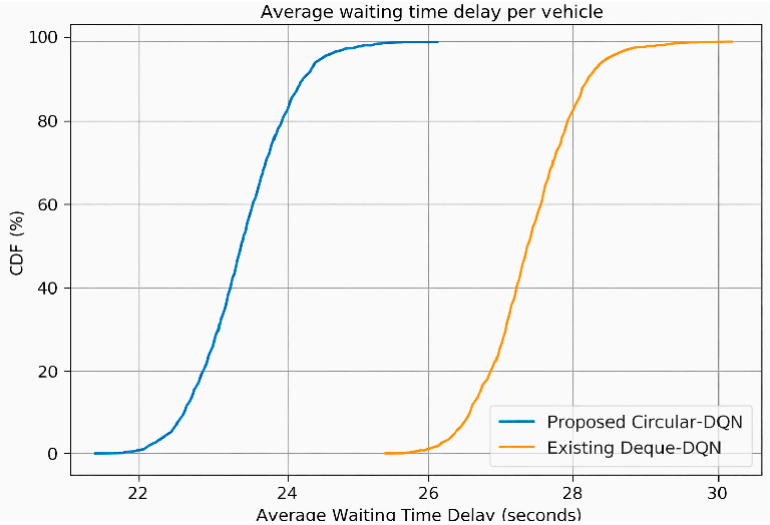


Figure 6. Average waiting time delay.

4. Conclusions

We developed a DQN-based prioritization LLM agent for better traffic signal control. We sent only high and medium traffic flows to the DRL model. We also used a YOLOv11-based drone vision system for real-time vehicle detection. This method reduces waiting times and improves signal use. Experimental results show better traffic flow and congestion management. In the future, we will focus on hardware implementation and testing in challenging urban traffic situations.

Author Contributions: Conceptualization, S.M.S. and T.A.; methodology, M.M.H.; software, M.M.H.; validation, A.S., M.A.K. and S.M.S.; formal analysis, M.M.H.; investigation, M.M.H.; resources, S.H.A.; data curation, M.M.H.; writing—original draft preparation, M.M.H.; writing—review and editing, T.A.; visualization, A.S.; supervision, T.A.; project administration, S.M.S.; funding acquisition, none. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The datasets can be found in the links below: https://github.com/mujahid11hasan/DroneDeepVision.git (accessed on 30 October 2024).

Conflicts of Interest: Author Salman Md Sultan was employed by the company Innovative Skills Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Kumar, S.; Vishal; Sharma, P.; Pal, N. Object tracking and counting in a zone using YOLOv4, DeepSORT and TensorFlow. In Proceedings of the IEEE International Conference on Artificial Intelligence and Smart Systems (ICAIS), Coimbatore, India, 25–27 March 2021; pp. 1017–1022. [CrossRef]
2. Asha, C.S.; Narasimhadhan, A.V. Vehicle counting for traffic management system using YOLO and correlation filter. In Proceedings of the IEEE International Conference Communication and Signal Processing (ICCSP), Tamilnadu, India, 3–5 April 2018; pp. 1–5.
3. Chaudhuri, A. Smart Traffic Management of Vehicles Using Faster R-CNN Based Deep Learning Method. *Sci. Rep.* **2024**, *14*, 10357. [CrossRef] [PubMed]
4. Mahmud, D.; Hajmohamed, H.; Almentheri, S.; Alqaydi, S.; Aldhaheri, L.; Khalil, R.A.; Saeed, N. Integrating LLMs with ITS: Recent advances, potentials, challenges, and future directions. *arXiv* **2025**, arXiv:2501.04437v1. [CrossRef]
5. Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. Internet of Things: A survey on enabling technologies, protocols, and applications. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2347–2376. [CrossRef]
6. Mohammed, A.K.; Anwer, R.; Hussain, F. Drone-Based Real-Time Traffic Monitoring System Using Computer Vision. *IEEE Access* **2022**, *10*, 84562–84573.
7. Lin, T.; Chen, L.; Li, Q. Lightweight Object Detection Network for UAV Traffic Monitoring. *Sensors* **2022**, *22*, 7321–7335.
8. Wang, C.; Zhang, X.; Xu, J. YOLOv11: Enhanced Lightweight Object Detection for Edge Devices. *arXiv* **2025**, arXiv:2503.01456.
9. Li, L.; Lv, Y.; Wang, F.-Y. Traffic Signal Timing via Deep Reinforcement Learning. *IEEE/CAA J. Autom. Sin.* **2016**, *3*, 247–254. [CrossRef]
10. Kővári, B.; Tamás, T.; Bécsi, T. Deep Reinforcement Learning-Based Approach for Traffic Signal Control. *Procedia Comput. Sci.* **2022**, *62*, 278–285.
11. Chu, Y.; Wang, X.; Gao, Y. Multi-Agent deep reinforcement learning for large-scale traffic signal control. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 4482–4496. [CrossRef]
12. Wei, Z.; Chen, C.; Zheng, K.; Wang, X. IntelliLight: A Reinforcement Learning Approach for Intelligent Traffic Light Control. In Proceedings of the ACM SIGKDD International Conference Knowledge Discovery & Data Mining, London, UK, 19–23 August 2018; pp. 2496–2505.
13. Zheng, Y.; Luo, J.; Gao, H.; Zhou, Y.; Li, K. Pri-DDQN: Learning Adaptive Traffic Signal Control Strategy. *Complex Intell. Syst.* **2025**, *11*, 4. [CrossRef]
14. Kwesiga, D.K.; Guin, A.; Hunter, M. Adaptive Traffic Signal Control based on Multi-Agent Reinforcement Learning: Case Study on a Simulated Real-World Corridor. *arXiv* **2025**, arXiv:2503.02189.
15. Bohra, A.R.; Selvi, T. Reinforcement Learning for Adaptive Traffic Signal Control Using Deep Q-Networks. In Proceedings of the 2025 International Conference on Sustainable Energy Technologies and Computational Intelligence (SETCOM), Gandhinagar, India, 21–23 February 2025. [CrossRef]
16. Alif, M.A.R. YOLOv11 for vehicle detection: Advancements, performance, and applications in intelligent transportation systems. *arXiv* **2024**, arXiv:2410.22898. Available online: https://arxiv.org/abs/2410.22898 (accessed on 30 October 2024). [CrossRef]
17. Talbi, D.; Boukhtouta, A.; Chraibi, M.; Tembine, H. Integrating Reinforcement Learning into M/M/1/K Retry Queuing for 6G Networks. *Preprints* **2025**. Available online: https://pmc.ncbi.nlm.nih.gov/ (accessed on 9 April 2026).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.