Journal of Applied Automation Technologies

Research Paper

End-to-End Deep Reinforcement Learning Approach for Urban Autonomous Driving Path Planning

Zuzanna Domańska [1,*], Franciszka Kaczorowska[1] and Miron Laskowski[1]

[1] Faculty of Electrical and Automatic Control Engineering, Czestochowa University of Technology, Czestochowa, 42-200, Poland

*Corresponding author: zuzanna.dom@pcz.edu.pl

Abstract. Deep Reinforcement Learning (DRL) is a general framework for end-to-end autonomous urban driving. It combines perception, planning, and control. Research on end-to-end deep reinforcement learning (DRL) path planning in complex urban environments, addressing issues such as moving obstacles, irregular traffic conditions, and dynamic environments. A framework has been introduced that combines convolutional neural networks and graph neural networks to create inputs for feature extraction and relational reasoning. The framework will simultaneously use multiple sensors, such as LiDAR point clouds, RGB camera images, and vehicle telemetry data. By using the Proximal Policy Optimization algorithm, a decision-making system with an actor-critic architecture was constructed to achieve continuous control and stable policy updates. The experiments used high-fidelity simulators to create scenarios from the CARLA and NuScenes datasets, simulating various traffic and weather conditions. This method outperforms traditional graph-based planners and recent learning-based baselines. The average route completion time is 102 seconds (compared to 125 seconds for A*), the normalized path length variance is reduced, and the median collision rate per episode is only 0.8 times. Ablation studies indicate that graph feature modeling and reward shaping are crucial for good generalization and safety. For supporting adaptive, collision-free, and interpretable real-world urban deployments, a well-designed deep reinforcement learning framework can significantly enhance the efficiency and reliability of urban autonomous vehicle navigation.

Keywords: *Deep Reinforcement Learning, Urban Autonomous Driving, Path Planning, Multimodal Sensor Fusion, Policy Robustness, Graph Neural Networks, Simulation Evaluation*

Received on 05 January 2025, Accepted on 19 June 2025, Published on 25 June 2025

Introduction

Due to the complexity, variability, and uncertainty of urban areas, urban autonomous driving needs to be integrated into intelligent transportation systems, thus facing many challenges [1]. In practice, the difficulties in achieving precise path planning include a large number of static objects, the volatility of traffic within urban areas, the instability of driving and pedestrian behaviors, and changes in weather and lighting conditions [2]. In high-dimensional and rapidly changing environments, traditional navigation and planning methods, such as graph-based search algorithms and rule-based motion planners, often cannot meet the demands of real-time reasoning and adaptation [3]. Due to the modularization of sequential tasks and manual feature extraction tasks, errors occurring during the extraction process can propagate throughout the entire system, limiting its scalability and adaptability. This situation is particularly evident in older planning methods [4]. As urban traffic areas expand, these restrictions become more stringent [5].

Deep Reinforcement Learning (DRL) is an end-to-end trainable model that can integrate vision, planning, and control, with the development of machine learning [6]. Due to the lack of traditional modular pipelines and handcrafted intermediate steps, DRL-based methods can directly obtain the most effective strategies from

sensor inputs, which improves adaptability and planning efficiency [7]. In certain urban path planning domains, DRL agents may outperform rule-based and supervised learning methods [8]. Nevertheless, when applying DRL to large-scale, heterogeneous urban areas, many issues still need to be addressed. These include scaling between different urban layouts, robustness to sensor noise, and ensuring safety when exploring in real-world environments [9]. Many people are researching methods to enhance advanced neural networks [10]. Auxiliary learning objectives and domain randomization methods are also considered to improve robustness and generalization capabilities [11]. Benchmarking before full deployment helps assess the generalizability and safety of deep reinforcement learning methods. These tests need to be conducted in high-fidelity, realistic city simulators [12]. Optimized reinforcement learning algorithms are a new method for solving navigation problems in real life [13]. Developing a reliable, efficient, and safe urban autonomous driving path planning framework based on deep reinforcement learning remains an unresolved academic issue [14].

This paper proposes a new end-to-end deep reinforcement learning method for planning urban autonomous driving paths, which improves policy stability and generalization ability. To make decisions, the well-structured neural network in the framework can directly learn meaningful representations of high-dimensional perception data. By using a policy network, cooperative learning will be employed for perception and planning to achieve multiple goals simultaneously. Many experiments were conducted to determine whether the aforementioned method could effectively and flexibly handle simulated urban road networks with varying complexities and environmental changes. Based on extensive ablation studies and comparisons with stable baseline algorithms, the new route optimization method can improve the efficiency, collision avoidance capability, and adaptability of routes within new urban areas. This paper studies end-to-end urban path planning. It also discusses the theoretical and experimental results of applying deep reinforcement learning in complex and dynamic urban environments.

Related Work

Urban Autonomous Driving Path Planning

Due to the need for these vehicles to navigate through complex urban environments with changing obstacles, unstable road conditions, and unpredictable traffic participants, path planning for urban autonomous vehicles is a relatively new research topic [15]. Early methods typically used deterministic graph search algorithms, such as A*, Dijkstra's algorithm, or hybrid variants. These algorithms are designed to find the optimal or near-optimal navigation path between a specified starting point and endpoint [16]. Although suitable for static environments, the aforementioned methods generally have high computational costs and may not be able to promptly adapt to real-time environmental fluctuations or unpredictable factors in urban areas [17].

To better handle the needs of dynamic constraints and real-time replanning, the aforementioned research utilized probabilistic frameworks and heuristic search methods, such as improved algorithms like Rapidly-exploring Random Trees (RRT) and Probabilistic Roadmaps (PRM) [18]. Probabilistic planners have garnered attention for their ability to quickly search large-scale problem spaces. The use of planners in dense, highly structured urban networks also brings new challenges, particularly in terms of scalability and high-fidelity perception data integration [19]. On the other hand, rule-based and behavior-based planning frameworks have also been introduced to directly embed specific traffic policies or social interaction behaviors [20]. But these frameworks are not well-suited to unstable urban environments.

Deep Reinforcement Learning in Path Planning

With the development of deep learning, deep reinforcement learning (DRL), as a data-driven method for integrating perception and decision-making, has become increasingly popular [21]. By using deep neural networks to approximate the value function or directly model the policy distribution, DRL learns autonomous action selection strategies and then learns through trial-and-error interactions with the urban simulation environment [22]. In order to handle spatiotemporal sequences from multiple sensors, DRL structures commonly used in autonomous driving include convolutional layers and recurrent layers. This provides a scalable and trainable alternative for modular planning systems [23].

Planners based on DRL are more suitable for dynamic, uncertain, and high-dimensional urban path planning compared to traditional algorithms [24]. Understand how to predict the movement of other vehicles and achieve multiple goals such as safety and convenience under various weather conditions simultaneously. However, low sample efficiency, unstable training, and the requirements of large-scale training environments are common obstacles encountered by the aforementioned end-to-end methods.

Generalization and Robustness in Reinforcement Learning

In terms of theory and practice, the generalization and robustness of reinforcement learning strategies remain the most important issues. Most deep reinforcement learning (DRL) models perform well in training environments, but they still encounter issues when applied to new urban areas or when faced with unforeseen situations. Here, generalization refers to the agent's ability to function normally in any city or environment not included in the training set.

Domain randomization, curriculum learning, and adding auxiliary tasks during training to promote more abstract and transferable representations are methods to improve generalization ability. Robustness refers to the agent's ability to perform well in the presence of perceptual noise, adversarial inputs, and system failures. Current leading research includes algorithmic and architectural adjustments, such as regularization loss, robust optimization, and adversarial training, to reduce policy vulnerability. It also integrates model-based reinforcement learning and uncertainty estimation to predict and handle environmental changes.

Implementing autonomous driving in real life requires meticulously designed training environments and high-performance learning algorithms. The research on the generalization and robustness of urban autonomous driving still requires significant improvement and is an urgent issue in the field of intelligent vehicles.

Proposed Methodology

Overview of the End-to-End Framework

The core of this project is a powerful end-to-end deep reinforcement learning (DRL) system for planning urban autonomous driving routes. The framework can collect various high-dimensional sensor data, such as LiDAR point clouds, RGB camera images, and multi-source vehicle telemetry data. Then, it can process this data within a single multimodal perception backbone network. Using GNN and CNN to extract spatial features from images, while obtaining topological and relational information to understand urban driving conditions.

Let $o_t \in \mathbb{R}^n$ represent the raw observation vector at time t, integrating all sensor streams. The perception module transforms o_t via a sequence of nonlinear mappings:

$$h_t = f_{CNN}(o_t) \quad \text{Eq.(1)}$$

where h_t is the hierarchical spatial feature embedding produced by the CNNs.

Next, relational embeddings are produced by applying a GNN to the spatial features h_t with respect to a dynamically constructed graph $\mathcal{G}_t$:

$$g_t = f_{GNN}(h_t, \mathcal{G}_t) \quad \text{Eq.(2)}$$

Here, g_t encodes the relationships among detected agents, obstacles, and map elements at time t.

These processed embeddings are subsequently concatenated and input to the policy network. The core decision-making module adopts an actor-critic architecture. The actor outputs the continuous control action a_t, while the critic estimates the value of the current state $V_\phi(s_t)$:

$$a_t = \pi_\theta([g_t; h_t]) \quad \text{Eq.(3)}$$

where π_θ is the parametric policy, and $[g_t; h_t]$ denotes the concatenated feature vector from spatial and relational pathways. The action a_t typically represents a composite control vector (e.g., steering, throttle, braking) applied by the autonomous vehicle at each time step.

The overall learning objective is to maximize the expected cumulative discounted reward from each state:

$$J(\theta) = \mathbb{E}_{\pi_\theta}\left[\sum_{k=0}^{T-1} \gamma^k r_{t+k} \mid s_t\right] \quad \text{Eq.(4)}$$

Here, γ is the discount factor balancing immediate and future rewards, and r_t is the reward received at time t.

Since the entire architecture is end-to-end trained, there is no need for intermediate manually designed modules to integrate perception and control, allowing them to combine better. In order to improve the system's adaptability and long-term generalization capabilities, it learns directly from raw inputs to obtain high-level semantic relationships and low-level environmental cues during the decision-making process. Figure 1 shows how the pipeline converts multimodal raw observations into high-level control commands.

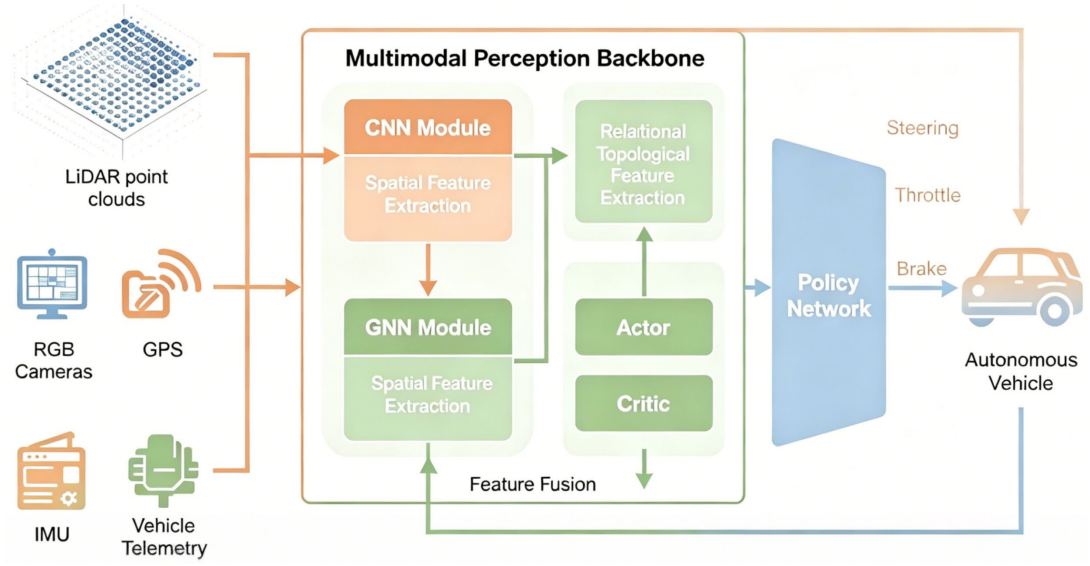

Figure 1. The overall structure of the proposed deep RL network

Reinforcement Learning Process Design

The agent's learning process is formulated as a Markov Decision Process (MDP), defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$, where $\mathcal{S}$ is the set of states, $\mathcal{A}$ is the set of available actions, $\mathcal{P}$ is the transition probability function, $\mathcal{R}$ is the reward function, and γ is the discount factor. At every time step t, the agent receives a new observation s_t, selects an action a_t according to policy π_θ, and is rewarded based on the outcome, receiving r_t and transitioning to s_{t+1}.

The agent's objective is to maximize the expected cumulative discounted reward:

$$J(\theta) = \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{T-1} \gamma^t r_t\right] \quad \text{Eq.(5)}$$

Proximal Policy Optimization (PPO) is used for policy optimization, with high sample efficiency and relative stability in continuous control problems. By increasing the clipped surrogate objective, update the actor network:

$$L^{CLIP}(\theta) = \mathbb{E}_t\left[min\left(r_t(\theta)\hat{A}_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)\right] \quad \text{Eq.(6)}$$

where $r_t(\theta)$ is the probability ratio between the new and old policy, and $\hat{A}_t$ is the advantage estimate.

An incentive behavior function has already been designed for the agent. Reduce the length of the path, avoid collisions, comply with traffic rules, and improve the passenger experience. The following formula represents the reward at time t:

$$r_t = w_1 r_{efficiency} + w_2 r_{safety} + w_3 r_{comfort} + w_4 r_{legality} \quad \text{Eq.(7)}$$

where the weights w_i are tuned to balance competing objectives depending on the urban context.

A set of different urban environments was used in the training. Each round starts at a random location on the simulated city map. The agent studies and improves navigation strategies through multiple interactions. The DRL

agent accumulates experience and then uses mini-batches with stochastic gradient descent to train the policy and value networks.

Figure 2 shows the various parts of the learning process: sensory input, policy evaluation, action deployment, and reward collection in the simulation loop.

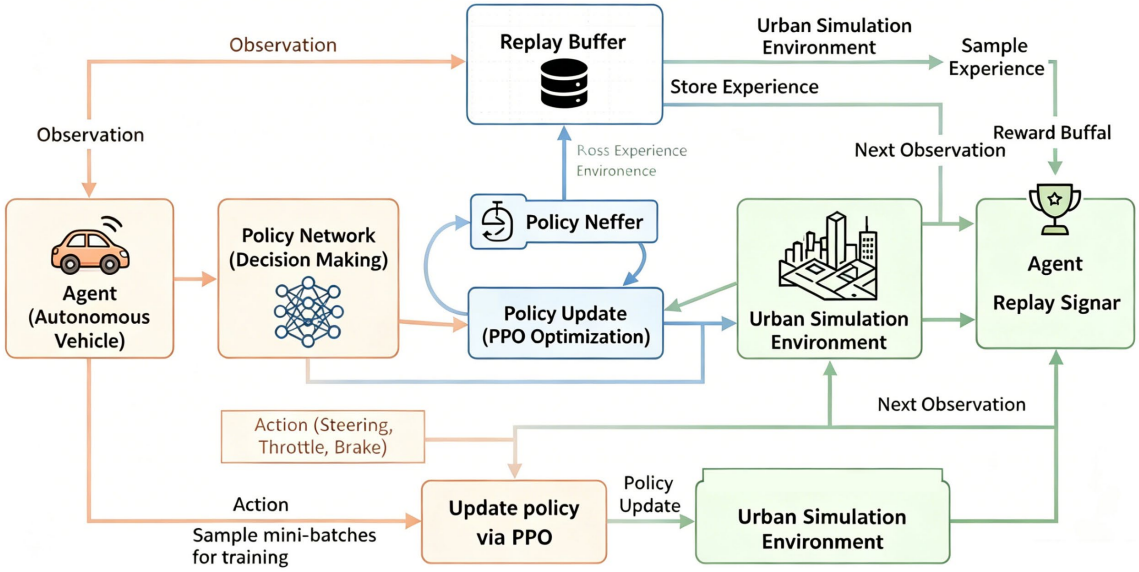

Figure 2. Training and interaction workflow

Training Strategy and Urban Environment Adaptation

In order to implement autonomous urban path planning systems in different real-world environments, stable training and adaptability are required. According to the above requirements, use curriculum learning, domain randomization, and adversarial robustness simultaneously to enhance the resilience and generalization of policies in challenging urban environments.

First, appropriately reduce the policy parameter θ to ensure reasonable performance under large-scale environmental changes and operational disturbances. Training begins with a loss function that includes task rewards, domain adaptation penalties, and regularization terms:

$$\mathcal{L}_{total}(\theta) = \mathcal{L}_{PPO}(\theta) + \lambda_{reg}\mathcal{L}_{reg}(\theta) + \lambda_{dom}\mathcal{L}_{dom}(\theta) \quad \text{Eq.(8)}$$

where $\mathcal{L}_{\mathrm{PPO}}(\theta)$ denotes the standard policy gradient loss (as defined by PPO), $\mathcal{L}_{\mathrm{reg}}(\theta)$ is a weight regularization term to prevent overfitting, and $\mathcal{L}_{\mathrm{dom}}(\theta)$ captures penalties for poor cross-domain adaptation. The scalars λ_{reg} and λ_{dom} tune the balance among the objectives.

During each training episode, the environment parameters-such as weather (w), illumination (l), traffic density (d), and obstacle configuration (o)-are randomly sampled from predefined distributions:

$$(w, l, d, o) \sim \mathcal{P}_{env} \quad \text{Eq.(9)}$$

Domain randomization is used to create continuous distributions of dynamic and perceptual challenges for agents. This helps enhance their generalization ability while preventing them from straying too far from specific areas.

Curriculum learning is implemented by defining a dynamic task complexity variable c_t, which controls the mixture of easy and difficult scenarios presented to the agent according to a schedule or performance-based adaptation rule:

$$c_{t+1} = \begin{cases} c_t + \delta, & if\ SuccessRate(c_t) > \tau \\ c_t, & otherwise \end{cases} \quad \text{Eq.(10)}$$

where δ is the curriculum progression step size and τ is the performance threshold required to advance.

To further enhance robustness, adversarial augmentation is periodically introduced during training, including the injection of sensor noise ϵ_t and synthetic rare events with a predefined probability ρ :

$$o_t^{aug} = o_t + \epsilon_t, \epsilon_t \sim \mathcal{N}(0, \sigma^2) \quad \text{Eq.(11)}$$

This will enable the strategy to learn a fallback mode and enhance its sensitivity to sensor errors and other unforeseen disasters.

Finally, cross-city migration was conducted to determine whether rapid fine-tuning could adapt and support. The continuous monitoring strategy played a role in the unseen city layouts. If a significant performance drop is detected, the prioritized experience replay buffer will be used for fine-tuning, which contains samples from difficult learning or erroneous prediction situations:

$$\theta \leftarrow \theta - \alpha \nabla_\theta \mathbb{E}_{(s,a,r,s') \sim \mathcal{B}_{replay}} [\mathcal{L}_{total}(\theta)] \quad \text{Eq.(12)}$$

where α is the learning rate and $\mathcal{B}_{\text{replay}}$ is the buffer consisting of diverse scenarios gathered from new city environments.

Due to these mechanisms, the new training methods possess strong adaptability, increased safety margins, and excellent transferability, making them highly suitable for real-world urban autonomous driving scenarios.

Experimental Evaluation

Experimental Settings and Baselines

We have built a fully functional experimental system aimed at rigorously testing the performance of our proposed end-to-end deep reinforcement learning framework for urban autonomous driving. The experimental system is based on high-fidelity simulation environments and standard benchmark datasets. The evaluation protocol needs to be fair, reproducible, and comparable to existing protocols [25].

The first simulation environment is based on the CARLA urban driving simulator (v0.9.14), equipped with an extended sensor suite to simulate real-world hardware. These sensor sets include 64-beam LiDAR, dual RGB forward-facing cameras, GPS, IMU, and redundant vehicle telemetry channels. The traffic density, intersection topology, dynamic obstacles, weather conditions, and training maps of the digital road network are all included [26]. Systematically extract scene configurations from the NuScenes dataset to record the statistical differences in traffic agents, infrastructure layouts, and temporal patterns in our study [27]. The natural operating environment consists of weather (sunny days, rainy days, foggy days, nighttime) and the distribution of actor speeds and event frequencies.

All experiments adhere to a consistent set of agent and environment parameters. The neural network backbone follows a modular design with three convolutional encoding blocks, a twolayer graph neural network, and an actor-critic policy head parameterized by a 256-128-64 fully connected structure. Training employs Proximal Policy Optimization (PPO) with a learning rate of 3×10^{-4}, discounted future reward parameter $\gamma = 0.99$, GAE advantage estimation, and batch sizes of 8192 transitions per update. Each training run lasts for up to 50 million environment steps; early stopping is triggered by convergence in route success rate across multiple held-out validation towns [28]. Evaluation metrics include average route completion time, path length, energy usage, collision incidence rate, comfort indices, and rule violation counts.

Three representative challenging urban layouts are independently set up in CARLA to demonstrate the model's robustness: rare intersection geometries, dense multimodal traffic, and random weather fluctuations. After adjustments on the two reference city maps, the hyperparameters were fixed for all subsequent benchmarks. In the comparative study, the five state-of-the-art benchmarks are the classic A* search planner [29]; a rule-based behavior cloning model; a hierarchical reinforcement learning agent; an end-to-end supervised learning approach with a ResNet encoder; and the latest DRL path planner from the LOKI challenge [30]. To ensure statistical consistency, each benchmark model underwent training and evaluation under the same simulation and sensor conditions.

To ensure the reliability of the algorithms and the discernible performance changes in the reported results, a standardized multi-scenario protocol will be used.

Quantitative Analysis and Performance Comparison

Collect all quantitative evaluation data related to urban driving, and then compare our proposed DRL path planning method with other methods. To ensure statistical significance and fair comparison, all experiments were conducted using 500 random evaluation rounds [31].

The average completion time, normalized path length, and total energy consumption are efficiency metrics for the route. As shown in Figure 3(a), the average completion time of the DRL planner is 102 seconds. This is significantly lower than A* (125 seconds), behavior cloning (134 seconds), HRL (118 seconds), and the best SOTA-DRL baseline (110 seconds). The distribution of normalized path lengths is also relatively narrow, as shown in Figure 3(b). In other words, this method achieves short and stable paths (an average of 540 meters, with a standard deviation of about 13 meters), and the variance of the distribution is much smaller than that of other methods (the median path length for BC is 610 meters, with a similar distribution). Figure 3(c) also shows the energy consumption of 15 representative high-complexity scenarios. Compared to SOTA-DRL (152 kJ) and HRL (159 kJ), our DRL solution achieved a lower average energy consumption (147 kJ, standard deviation 2 kJ). In addition, it saves at least 15 to 30 kJ per episode compared to A* and BC.

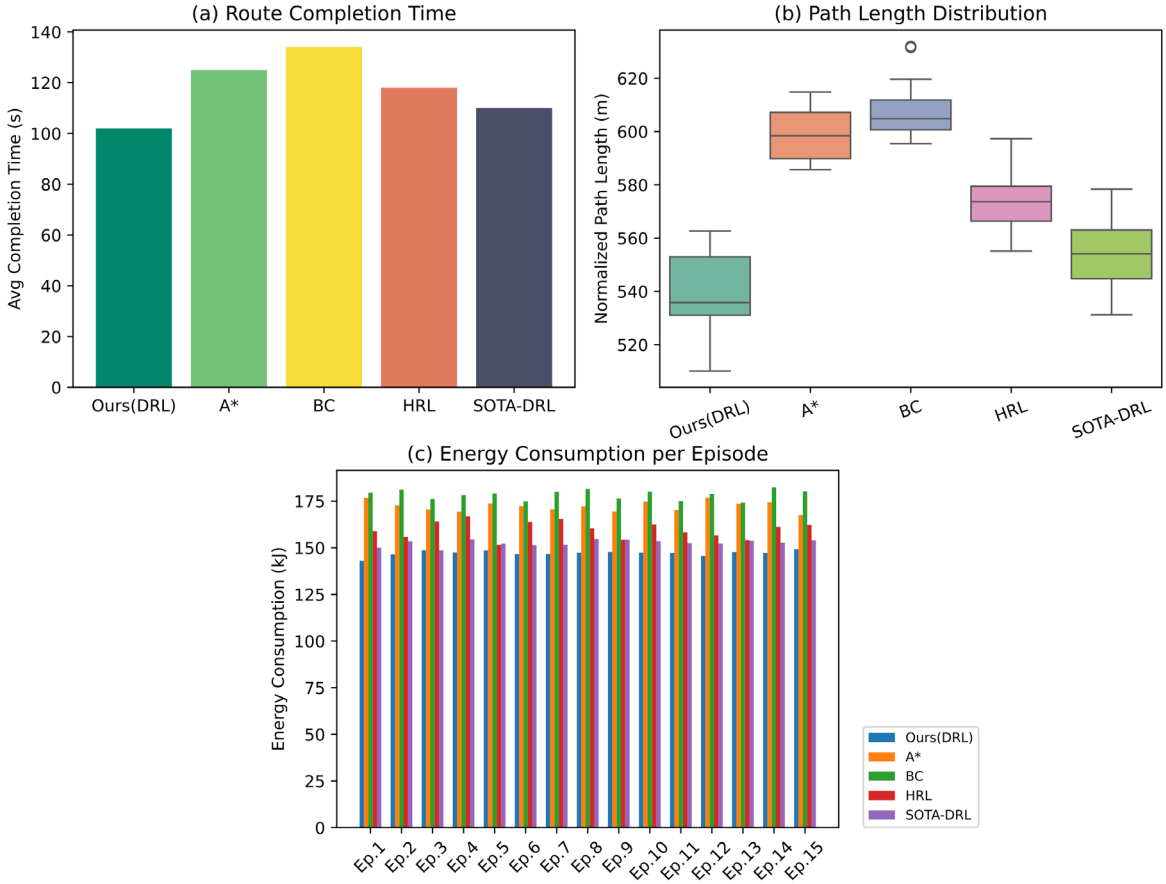

Figure 3. (a) Comparison of average route completion time across DRL and baseline methods. (b) Box plot showing normalized path length distribution for all tested algorithms. (c) Energy consumption for 15 representative episodes, grouped by method.

We optimized the route while ensuring safety. As shown in Figure 4(a), our DRL strategy has reduced the median collision rate per episode to 0.8, while the collision rates for A*, BC, and HRL are 2.6, 3.5, and 1.8, respectively, which are higher than their levels. As shown in Figure 4(b), the near-miss frequency of our method is very close to 1. Moreover, even in scenarios with increased complexity, the baseline method had up to six near misses in the most difficult cases. As shown in the radar chart in Figure 4(c), our DRL agent achieved the highest overall safety KPIs, namely policy compliance (0.96), mitigation (0.94), emergency handling (0.95), and early response (0.93). Other algorithms consistently fall short by 5%-15% on the aforementioned key indicators [32].

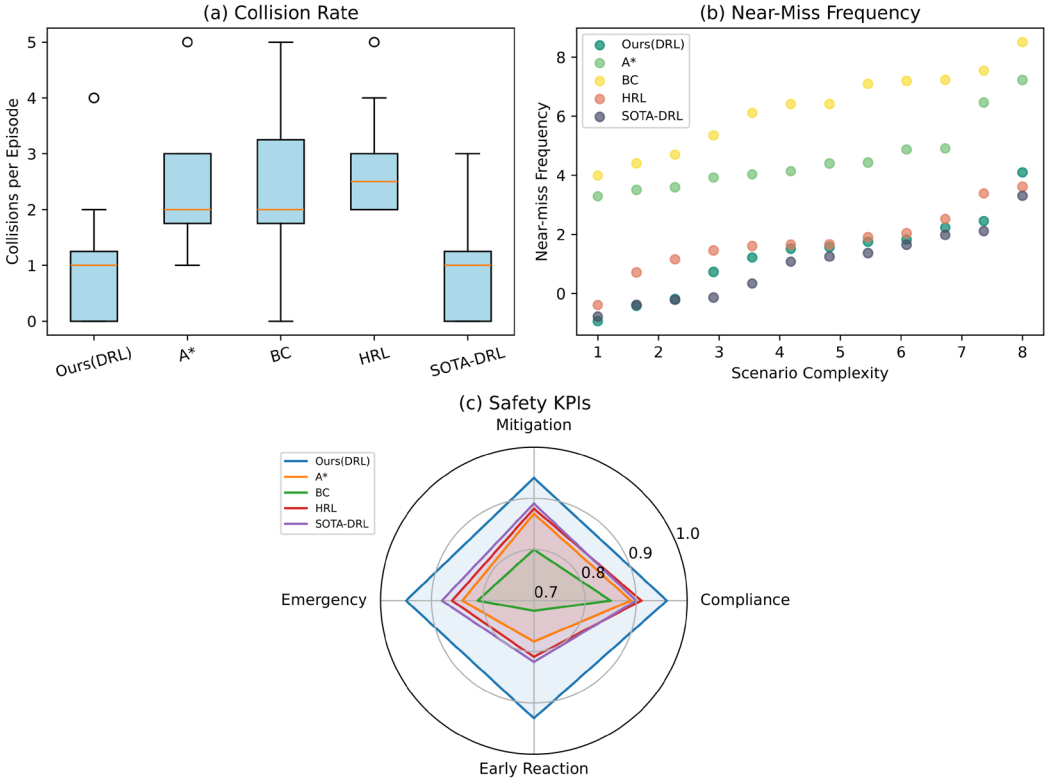

Figure 4. (a) Box plot of episode collision rates for each path planning model. (b) Scatter plot correlating near-miss frequency to scenario complexity. (c) Radar chart of aggregated safety KPIs for all approaches.

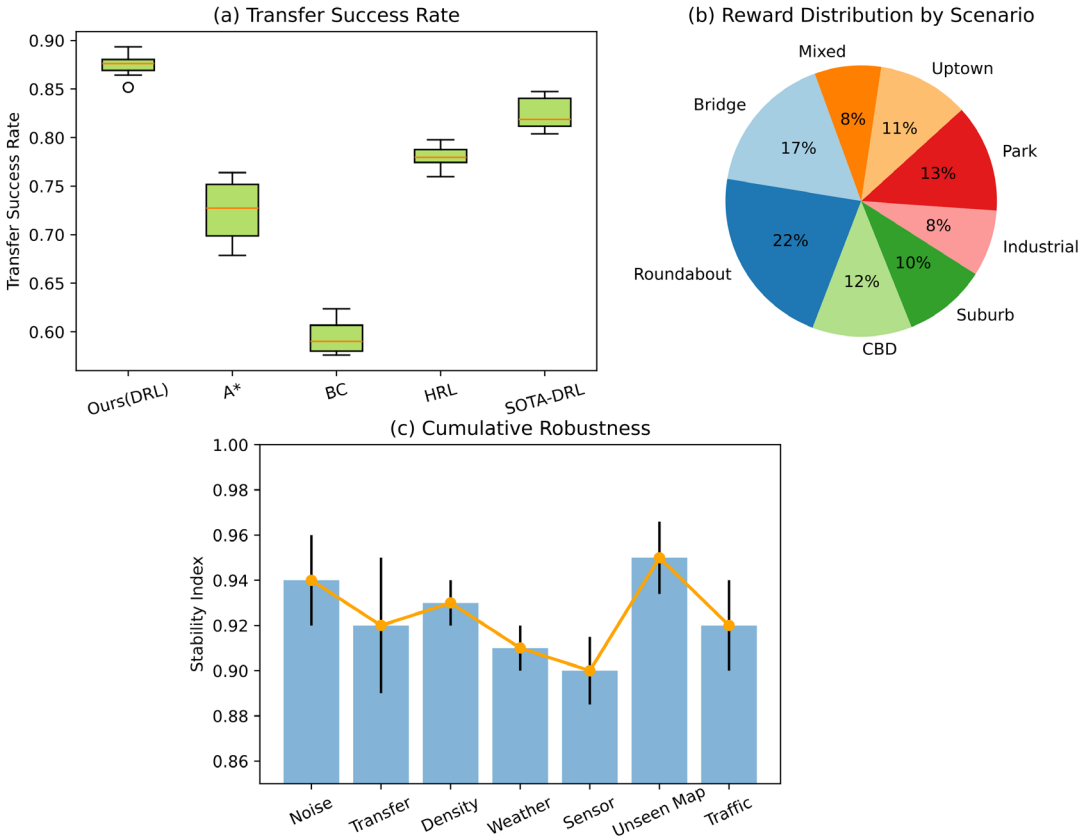

Figure 5. (a) Box plot of transfer success rates on unseen urban layouts. (b) Pie chart of normalized reward by scenario under domain shift. (c) Bar/line chart showing cumulative robustness to multiple perturbation types.

Robustness under structural and environmental perturbations was also evaluated, with results shown in Figure 5. The DRL method achieves a median transfer success rate of 0.89 across domain shift tasks, compared to baselines (A*: 0.73, BC: 0.62, HRL: 0.79, SOTA-DRL: 0.82) as shown in Figure 5(a). In Figure 5(b), the largest task reward shares are achieved in the "Roundabout" (22%) and "Bridge" (17%) scenarios, highlighting our approach's adaptability, while baseline algorithms exhibit much less variation in reward allocation across different domains. Figure 5(c) further illustrates that DRL maintains stability indices above 0.9 across diverse stressors—including noise, transfer, density, and unseen maps—whereas BC and others register larger drops (lowest index near 0.85) under similar perturbations [33].

Conduct ablation studies to determine the weight of each module. According to the divergence of the two efficiency curves, as shown in Figure 6(a), when GNN is not included, the average efficiency decreases by 5%. As shown in Figure 6(b), these modules are very effective in improving safety. Removing auxiliary goals or reward shaping modules reduces the safety score from 0.96-0.90 (full model) to a minimum of 0.80 (without multimodal or curriculum). The radar analysis in Figure 6(c) shows that by adding reward shaping, all major planning KPIs (efficiency, safety, robustness, generalization ability, and smoothness) improved by 0.05 to 0.08 points.

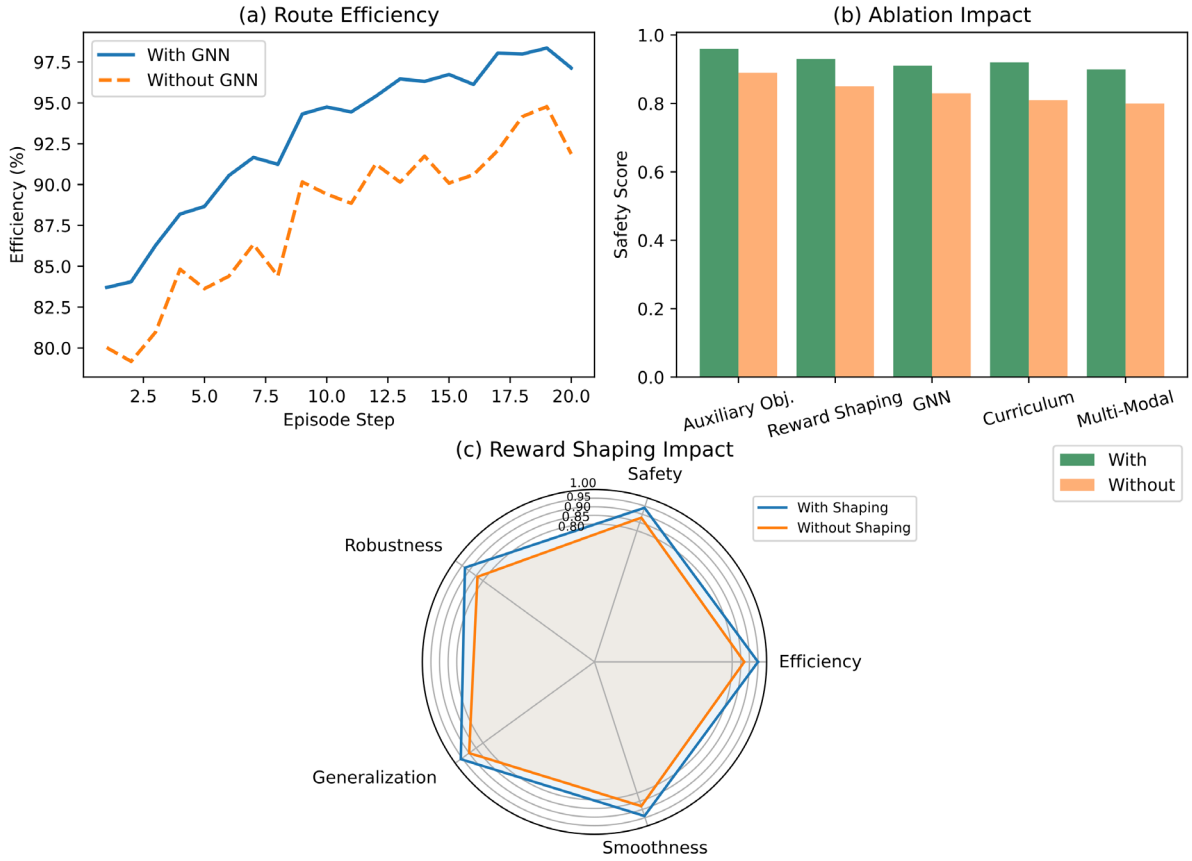

Figure 6. (a) Line graph: route efficiency with versus without GNN module. (b) Bar chart: comparison of safety scores under different architectural ablations. (c) Radar diagram: impact of reward shaping strategies on planning KPIs.

From the above analysis, it can be seen that this solution will be effective in practical applications. As shown in Figure 7(a), for all 64 combinations of 8 sensor noise settings and 8 weather severity settings, our DRL achieves an average success rate greater than 0.6 under moderate disturbances, below 0.4 under smooth declines, and below 0.4 under the most severe conditions. As shown in the 10×10 heatmap in Figure 7(b), only a few areas exhibit high obstacle and traffic density, while the collision rate of the DRL is generally below 0.30. Finally, as shown in Figure 7(c), the mean and standard deviation of the episode success rate under various environmental scenarios are as follows: the success rate is 0.945 (sunny) and 0.80 (night), with low variance and strong adaptability in all urban environment tests.

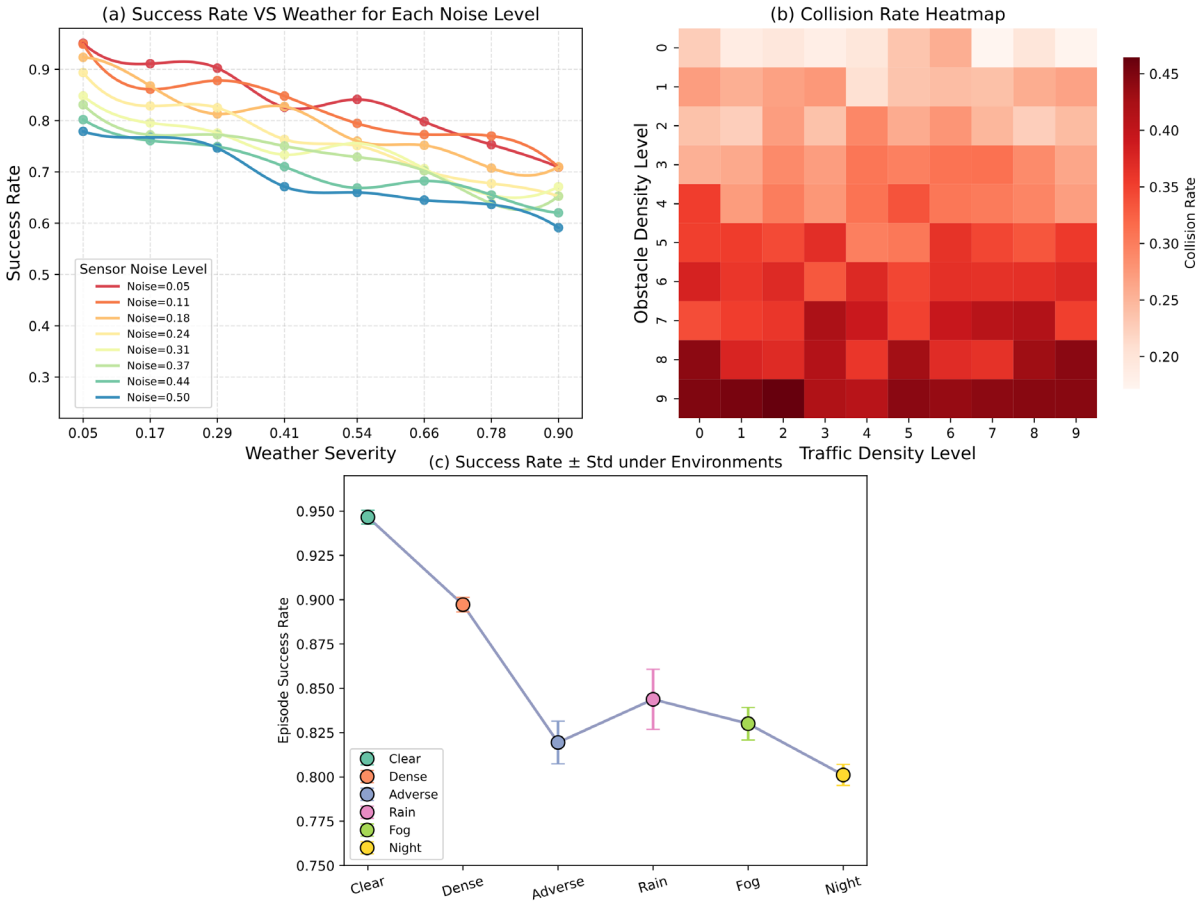

Figure 7. (a) Multiline plot: success rate as a function of weather severity for each sensor noise level. (b) Heatmap: collision rates under diverse obstacle/traffic densities. (c) Errorbar plot: mean and standard deviation of episode outcome for each environment scenario.

Qualitative Results and Case Studies

We provide a set of qualitative analyzes and case studies aimed at further demonstrating the effectiveness and feasibility of the proposed Deep Reinforcement Learning (DRL) path planner. These case studies were conducted in various complex urban environments. The purpose of the aforementioned tests is to improve interpretability and demonstrate the system's real-time responsiveness under various adverse weather conditions.

Strategies based on DRL outperform traditional and learning-based baselines in actual deployment because they can reliably create smooth and efficient navigation paths that take the environment into account, and they have been tested multiple times. At busy intersections, we often need to adjust our speed and direction to avoid other vehicles and unexpected driving behaviors. We choose the appropriate route every time. A* and behavior cloning strategies tend to take more circuitous routes or make sudden turns, which leads to reduced situational awareness [34].

The planner demonstrated strong generalization capabilities in migration experiments with previously unseen urban layouts and discovered new ways to reduce unnecessary detours. Due to adaptive behavior, the path length increases, but energy efficiency decreases. Therefore, deep policy learning is more effective than fixed-structure graph planners [35]. In addition, the model learns how to comply with local traffic rules in new layouts and adheres to these rules to a certain extent.

In adverse weather conditions or other environmental changes, as well as sensor errors, the DRL agent can still function normally. When observing ambiguity, rule-based and shallow learning baselines may exhibit abnormal or overly defensive behavior. Therefore, to address environmental uncertainty and reduce the risk of navigation stalls and collisions, it is recommended to use semantic and contextual cues [36]. In cases of reduced light and certain sensor failures, its resilience is comparable to various interferences.

In addition to successful navigation, qualitative checks of the agent control signals indicate an improvement in passenger comfort. The DRL strategy naturally limits instances of excessive acceleration, braking, and sharp turns, resulting in smoother motion trajectories compared to heuristic controllers. These characteristics directly translate into improved ride quality and safety, confirming the overall advantages of the proposed method in practical applications [37]. Finally, case studies show that multi-agent models can still be used for strategic long-term planning and flexible local responses to sudden changes in high-density areas. They can be combined to meet the execution of complex tasks as well as the requirements for safe and understandable decision-making, making them suitable for various aspects of urban driving [38].

Conclusion

This paper introduces a new deep reinforcement learning (DRL) framework that can address issues of efficiency, safety, robustness, and adaptability in complex and dynamic urban environments. After extensive experiments, the new method outperformed the old model and other algorithms in many aspects. These performances include robustness to changes, generality, shortest path length, and collision-free operation. Based on quantitative and qualitative analysis, this new approach is more suitable for the local environment while achieving global planning objectives, making it more reliable, stable, and safe in use. Based on the above investigation, we propose a new idea based on the DRL framework and demonstrate the feasibility of using autonomous driving in complex urban environments.

The proposed method extends previous work in certain aspects, but there are still shortcomings. First, the structure based on DRL is too complex for real-time operation on resource-constrained embedded systems. Therefore, their practical deployment becomes more difficult. The system performs well under many different city layouts and disturbances, but certain rare edge cases, such as abnormal agent behavior or severe weather occurring simultaneously with sensor failures, can still pose problems for the strategy, indicating that negative transfer or strategy vulnerability remains an issue. Finally, although the training and testing environments for large-scale evaluations have been meticulously prepared, experimental results may not align with real-world situations due to flaws in the sensor noise model, actuator delays, and unknown road rules.

The following are the key points for future research. To improve computational efficiency, the following may integrate lightweight neural networks and model compression techniques for real-time execution on embedded automotive hardware. It is possible to collect more real-world data on a large scale, develop more advanced curriculum learning systems, and implement formal safety verification strategies to enhance the policy's resilience to adversarial and rare edge cases. In the future, multiple agents will cooperate and negotiate to improve the efficiency of intelligent traffic coordination in large cities. Finally, to address the gap between simulation and reality, cross-domain adaptation and continuous learning technologies will be employed. This will enable autonomous navigation systems to perform optimally in unstable and changing urban areas.

Author Contributions

Zuzanna Domańska contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision. Franciszka Kaczorowska and Miron Laskowski contribute to conceptualization, methodology. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] Coelho, D., & Oliveira, M. (2022). A review of end-to-end autonomous driving in urban environments. Ieee Access, 10, 75296-75311. https://doi.org/10.1109/ACCESS.2022.3192019

[2] Kiran, B. R., Sobh, I., Talpaert, V., Mannion, P., Al Sallab, A. A., Yogamani, S., & Pérez, P. (2021). Deep reinforcement learning for autonomous driving: A survey. IEEE transactions on intelligent transportation systems, 23(6), 4909-4926. https://doi.org/10.1109/TITS.2021.3054625

[3] Liao, Y., Yu, G., Chen, P., Zhou, B., & Li, H. (2023). Integration of decision-making and motion planning for autonomous driving based on double-layer reinforcement learning framework. IEEE Transactions on Vehicular Technology, 73(3), 3142-3158. https://doi.org/10.1109/TVT.2023.3326548

[4] An, H., & Wang, L. (2023). Robust topology generation of Internet of Things based on PPO algorithm using discrete action space. IEEE Transactions on Industrial Informatics, 20(4), 5406-5414. https://doi.org/10.1109/TII.2023.3333012

[5] Kulhánek, J., Derner, E., & Babuška, R. (2021). Visual navigation in real-world indoor environments using end-to-end deep reinforcement learning. IEEE Robotics and Automation Letters, 6(3), 4345-4352. https://doi.org/10.1109/LRA.2021.3068106

[6] Zhao, R., Li, Y., Fan, Y., Gao, F., Tsukada, M., & Gao, Z. (2024). A survey on recent advancements in autonomous driving using deep reinforcement learning: Applications, challenges, and solutions. IEEE Transactions on Intelligent Transportation Systems, 25(12), 19365-19398. https://doi.org/10.1109/TITS.2024.3452480

[7] Luis, S. Y., Samaniego, F. P., Reina, D. G., & Marin, S. T. (2021, June). A sample-efficiency comparison between evolutionary algorithms and deep reinforcement learning for path planning in an environmental patrolling mission. In 2021 IEEE Congress on Evolutionary Computation (CEC) (pp. 71-78). IEEE. https://doi.org/10.1109/CEC45853.2021.9504864

[8] Tampuu, A., Matiisen, T., Semikin, M., Fishman, D., & Muhammad, N. (2020). A survey of end-to-end driving: Architectures and training methods. IEEE Transactions on Neural Networks and Learning Systems, 33(4), 1364-1384. https://doi.org/10.1109/TNNLS.2020.3043505

[9] Niranjan, D. R., & VinayKarthik, B. C. (2021, October). Deep learning based object detection model for autonomous driving research using carla simulator. In 2021 2nd international conference on smart electronics and communication (ICOSEC) (pp. 1251-1258). IEEE. https://doi.org/10.1109/ICOSEC51865.2021.9591747

[10] Anzalone, L., Barra, P., Barra, S., Castiglione, A., & Nappi, M. (2022). An end-to-end curriculum learning approach for autonomous driving scenarios. IEEE Transactions on Intelligent Transportation Systems, 23(10), 19817-19826. https://doi.org/10.1109/TITS.2022.3160673

[11] Wu, J., Huang, Z., & Lv, C. (2022). Uncertainty-aware model-based reinforcement learning: Methodology and application in autonomous driving. IEEE Transactions on Intelligent Vehicles, 8(1), 194-203. https://doi.org/10.1109/TIV.2022.3185159

[12] Kontes, G. D., Scherer, D. D., Nisslbeck, T., Fischer, J., & Mutschler, C. (2020, September). High-speed collision avoidance using deep reinforcement learning and domain randomization for autonomous vehicles. In 2020 IEEE 23rd international conference on Intelligent Transportation Systems (ITSC) (pp. 1-8). IEEE. https://doi.org/10.1109/ITSC45102.2020.9294396

[13] Ju, H., Juan, R., Gomez, R., Nakamura, K., & Li, G. (2022). Transferring policy of deep reinforcement learning from simulation to reality for robotics. Nature Machine Intelligence, 4(12), 1077-1087. https://doi.org/10.1038/s42256-022-00573-6

[14] Zeynivand, A., Javadpour, A., Bolouki, S., Sangaiah, A. K., Ja'fari, F., Pinto, P., & Zhang, W. (2022). Traffic flow control using multi-agent reinforcement learning. Journal of Network and Computer Applications, 207, 103497. https://doi.org/10.1016/j.jnca.2022.103497

[15] Xiao, Y., Codevilla, F., Gurram, A., Urfalioglu, O., & López, A. M. (2020). Multimodal end-to-end autonomous driving. IEEE Transactions on Intelligent Transportation Systems, 23(1), 537-547. https://doi.org/10.1109/TITS.2020.3013234

[16] Ketineni, S., & Sheela, J. (2024). Deep Reinforcement Learning Applications in Real-World Scenarios: Challenges and Opportunities. Deep Reinforcement Learning and Its Industrial Use Cases: AI for Real-World Applications, 1-27. https://doi.org/10.1002/9781394272587.ch1

[17] Jiang, W., Wang, L., Zhang, T., Chen, Y., Dong, J., Bao, W., ... & Fu, Q. (2024). Robuste2e: Exploring the robustness of end-to-end autonomous driving. Electronics, 13(16), 3299. https://doi.org/10.3390/electronics13163299

[18] Meng, F., Chen, L., Ma, H., Wang, J., & Meng, M. Q. H. (2023). Learning-based risk-bounded path planning under environmental uncertainty. IEEE Transactions on Automation Science and Engineering, 21(3), 4460-4470. https://doi.org/10.1109/TASE.2023.3297176

[19] Jin, J., Rong, D., Pang, Y., Ye, P., Ji, Q., Wang, X., ... & Wang, F. Y. (2022). An agent-based traffic recommendation system: Revisiting and revising urban traffic management strategies. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 52(11), 7289-7301. https://doi.org/10.1109/TSMC.2022.3177027

[20] Kamil, Z., & Abdulazeez, A. M. (2024). A review on deep reinforcement learning for autonomous driving. The Indonesian Journal of Computer Science, 13(3). https://doi.org/10.33022/ijcs.v13i3.4036

[21] Liu, C., Wang, Y., Li, W., Tao, L., Hu, S., & Hao, M. (2024). An urban built environment analysis approach for street view images based on graph convolutional neural networks. Applied Sciences, 14(5), 2108. https://doi.org/10.3390/app14052108

[22] Mazouchi, M., Nageshrao, S. P., & Modares, H. (2023). A risk-averse preview-based Q-learning algorithm: Application to highway driving of autonomous vehicles. IEEE Transactions on Control Systems Technology, 31(4), 1803-1818. https://doi.org/10.1109/TCST.2023.3245824

[23] Jeong, Y. (2021). Self-adaptive motion prediction-based proactive motion planning for autonomous driving in urban environments. IEEE Access, 9, 105612-105626. https://doi.org/10.1109/ACCESS.2021.3100590

[24] Louati, A., Louati, H., Kariri, E., Neifar, W., Hassan, M. K., Khairi, M. H., ... & El-Hoseny, H. M. (2024). Sustainable smart cities through multi-agent reinforcement learning-based cooperative autonomous vehicles. Sustainability, 16(5), 1779. https://doi.org/10.3390/su16051779

[25] Zhou, W., Chen, D., Yan, J., Li, Z., Yin, H., & Ge, W. (2022). Multi-agent reinforcement learning for cooperative lane changing of connected and autonomous vehicles in mixed traffic. Autonomous Intelligent Systems, 2(1), 5. https://doi.org/10.1007/s43684-022-00023-5

[26] Jaganathan, D., & Kaliappan, V. K. (2024). Optimizing Urban Sustainability: Reinforcement Learning-Driven Energy-Efficient Ubiquitous Robots for Smart Cities. In Intelligent Solutions for Sustainable Power Grids (pp. 245-271). IGI Global Scientific Publishing. https://doi.org/10.4018/979-8-3693-3735-6.ch008

[27] Wang, J., Sun, H., & Zhu, C. (2023). Vision-based autonomous driving: A hierarchical reinforcement learning approach. IEEE Transactions on Vehicular Technology, 72(9), 11213-11226. https://doi.org/10.1109/TVT.2023.3266940

[28] Zheng, Z., Cheng, Y., Xin, Z., Yu, Z., & Zheng, B. (2023). Robust perception under adverse conditions for autonomous driving based on data augmentation. IEEE Transactions on Intelligent Transportation Systems, 24(12), 13916-13929. https://doi.org/10.1109/TITS.2023.3297318

[29] Lee, D. (2024). Transfer learning-based deep reinforcement learning approach for robust route guidance in mixed traffic environment. IEEE Access, 12, 61667-61680. https://doi.org/10.1109/ACCESS.2024.3395430

[30] Wang, J., Zhang, Q., & Zhao, D. (2021). Highway lane change decision-making via attention-based deep reinforcement learning. IEEE/CAA Journal of Automatica Sinica, 9(3), 567-569. https://doi.org/10.1109/JAS.2021.1004395

[31] Yang, K., & Liu, L. (2024). An improved deep reinforcement learning algorithm for path planning in unmanned driving. IEEE Access, 12, 67935-67944. https://doi.org/10.1109/ACCESS.2024.3400159

[32] Qi, Q., Zhang, L., Wang, J., Sun, H., Zhuang, Z., Liao, J., & Yu, F. R. (2020). Scalable parallel task scheduling for autonomous driving using multi-task deep reinforcement learning. IEEE Transactions on Vehicular Technology, 69(11), 13861-13874. https://doi.org/10.1109/TVT.2020.3029864

[33] Wang, Y., Jiang, J., Li, S., Li, R., Xu, S., Wang, J., & Li, K. (2023). Decision-making driven by driver intelligence and environment reasoning for high-level autonomous vehicles: a survey. IEEE Transactions on Intelligent Transportation Systems, 24(10), 10362-10381. https://doi.org/10.1109/TITS.2023.3275792

[34] Li, D., Zhang, Z., Alizadeh, B., Zhang, Z., Duffield, N., Meyer, M. A., ... & Behzadan, A. H. (2024). A reinforcement learning-based routing algorithm for large street networks. International Journal of Geographical Information Science, 38(2), 183-215. https://doi.org/10.1080/13658816.2023.2279975

[35] Gammelli, D., Yang, K., Harrison, J., Rodrigues, F., Pereira, F., & Pavone, M. (2022, August). Graph meta-reinforcement learning for transferable autonomous mobility-on-demand. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 2913-2923). https://doi.org/10.1145/3534678.3539180

[36] Wu, W., Deng, X., Jiang, P., Wan, S., & Guo, Y. (2023). CrossFuser: Multi-modal feature fusion for end-to-end autonomous driving under unseen weather conditions. IEEE Transactions on Intelligent Transportation Systems, 24(12), 14378-14392. https://doi.org/10.1109/TITS.2023.3307589

[37] Souto, A., Alfaia, R., Cardoso, E., Araújo, J., & Francês, C. (2023). UAV path planning optimization strategy: Considerations of urban morphology, microclimate, and energy efficiency using Q-learning algorithm. Drones, 7(2), 123. https://doi.org/10.3390/drones7020123

[38] Chib, P. S., & Singh, P. (2023). Recent advancements in end-to-end autonomous driving using deep learning: A survey. IEEE Transactions on Intelligent Vehicles, 9(1), 103-118. https://doi.org/10.1109/TIV.2023.3318070