DEEP REINFORCEMENT LEARNING-BASED OBSTACLE AVOIDANCE FOR UAV IN AUTONOMOUS PATROL SYSTEMS

Long-Nhat Dang[1], Gia-Bao Le-Tran[1], Duc-Quan Tran[1], Manh-Cuong Nguyen[1,*]

DOI: https://doi.org/10.57001/huih5804.2026.077

ABSTRACT

In recent years, research on models and methods for UAVs navigation in complex environments has attracted significant attention from the research community. In particular, within autonomous UAV-based patrol systems, ensuring flight safety in obstacle-rich environments remains a considerable challenge. This paper presents a UAVs navigation solution for obstacle avoidance based on deep reinforcement learning models in autonomous patrol systems. The proposed solution is implemented within an integrated software, along with simulation tools that enable the evaluation of UAV obstacle avoidance capabilities in both dynamic and static scenarios during patrol missions. Experimental results across environments of varying complexity demonstrate that the UAVs can effectively avoid obstacles while maintaining a stable flight trajectory.

Keywords: *Obstacle Avoidance, Navigation, Reinforcement Learning, Autonomy, Deep Learning.*

[1]School of Information and Communications Technology, Hanoi University of Industry, Vietnam

[*]Email: manhcuong.nguyen@haui.edu.vn

Received: 18/01/2026

Revised: 20/3/2026

Accepted: 30/3/2026

1. INTRODUCTION

Autonomous patrol systems using unmanned aerial vehicles (UAVs) are becoming an important area of research and application. The goal is to build a system capable of comprising multiple UAVs to patrol a large area with complex terrain while minimizing human intervention. They are commonly used for various purposes such as security surveillance, search and rescue, infrastructure inspection, environmental monitoring, and resource protection. Thanks to their high maneuverability, rapid deployment, and ability to operate in challenging terrain conditions, UAVs significantly expand the range and effectiveness of patrol and surveillance missions compared to traditional methods.

To develop such a system effectively, we need to address not only the simple flight control problem, but also combine the solutions of many closely related sub-problems. First, UAVs need to perform optimal Coverage Path Planning (CPP) to patrol the area in order to ensure full observation of the area of interest [1, 14]. During task execution, the system must continuously be aware of the environment and address the problem of obstacle avoidance (dynamic and static obstacles), object detection and classification, object tracking and target searching [8]. To improve operational efficiency, especially in large-scale scenarios, the multi-UAV coordination problem [4] is also posed in order to divide tasks, optimize resources and coordinate the handling of events.

In this series of problems, obstacle avoidance plays a crucial role in ensuring the safety and continuity of the navigation process. UAVs need to be able to detect and react promptly to static and dynamic obstacles, while maintaining their original flight path to serve their target and mission. This is a difficult problem due to the uncertainty of the environment, especially in environments with noise and incomplete observations.

Traditional methods often rely on manually designed geometric models, environmental maps, and control laws [12]. They are considered to work effectively in simple environments. However, with environments becoming more complex or difficult to model, more flexible solutions are required. Deep Reinforcement Learning (DRL)-based methods for UAVs navigation with obstacle avoidance are an effective alternative research and experimental direction [13]. Complex network structures such as Dueling

Double Deep Q-Learning combined with computer vision have been proposed to enhance autonomous navigation in confined spaces [11]. Recently, research has focused on improving these models to create more efficient variants. Advanced variants such as D3QN with prioritization mechanism [7], ARDDQN network combined with attention mechanism [6], and double-map method [10] have been applied to simultaneously solve the problem of obstacle avoidance and connection maintenance. Improvements on Double DQN have also yielded many results in learning complex navigation strategies [2, 3, 5, 9]. Although current research has made significant progress in terms of models and methods, further testing in different environments is still required, especially in practical implementation.

This paper proposes a navigation and obstacle avoidance solution for UAVs in autonomous patrol systems based on DDQN model. The research's contribution is evident in two points: Firstly, the authors propose a general procedure with flexibly operated modes to combine obstacle avoidance, trajectory tracking, and flight safety. Secondly, the study designed and refined key components of the DDQN, such as the reward function, step mechanism, state space, and learning strategy, to improve the system's efficiency and stability under real-world conditions.

The remainder of the paper is organized as follows: Section 2 presents the obstacle avoidance problem in UAV-based autonomous patrol systems. Section 3 describes the architecture of the solution as well as the navigation and obstacle avoidance mechanism. Section 4 presents simulation settings and some experimental results. Finally, the conclusion and future development directions are presented.

2. OBSTACLE AVOIDANCE IN AUTONOMOUS PATROL SYSTEMS

Consider the problem of coverage path planning for UAVs in an autonomous patrol system. Given a Region Of Interest (ROI) denoted by Ω which is discretized into a grid of $m \times n$ equally spaced cells:

$$C = \{C_{ij} \mid i = 1, \dots, m, j = 1, \dots, n\}.$$

The centers of the cells are called waypoints. The coverage path planning problem aims to determine a trajectory P that is a finite sequence of waypoints: $P = \{p_0, p_1, p_2, \dots, p_N\}$ where p_0 is the starting point such that every cell in C is visited at least once, while optimizing a cost function $J(P)$. Let the obstacle set at time *t* be defined as

$$\mathcal{O}_t = \mathcal{O}_s \cup \mathcal{O}_d(t)$$

where $\mathcal{O}_s \subset \Omega$ denotes the set of static obstacles and $\mathcal{O}_d(t) \subset \Omega$ represents the set of dynamic obstacles at time *t*. Accordingly, the safe movement space is defined as:

$$\Omega_{free}(t) = \Omega \setminus \mathcal{O}_t$$

Due to the time-varying nature of $\mathcal{O}_d(t)$, a waypoint $p \in P$ may become infeasible at excution time *t* if

$$p \notin \Omega_{free}(t).$$

Let $\mathcal{S}$ and $\mathcal{A}$ be the state space and action space, respectively. The UAV dynamics are modeled as a discrete-time system:

$$s_{k+1} = f(s_k, a_k)$$

where $s_k \in \mathcal{S}$ is the system state at step k, $a_k \in \mathcal{A}$ is the control signal, and $f(.)$ is the state transition function. Let x_k denote the UAV position extracted from the state s_k. In the presence of obstacles, the navigation process must satisfy the following safety constraint:

$$dist(x_k, O_k) \geq d_{min}, \forall k$$

where $d_{min} > 0$ is the minimum safe distance for collision avoidance. The navigation process can be divided into the following phases:

– Mission execution: The UAV moves along reference points in the initial target trajectory to complete the mission of covering the patrol area.

– Collision avoidance: When the system detects a risk of collision with an obstacle, the obstacle avoidance mechanism is activated. At this situation, the UAV can temporarily suspend its original trajectory and focus on both avoiding the obstacle and checking safety conditions before re-establishing the initial route.

– Path rejoining: When safety conditions for re-establishing the initial path are ensured, the UAV performs the re-establishment action and continues its journey.

However, in real-world situations, the process of obstacle avoidance and path re-establishment can repeat infinitely, or repeat so many times that it affects the energy level of the UAV.

The above problem presents several challenges in implementation. Because the set $\mathcal{O}_d(t)$ can change rapidly over time and does not follow a defined rule, it leads to high uncertainty. Furthermore, the state s_k is often only observed indirectly through sensors with noise and limited range. This leads to partial observability in the

problem formulation. Typically, the control set $\mathcal{A}$ is limited by the kinematic constraints of the UAV. Therefore, not every obstacle avoidance action is feasible.

3. OBSTACLE AVOIDANCE SOLUTION

In this section, we introduce a solution to ensure that UAVs can avoid obstacles in a dynamic environment while ensuring maximum ROI coverage for patrol missions. The main idea is to separate the phases and design a flexible phase transition mechanism, where the UAV temporarily deviates from its initial path when encountering an obstacle and returns when safe conditions are ensured. Consider the initial flight path defined by a set of waypoints:

$$P = \{p_0, p_1, \ldots, p_N\}, p_0 \equiv p_N \equiv p_{home}$$

At each time t, the UAV moves towards the target waypoint p_i. In the obstacle-free conditions, the UAV follows the route P. This operating mode is called "Normal" and is set by default at the time of departure. When obstacles appear in the path, the UAV does not continue following P but switches to “Avoidance” mode. The goal at this stage is to ensure flight safety while maintaining the direction toward waypoint p_i. At this situation, the UAV's trajectory may deviate significantly from the original path. Once in "Avoidance" mode and the original path re-establishment is safe, the UAV switches to "Recovery" mode to move to p_i and continue its patrol.

However, practical deployment may encounter many situations where the environment exhibits complex spatial distributions and dynamic variations. In such cases, avoiding obstacles and returning to continue the journey can become a loop, leading to insufficient energy to complete the original mission. To handle this situation, an emergency control mechanism is proposed with the highest priority. Two different levels of emergency are used. At level 1, when the remaining energy level drops to just enough for landing, an emergency landing command is activated. At level 2, when the remaining energy level is only enough to return to the starting point, the "Go home" command is activated. This emergency mechanism prioritizes the safety of the device and accepts mission abort without abandoning any points along the patrol path.

Specifically, the obstacle avoidance mechanism is implemented according to the following process:

Phase 1: Obstacle detection and redirection. During movement, the UAV continuously receives sensor data. When an obstacle is detected within the danger zone of the current direction of movement, the UAV activates the avoidance mechanism and may temporarily deviate from the original path.

Phase 2: Obstacle avoidance. In this phase, the UAV activates and uses an AI-powered control model (possibly deep reinforcement learning or its variants) to avoid collisions with obstacles while maintaining the direction toward the target waypoint. The avoidance and target tracking process is repeated continuously step by step, based on updated environmental conditions.

Phase 3: Re-establishing the original path. When there are no more obstacles in the direction toward the target waypoint, the UAV returns to its original patrol route and head towards the target point p_i. If obstacles reappear during the re-establishment process, the UAV reverts to the obstacle avoidance phase. Conversely, once the UAV reaches a safe trajectory and the original path, it resumes its patrol mission.

In addition to the obstacle avoidance mechanism, the system also integrates safety conditions related to energy. These conditions are continuously monitored and have higher priority than the navigation mechanism. The proposed mechanism ensures a balance between mission integrity and equipment safety. The solution focuses not only on finding a way to avoid obstacles in real time but also on handling cases where the UAV falls into areas with complex obstacle structures. By establishing a highest-priority emergency control layer, this solution ensures that the UAV can operate safely. Furthermore, in specialized missions such as infrastructure inspection or environmental monitoring, missing any waypoint would lead to a data chain disruption. Therefore, this operating mechanism will help ensure data continuity. Separating the "Avoidance" mode allows the system to flexibly integrate advanced machine learning models without affecting the other two modes. Thus, the solution can be seen as a behavioral management strategy that addresses navigation risks while adapting flexibly to the uncertainty of the real-world environment.

4. SIMULATION SETUP AND EXPERIMENTAL RESULTS

The system is implemented using Visual Studio to provide an integrated interface that allows for basic control commands and monitoring of UAV behavior. The simulation environment utilizes various tools such as AirSim, PX4-Autopilot, and QGroundControl. AirSim provides the environmental and sensor models. PX4-Autopilot handles flight dynamics and control simulation. QGroundControl software is used to track the flight path.

Scenarios are constructed with the presence of static and dynamic obstacles to evaluate the UAV's navigation capabilities under near-realistic conditions.

The UAV uses a sensing mechanism based on sensing rays distributed around it over a 360° field of view, discretized into 36 uniformly spaced directions. The maximum signal measurement distance of sensor d_{max} is set to 35 meters. The avoidance mechanism is activated to ensure a minimum safe distance from the UAV to obstacles $d_{min} = 6$ meters. Regarding the UAV's speed, the considered speed is set to 7 meters/second. The optimal patrol path with waypoints is obtained from solving the CPP optimization problem corresponding to the patrol area using the CPLEX solver and saved as a .json file.

The obstacle avoidance model used in "Avoidance" mode is DDQN [15]. The state space consists of 36-dimensional vectors representing obstacle distances measured by 36 LiDAR sensing beams distributed around the UAV on the flight plane. The action space includes discrete steering commands used to adjust the UAV flight direction during navigation. The DDQN is trained using an ε-greedy exploration strategy and experience replay mechanism. The main training hyperparameters are summarized in Table 1.

Table 1. Main hyperparameters used for DDQN training

Parameter	Value
Learning rate	0.0001
Discount factor (γ)	0.99
Batch size	128
Replay buffer size	200000

The reward function is designed to ensure components safety, trajectory tracking, and smooth control. The specific setup of this function is presented in Table 2.

Table 2. Reward mechanism: v_x be the forward velocity of the UAV

Component	Description	Reward value
Collision	Collision penalty	-100
Distance maintenance	Distance-based safety penalty	$-50 * e^{-0.3*d_{min}}$
Path length	Path-length reward	$2.0 * v_x$
Steering	Oscillatory motion penalty	-2
Steering	Steering magnitude penalty	$-2 * (e^{0.006*\lvert yaw\_rate\rvert}) - 1$
Survival	Survival reward	1

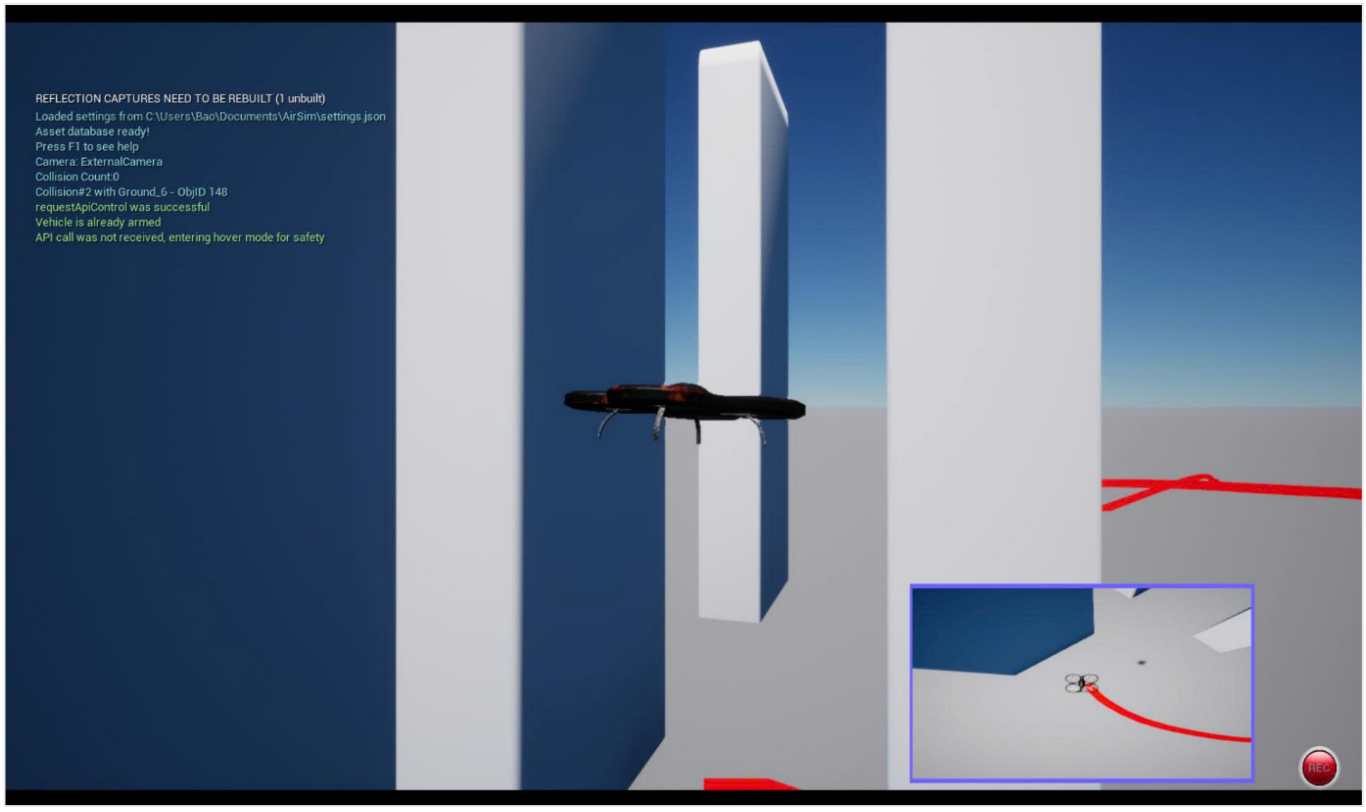

Figure 1. Blocks simulation environment with obstacles

Figure 1 presents the Blocks simulation environment in AIRSIM. The environment is designed with various obstacles distributed differently, creating complex navigation scenarios. In this environment, the UAV performs the mission of moving along a predetermined path while continuously identifying and avoiding static, box-shaped obstacles. The secondary window (bottom right corner) provides a top-down view, supporting visual observation of the UAV's trajectory.

This environment is embedded in an integrated software system that we developed ourselves, allowing for monitoring the UAV's movement and obstacle avoidance. Figure 2 shows the main interface of this software system. The main window shows the UAV's trajectory in the obstacle-filled environment. The secondary window on the right shows the entire UAV trajectory (blue line) and the obstacles (red circles). A more complex environment with lower-contrast tree obstacles was also included in the experiment, as shown in Figure 3.

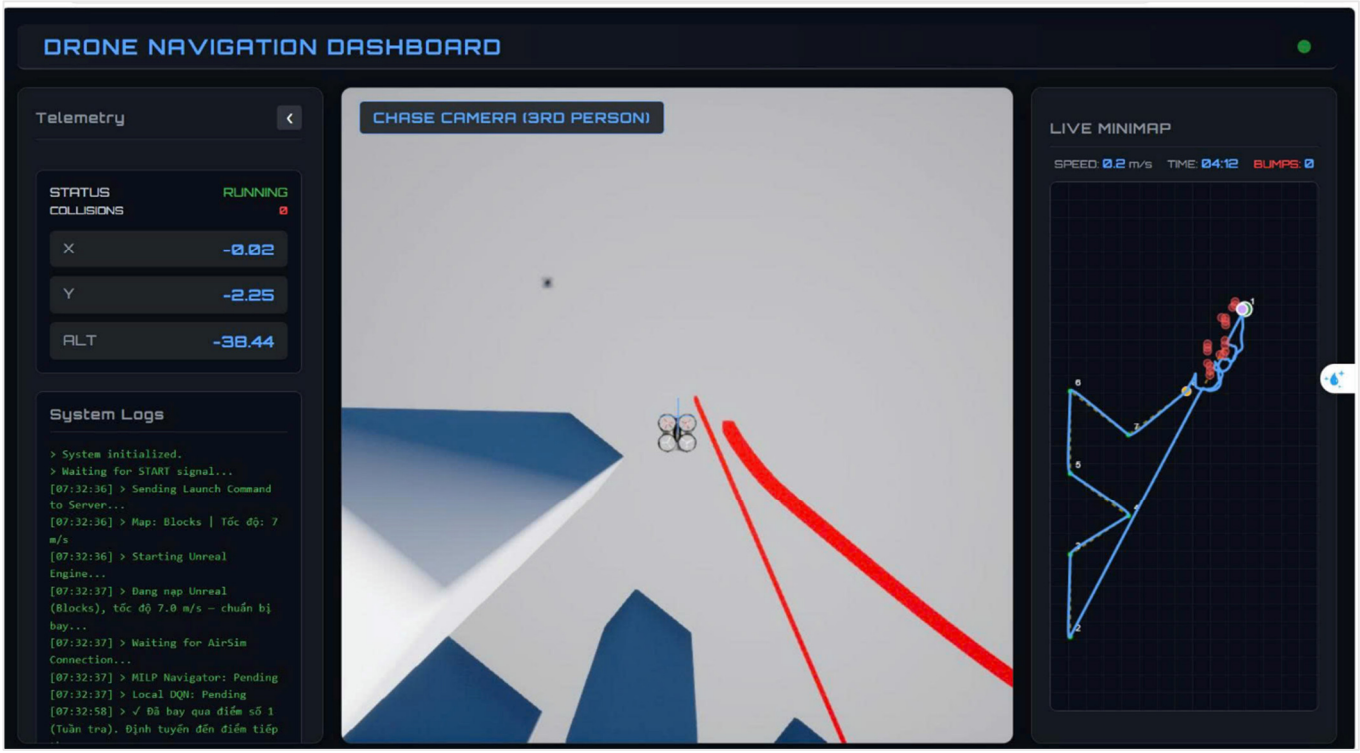

Figure 2. UAV trajectory during obstacle avoidance

Initial simulation results show that the UAV can handle obstacle scenarios quite effectively. When approaching complex geometric structures, the obstacle avoidance mechanism made flexible and appropriate navigation decisions to perform both "Avoidance" and "Recovery" modes. The UAV completed its travel path and overcame obstacles effectively. Furthermore, the resulting trajectory demonstrates a smooth movement strategy (see Figure 2). In particular, the UAV avoids sudden or sharp turns that could lead to instability or unnecessary energy consumption. The fact that the UAV maintained a safe distance despite the dense arrangement of obstacles indicates that the reward function was designed appropriately. Overall, the results demonstrate that the proposed approach is capable of achieving stable and reliable navigation in complex environments.

To further evaluate the effectiveness of the proposed approach, several quantitative performance metrics were measured, including collision avoidance success rate, trajectory length, mission completion time, and estimated energy consumption. Experiments were conducted in two different simulated environments, namely Blocks and CityPark, with different obstacle structures and sensor conditions. The Blocks environment contained structured geometric obstacles to train basic DDQN and evaluate rewards, while the CityPark environment included natural obstacles such as trees and vegetation to assess the robustness of the proposed method under more realistic conditions. Each scenario was evaluated over 10 independent runs under identical initial conditions to verify the consistency and stability of the proposed method. The results are summarized in Table 3.

Table 3. Quantitative navigation performance in different environments

Test	Environment	Waypoints	Number of obstacles	Travel Time (s)	Travel Distance (m)	Collisions
1	CityPark	10	6	132	783	0
2	CityPark	27	5	331	1742	0
3	Blocks	10	6	306	2004	0
4	Blocks	7	5	205	1302	0

The quantitative results show that the proposed method successfully completed all navigation scenarios without collisions in both Blocks and CityPark environments. The UAV maintained stable navigation behavior across different waypoint configurations while achieving effective obstacle avoidance in environments with varying complexity levels.

In an environment with low-contrast obstacles (see Figure 3), the UAV's trajectory shows that the system maintained stable obstacle detection and flexible avoidance reflexes. Although the ability to visually distinguish between obstacles and the surrounding environment is limited, the cognitive mechanism continues to provide enough information to make decisions.

Figure 3. Navigation in an environment with low-contrast obstacles

5. CONCLUSION

This paper proposes an integrated solution for UAV obstacle avoidance navigation within an autonomous patrol system. The obstacle avoidance mechanism is separated into a distinct phase, allowing for the selection of different deep learning models and improvements within these models without affecting other phases of the patrol process. The data model utilizes LiDAR with a 360-degree scanning angle to ensure maximum UAV safety. Double Deep Q-Network model is used to make decisions on obstacle avoidance. Simulation results in AirSim, PX4-Autopilot, and integrated flight control systems demonstrate the UAV's ability to effectively detect and avoid obstacles in various environments while maintaining the continuity of the patrol mission. Furthermore, the system remains stable even in environments with noise and low reflectivity. This provides a solid foundation for future real-world UAV experiments.

ACKNOWLEDGMENTS

The authors express their sincere gratitude to the Artificial Intelligence Research and Applications Center (AIC), School of Information and Communications Technology, Hanoi University of Industry, for their support and valuable resources provided during this research.

REFERENCES

[1]. Ni J., Ge Y., Zhao Y., Gu Y., "An Improved Multi-UAV Area Coverage Path Planning Approach Based on Deep Q-Networks," *Appl. Sci.*, 15, 11211, 2025.

[2]. Liu J., Luo W., Zhang G., Li R., "Unmanned Aerial Vehicle Path Planning in Complex Dynamic Environments Based on Deep Reinforcement Learning," *Machines*, 13, 162, 2025.

[3]. Yu R., Li Q., Ji J., et al., "Improved double DQN with deep reinforcement learning for UAV indoor autonomous obstacle avoidance," *Sci Rep,* 15, 28133, 2025.

[4]. Alejandro Puente-Castro, et al., "Q-Learning based system for Path Planning with Unmanned Aerial Vehicles swarms in obstacle environments," *Expert Systems with Applications,* 235, 121240, 2024.

[5]. He Yong, Hou Ticheng, Wang Mingran., "A new method for unmanned aerial vehicle path planning in complex environments," *Scientific Reports*, 14, 2024. Doi: 10.1038/s41598-024-60051-4.

[6]. Kumar Praveen, Priyadarshni, Misra Rajiv, "ARDDQN: Attention Recurrent Double Deep Q-Network for UAV Coverage Path Planning and Data Harvesting," *arXiv:2405.11013*, 2024. Doi: 10.48550/arXiv.2405.11013.

[7]. Mehmet Gök, "Dynamic path planning via Dueling Double Deep Q-Network (D3QN) with prioritized experience replay," *Applied Soft Computing*, 158, 111503, 2024.

[8]. Mehrez Boulares, Afef Fehri, Mohamed Jemni, "UAV path planning algorithm based on Deep Q-Learning to search for a floating lost target in the ocean," *Robotics and Autonomous Systems*, 179, 104730, 2024.

[9]. Tang J., Liang Y., Li K., "Dynamic Scene Path Planning of UAVs Based on Deep Reinforcement Learning," *Drones*, 8, 60, 2024.

[10]. Zhong W., Wang X., Liu X., et al., "Joint optimization of UAV communication connectivity and obstacle avoidance in urban environments using a double-map approach," *EURASIP J. Adv. Signal Process*, 35, 2024.

[11]. Jiajun Ou, Xiao Guo, Ming Zhu, Wenjie Lou, "Autonomous quadrotor obstacle avoidance based on dueling double deep recurrent Q-learning with monocular vision," *Neurocomputing,* 441, 300-310, 2021.

[12]. Zhu Kai, Zhang Tao, "Deep Reinforcement Learning Based Mobile Robot Navigation: A Review," *Tsinghua Science and Technology*, 26, 674-691, 2021. Doi: 10.26599/TST.2021.9010012.

[13]. H. Bayerlein, M. Theile, M. Caccamo, D. Gesbert, "UAV Path Planning for Wireless Data Harvesting: A Deep Reinforcement Learning Approach," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, Taipei, Taiwan, 1-6, 2020.

[14]. Theile Mirco, Bayerlein Harald, Nai Richard, Gesbert David, Caccamo Marco, "UAV Coverage Path Planning under Varying Power Constraints using Deep Reinforcement Learning," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Las Vegas, NV, USA, 2020-2021. Doi: 10.1109/IROS45743.2020.9340934.

[15]. Hado van Hasselt, Arthur Guez, David Silver, "Deep Reinforcement Learning with Double Q-learning," *arXiv:1509.06461*, 2015. https://doi.org/10.48550/arXiv.1509.06461.