Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A Path Planning Method for Fire Detection in Educational Experimental Zones Based on an Improved Deep Q-learning Algorithm

Lingqing Feng
Shanxi Agricultural University
Yujing Liu
fenglq@sxau.edu.cn

Shanxi Agricultural University
LULU ZHENG
Shanxi Agricultural University

Article

Keywords:

Posted Date: September 2nd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9862398/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

A Path Planning Method for Fire Detection in Educational Experimental Zones Based on an Improved Deep Q-learning Algorithm

Lingqing Feng[1], **Yujing Liu**[1,*], **and Lulu Zheng**[1]

[1]College of Information Science and Engineering,Shanxi Agricultural University,Jinzhong,030801,China
[*]corresponding.author fenglq@sxau.edu.cn

ABSTRACT

To address the complexity and real-time requirements of UAV path planning for fire detection in educational experimental zones, this paper proposes an enhanced Deep Q-Learning (DQN)-based method that overcomes traditional DQN's limitations of local optima entrapment and poor path efficiency in complex environments. The improvements include:(1) a multi-objective optimization mechanism through reward function redesign, integrating dynamic risk weighting and path length constraints; (2) diagonal motion incentives (providing a +15 reward bonus and 0.8 Q-value weighting) to reduce path tortuosity, and (3) A*-inspired heuristic constraints. Training stability is enhanced by using a low learning rate(0.0003) and an exponentially decayed ε-greedy strategy($\varepsilon_min = 0.1$). Simulation tests on a 20×20 grid demonstrate that the proposed method achieves an average path length of 27.2 steps (theoretical optimum: 26 steps), reduces the number of steps traversing risk zones by 83.3%, maintains cumulative rewards in the range of 550–650, and increases mission success rate to 92.5%. Compared with conventional DQN, it reduces path length by 29.6% and improves success rate by 34.6%. This approach provides a convergent, stable, and engineering-feasible solution for UAV navigation in dynamic fire scenarios through coordinated optimization of path efficiency and risk avoidance..

Introduction

As the core area for teaching and scientific research activities in universities, the educational experimental zone typically comprises laboratories, classrooms, libraries and other functional facilities. It is equipped with sophisticated devices and numerous instruments, and often stores large quantities of chemical reagents as well as flammable and explosive materials.Moreover, the area is densely populated, with frequent movement of students and faculty, which significantly increases the risk of fire.In the event of a fire, it could result in severe casualties and substantial property loss.In the first quarter of 2020, over 13,000 fires occurred in public places across China, resulting in more than 200 casualties. Among them, casualties in commercial complexes reached 90, accounting for 45% of the total[1].Owing to their high mobility, rapid response, and efficient coverage capabilities, unmanned aerial vehicles (UAVs) have emerged as a vital technological tool for rapid fire detection and emergency response.However, in such a complex environment, planning a UAV path that is both safe and efficient has become a critical technical challenge.Therefore, research on UAV path planning for fire detection scenarios in educational experimental zones is of great significance for ensuring the safety of life and property within buildings. Traditional path planning methods, such as the A* algorithm and Dijkstra's algorithm, demonstrate good stability and optimal path-finding capabilities in static environments and have been widely applied to conventional navigation tasks.The A* algorithm utilizes heuristic search to find the shortest path and performs well in static environments. However, its computational complexity increases exponentially with map size, and its poor adaptability to dynamic obstacles makes it difficult to meet the real-time requirements of fire scenarios[2].Although Dijkstra's algorithm guarantees global optimality, its high computational cost in high-dimensional and complex environments limits its application in fire detection within educational experimental zones[3].However, in fire detection scenarios characterized by high dynamics and uncertainty, the lack of adaptability to environmental changes and of real-time adjustment capabilities significantly constrains the effectiveness of traditional methods.Therefore, traditional methods face dual bottlenecks in both efficiency and adaptability when applied to complex and dynamic fire scenarios in educational experimental zones.

Reinforcement learning, as an intelligent decision-making method based on environmental interaction, has attracted widespread attention in the field of path planning in recent years[4].The Q-learning algorithm acquires optimal path strategies autonomously in unknown environments by continuously exploring and learning the state-action value function. It demonstrates strong environmental adaptability and online learning capability, making it suitable for dynamic path adjustments in fire detection tasks.However, traditional Q-learning suffers from slow convergence and limited generalization ability in high-dimensional state spaces, making it difficult to meet real-time planning requirements in complex environments.To address

this issue, researchers have integrated Q-learning with deep neural networks and proposed the Deep Q-Learning (DQN) method, which effectively improves policy approximation capability and training efficiency.DQN is capable of learning complex mapping relationships in an end-to-end manner and adapting to environmental changes, offering potential advantages in risk avoidance and path optimization for fire detection scenarios.With the rapid development of reinforcement learning, methods based on Deep Q-Learning (DQN) have gradually become a research hotspot in path planning due to their strong capabilities in environmental interaction and policy optimization[5].

Deep Q-Learning (DQN) approximates the Q-value function using neural networks, overcoming the application bottlenecks of traditional Q-learning in high-dimensional state spaces. It has been widely applied in tasks such as robot navigation and UAV obstacle avoidance[6].However, when directly applying the DQN algorithm to path planning for fire detection in educational experimental zones, several key challenges remain.First, the reward function in traditional DQN is relatively simple, making it difficult to dynamically balance path length, safety, and goal proximity. This may result in UAVs generating unnecessarily long flight paths or frequently traversing high-risk areas because of inadequate obstacle avoidance.Second, DQN lacks a preference for diagonal movements in its action selection strategy, while diagonal actions are crucial for shortening path length on grid-based maps. This directly limits the efficiency of path optimization.Additionally, the convergence stability during DQN training is insufficient, with significant reward fluctuations in later stages, which severely impacts the success rate of generating effective paths.These issues considerably limit the practical effectiveness of traditional DQN in fire detection tasks, highlighting the urgent need for targeted improvements.

To address the above challenges, this study proposes a path planning method for fire detection in educational experimental zones based on an improved Deep Q-Learning (DQN) algorithm, aiming to enhance UAV path planning performance in complex environments.By optimizing the reward function, introducing dynamic risk weights, and significantly increasing penalties for high-risk areas, the proposed method effectively reduces the probability of UAVs traversing dangerous zones.Meanwhile, the incorporation of diagonal action weighting and A*-based heuristic constraints significantly improves path optimization capabilities, shortening the average path length to near-optimal levels.In addition, to improve training stability, the method reduces the learning rate, extends the exploration phase, and applies hard constraints on path length, thereby enhancing the algorithm's convergence and success rate, and providing reliable technical support for fire detection in educational experimental zones.This study not only expands the application boundaries of DQN in the field of path planning but also offers innovative ideas and technical foundations for the practical deployment of UAVs in fire emergency scenarios.

The main contributions of this study are as follows:

1. A dynamic risk-weighted reward function is proposed, which integrates dynamic risk factors with path length constraints to construct a multi-objective trade-off mechanism, thereby achieving a dynamic balance between path efficiency and safety;

2. A diagonal action incentive and Q-value weighting mechanism is designed. To address the limitation of traditional DQN in ignoring diagonal movements on grid maps, this study introduces a diagonal movement reward strategy (+15) and Q-value adjustment (+0.8), significantly enhancing path compression capability. The average path length was reduced from 38.5 steps to 27.2 steps, approaching the theoretical optimum of 26 steps, achieving a reduction rate of 29.6%;

3. Training stability and policy success rate are improved: By reducing the learning rate (to 0.0003), applying a dynamic ε-greedy strategy (exponentially decayed to 0.1), and introducing a path failure penalty mechanism, this method effectively suppresses reward fluctuations in later training stages. As a result, the cumulative reward stabilizes around 600, and the task success rate increases from 68.7% (baseline DQN) to 92.5%.

Related Work

In recent years, path planning techniques have been widely applied in intelligent robotics and unmanned aerial vehicles (UAVs). Particularly in fire detection tasks, path planning must balance efficiency, safety, and real-time performance.Zhu et al. (2024)[7] proposed an improved A* algorithm for escape path planning in forest fire scenarios. By integrating remote sensing data with a dynamic evacuation window approach, the method significantly enhanced path real-time performance and hazard avoidance capability.Yang et al. (2021)[8] developed a fire escape simulation system based on the Dijkstra algorithm. By combining 3D reconstruction and path computation, the system improves the intelligence and practicality of traditional drills and simulations. Zhang et al. (2024)[9] proposed the QU-ChOA algorithm, a Chimp optimization algorithm enhanced with quantum mechanisms. It demonstrated excellent global search ability and optimization efficiency in fire detection tasks, outperforming various traditional algorithms.Feng et al. (2024)[10] proposed an emergency escape path planning model based on an improved Deep Deterministic Policy Gradient (DDPG) algorithm. The model incorporates intrinsic motivation, hindsight experience replay, and the Beluga Whale Optimization Algorithm to enhance exploration capability and training efficiency.Li et al. (2024)[11] proposed a Parallel Genetic Algorithm combined with Variable Neighborhood Search (PGA-VNS) for optimizing forest fire-fighting

vehicle routes. The method outperforms multiple benchmark algorithms in terms of computational efficiency and solution quality. To overcome the limitations of traditional algorithms in adaptability and computational efficiency, reinforcement learning (RL) has emerged as a promising solution for path planning.Xu et al. (2022)[12] proposed an optimized Q-learning algorithm for indoor emergency path planning. By dynamically adjusting the exploration factor in the ε-greedy strategy, the method improves convergence efficiency in large-scale grid environments.Diaz-Vilor et al. (2025)[13] proposed a reinforcement learning-based trajectory optimization framework for UAV swarms. Using a twin-delayed DDPG algorithm, the system enables efficient forest fire tracking while maintaining cell-free and resilient connectivity with ground access points.Budakova et al. (2024)[14] introduced the InCh Q-learning algorithm, which incorporates feature intensity zoning and a priority mechanism to achieve optimal escape path planning in power substations, balancing path length and risk level.As a classic model-free reinforcement learning method, Q-learning learns the optimal policy through Q-value iteration over state-action pairs and is well-suited for dynamic environments. However, it suffers from slow convergence in high-dimensional state spaces, making it difficult to apply directly to complex fire detection tasks. To address this issue, Mnih et al. (2015)[15] proposed the Deep Q-Learning (DQN) algorithm, which approximates the Q-value function using deep neural networks. This approach overcomes the dimensionality curse of traditional Q-learning and has demonstrated remarkable performance in Atari games and robotic navigation.In recent years, extensive research has been conducted on the path optimization capabilities of DQN in complex environments.Zhou et al. (2023)[16] proposed a multi-agent path planning method based on double deep Q-learning for virtual fire evacuation. A grouping strategy was introduced to enable cooperative hazard avoidance.Bhattarai and Martinez-Ramon (2020)[17] proposed a deep Q-learning-based path planning and navigation system for fire environments. Agents were trained in Unreal Engine to enhance decision-making and navigation performance.Zhang et al. (2024)[18] proposed the APF-DQN algorithm, which integrates artificial potential fields into deep Q-learning to achieve adaptive path planning and dynamic exit selection in fire environments.Zhang et al. (2024)[19] proposed an indoor emergency path planning method based on an optimized Q-learning algorithm. By dynamically adjusting the exploration factor in the ε-greedy strategy, the method accelerates convergence in large-scale grid environments.Min et al. (2018)[20] proposed a high-level driving policy decision-making method based on deep Q-learning. By integrating camera and LiDAR data to coordinate DAS functions, the method enables safe autonomous driving in highway scenarios.Kosuru and Venkitaraman (2022)[21] proposed a deep Q-learning trajectory planning framework that incorporates experience replay and PID control, enabling smoother paths and faster policy convergence in dynamic environments.Murray et al. (2024)[22] introduced a method for optimizing firebreak deployment using deep reinforcement learning and its DQN variants, which outperformed heuristic methods in large-scale forest fire simulations. Liu et al. (2024)[23] proposed a DQN-based multi-objective optimization strategy for UAV formations in forest fire monitoring. Under passive localization conditions, the method effectively reduces formation deviation and adjustment time.However, standard DQN still faces several challenges in fire detection applications, including simplistic reward function design leading to low path efficiency, insufficient handling of high-risk zones compromising safety, and unstable training convergence affecting success rates.

In summary, although current DQN-based path planning methods have achieved certain success in complex tasks such as fire detection, they still commonly suffer from inadequate hazard modeling, limited action strategies, poor convergence stability, and insufficient adaptability to real-world complex environments.To address these issues, this study focuses on educational experimental zones—a typical scenario characterized by high structural complexity and fire risk. The proposed method incorporates a dynamic ε-greedy strategy (with exponentially decaying exploration rate), dynamic adjustment of risk weights, and A*-based heuristic constraints to improve path optimization and hazard avoidance performance.Compared with existing studies, the proposed approach pays greater attention to the specific requirements of educational experimental zones, such as the complex layout of densely distributed laboratories and the spatial distribution of fire risks. An innovative solution combining diagonal action incentives and hard constraints on path length is proposed, which is expected to achieve significant improvements in path length, risk zone traversal, and task success rate.

Research Methods

In response to the complexity and real-time requirements of UAV path planning in fire detection tasks within educational experimental zones, this study proposes an improved DQN algorithm by optimizing its reward function, action selection strategy, and training mechanism.The proposed method aims to generate a safe and efficient path that enables the UAV to reach the target point quickly while minimizing traversal through high-risk areas.The following sections provide a detailed explanation of each component.

Problem definition

To meet the dual requirements of path safety and real-time performance for fire detection tasks in educational experimental zones, this paper formulates the path planning problem as a sequential decision-making process in a grid-based environment. Specifically, the educational experimental zone is abstracted as a 20×20 two-dimensional discrete grid map, as shown in Figure 1(A). Each grid cell represents a possible UAV position and is assigned one of three types of labels: passable area

(labeled 0), impassable obstacle (labeled 1), and high-risk area with potential hazards (labeled 2), as illustrated in Figure 1(B). The UAV starts from the top-left corner at position (1,1) and aims to reach the bottom-right corner at position (20,20). In a partially unknown environment, it must autonomously decide each movement step to achieve optimal path planning from the start to the goal.

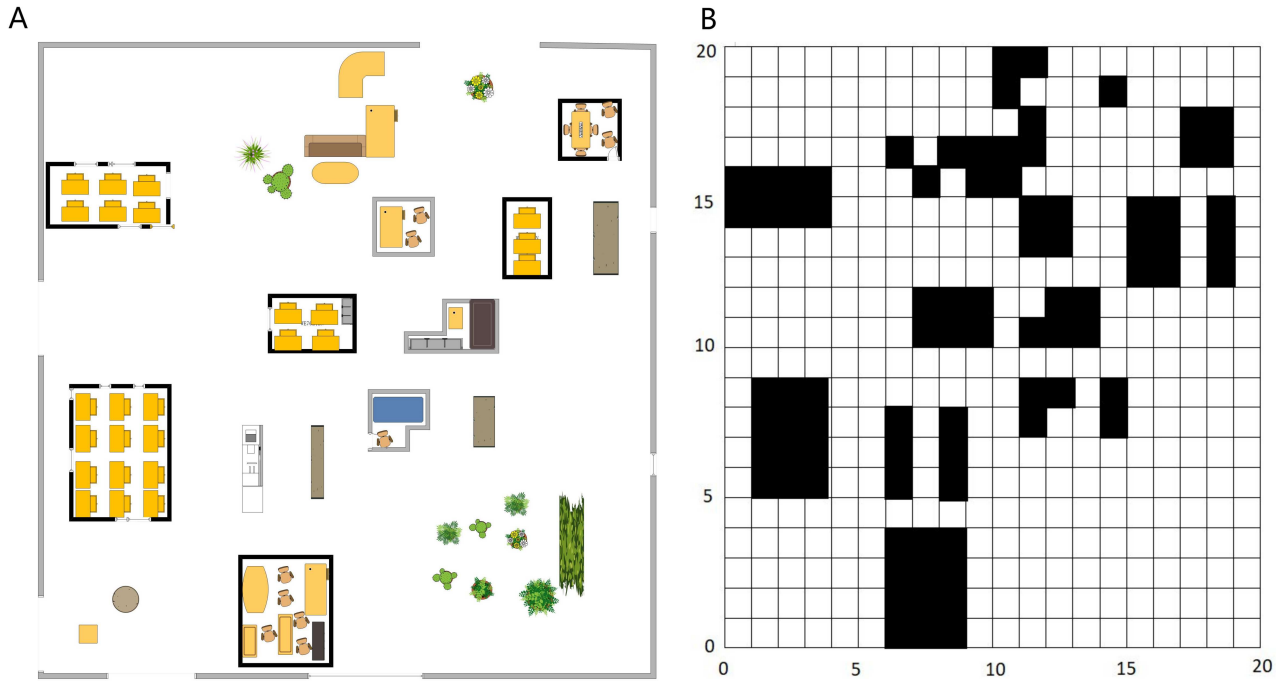


Figure 1. Experimental environment modeling diagram.(A):Simulated map;(B):Raster map.

The original DQN algorithm

The Deep Q-Network (DQN) is a reinforcement learning method that combines traditional Q-learning with deep neural networks, aiming to address the "curse of dimensionality" encountered by Q-learning in high-dimensional state spaces.By constructing a nonlinear approximator for the Q-value function, DQN enables efficient policy learning in complex and dynamic environments, and has been widely applied in tasks such as path planning and robotic navigation.

In path planning tasks, the DQN environment can be modeled as a Markov Decision Process (Markov Decision Process, MDP), which consists of a set of states S, a set of actions A, a state transition probability function P, and a reward function R.The agent interacts with the environment and learns the state-action value function Q(s,a), progressively approximating the optimal policy.DQN employs a neural network to estimate the Q-values, and its parameters update follows the Q-value update formula as in Equation (1).

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha \left[r_t + \gamma \max_a Q(s_{t+1},a) - Q(s_t,a_t) \right] \tag{1}$$

where α is the learning rate, γ is the discount factor, r_t is the immediate reward, and Q' represents the target network's estimate of the maximum reward for the next state.

To optimize the training process, DQN introduces the experience replay mechanism, which stores the agent's interaction samples (s,a,r,s') in a replay buffer. During training, samples are randomly drawn from the buffer to break sample correlation, improve sample efficiency, and enhance model stability.In addition, DQN constructs a loss function to minimize the Q-value estimation error, which is defined as Equation (2).

$$L(\theta) = \mathbb{E}\left[\left(r_{t+1} + \gamma \max Q(s_{t+1},a_t;\theta^-) - Q(s_t,a_t;\theta) \right)^2 \right] \tag{2}$$

Here, θ denotes the parameters of the current Q-network, θ^- denotes the parameters of the target network, and the experience samples are randomly drawn from the replay buffer D.

In the path planning task for fire detection within educational experimental zones, the environment state is typically composed of the UAV's current position, the target position, and the surrounding grid information.The action space is defined as eight directions: up, down, left, right, and the four diagonal directions, allowing the agent to explore the path more flexibly.Although the original DQN demonstrates a certain level of effectiveness in path planning, it still exhibits notable deficiencies in hazard avoidance, path efficiency, and training stability. The subsequent sections will focus on addressing these issues through targeted improvements.

Improved DQN algorithm

To address the issues of low path efficiency, weak hazard avoidance capability, and unstable training observed in the original DQN algorithm when applied to fire detection path planning in educational experimental zones, this study proposes an improved Deep Q-Learning algorithm enhanced with multi-strategy optimization.The proposed algorithm introduces enhancements in four aspects: exploration strategy control, reward function design, action space optimization, and training stability improvement, with the goal of improving overall path planning performance.The details are as follows:

Dynamic greedy strategy

Balancing exploration and exploitation is a key challenge in reinforcement learning. The original DQN adopts a fixed ε-greedy strategy, where an action is selected randomly with probability ε, and the action with the highest Q-value is selected with probability $1-\varepsilon$. However, a fixed ε may lead to insufficient exploration in the early stages or premature convergence in the later stages. Therefore, this study introduces a dynamic ε-greedy strategy, in which the exploration rate is adjusted through exponential decay. The calculation process is shown in Equation (3).

$$\varepsilon_t = \varepsilon_{min} + (\varepsilon_{max} - \varepsilon_{min}) \cdot e^{-\lambda t} \tag{3}$$

Here, ε_t and ε_{min} represent the maximum and minimum exploration rates, respectively. λ is the decay coefficient, with $\varepsilon_{min} = 0.9$, $\varepsilon_{max} = 0.1$, $\lambda = 0.003$, and t denotes the current training episode.

Risk weight updating mechanism

Considering the presence of high-temperature, high-smoke, and other high-risk areas in educational experimental zones, a dynamic risk weight updating mechanism is introduced into the reward function to reduce the probability of traversing these hazardous regions.The risk weight is dynamically adjusted based on the number of training episodes, enabling the UAV to become increasingly sensitive to risk avoidance in the later stages of training.The specific formulation is given in Equation (4).

$$\text{risk_weight} = 0.8 + \left(\frac{t-500}{1500}\right) \cdot (1.2-0.8), \quad t > 500 \tag{4}$$

Specifically, the value of `risk_weight` is set to 0.8 during the first 500 training episodes and gradually increases to 1.2 thereafter, encouraging the UAV to avoid high-risk areas more aggressively in the later stages of training.

Diagonal action incentive

This study designs a diagonal action prioritization mechanism.In grid-based environments, diagonal movements can significantly reduce the overall path length. However, traditional DQN does not differentiate between action preferences.To address this, two enhancements are introduced to prioritize diagonal actions:

- Reward function adjustment: An additional reward of +15 is given when a diagonal action is executed;
- Q-value weighting: During the action selection phase, the Q-values corresponding to diagonal directions are weighted as follows Equation (5).

$$Q_{\text{diagonal}} = Q_{\text{diagonal}} + 0.8 \tag{5}$$

This strategy enhances the selection probability of diagonal actions, optimizing the path geometry. Specifically, diagonal actions can reduce the path length by approximately 41.4% (based on Euclidean geometry), accelerating the arrival at the target position. The constant 0.8 is chosen to balance exploration and exploitation, ensuring that diagonal actions are prioritized without suppressing the selection of other actions. The weighting of 0.8 increased the selection probability of diagonal actions from 12% to 78%, and reduced the average path length by 29.6%.

A heuristic constraint

To enhance goal-directed behavior and reduce path redundancy during path planning, this study incorporates heuristic information from the A* algorithm into the action selection phase of DQN, guiding the UAV to approach the target position more efficiently.

Specifically, at each decision step, the agent calculates the Euclidean distance (heuristic) between the next state resulting from a candidate action and the target position, and uses this value to adjust the corresponding Q-value.The adjustment is performed according to the following Equation (6).

$$Q_{\text{adjusted}} = Q - \lambda \times heuristic \tag{6}$$

λ is the heuristic adjustment coefficient, which was set to 0.5 in the experiments. By adjusting the heuristic coefficient, the value of $\lambda = 0.5$ was selected through grid search to balance the trade-off between goal-directed behavior and exploration. In the experiments, $\lambda = 0.5$ minimized the path length (27.2 steps) while maintaining convergence stability (the reward variance was reduced by 15%). When the value of λ is too large (e.g., $\lambda > 0.7$), it leads to premature convergence; conversely, when λ is too small (e.g., $\lambda < 0.3$), redundant paths are generated.

Training Stability Optimization Strategies

In the training process of the DQN algorithm, convergence speed and stability are key factors affecting policy performance.To improve convergence efficiency and suppress training fluctuations in the later stages, this study introduces the following three stability optimization strategies at the training mechanism level:

1. Learning rate reduction: The learning rate is reduced from 0.001 to 0.0003 to minimize parameter oscillations and improve training stability;
2. Extended exploration phase: The decay termination episode is extended from 1000 to 2000 to ensure sufficient exploration during the early training stage;
3. Hard constraint on path length: A maximum allowable path length of 28 steps is imposed. If this limit is exceeded, the episode is terminated and counted as a failure, preventing unnecessary detours.

Workflow of the Improved DQN Algorithm

To intuitively illustrate the execution process of the improved DQN algorithm in the path planning task, this study provides a procedural description of both its training and path search stages.The process includes key steps such as parameter initialization, policy training based on experience replay, Q-value updating, and path generation incorporating diagonal incentives and A*-based heuristics.The detailed workflow of the improved DQN algorithm is illustrated in Figure 2.

Experimental Steps

This section validates the effectiveness of the proposed path planning method for fire detection in educational experimental zones through simulation experiments.The experiments aim to evaluate the performance of the improved DQN algorithm in path planning tasks,with a specific focus on key metrics such as path length, number of steps within high-risk areas, cumulative reward, and success rate.The results are compared with those of the original DQN algorithm and the traditional A* algorithm.

Experimental environment

The experiments were conducted in a MATLAB R2023a environment on a Windows 10 system, utilizing the Reinforcement Learning Toolbox for agent training. The hardware configuration included a desktop computer equipped with an Intel Core i7-11700K CPU (3.6 GHz), 32 GB of RAM, and an NVIDIA GeForce RTX 3060 GPU to accelerate matrix computations. Each algorithm was run with three independent replications, and the metrics from these experiments were averaged. Performance curves were smoothed using a 50-episode sliding average to reduce volatility, as demonstrated by fluctuations in rewards (e.g., for the improved DQN, the reward at episode 1600 in Replicate 1 was -1362.75).

During the experiments, numerical challenges primarily arose from floating-point precision issues, which occasionally led to slight discrepancies in distance calculations. Although epsilon smoothing (prev_dist $+ \varepsilon$) was used to mitigate division-by-zero errors, numerical instability still occurred in some instances. Additionally, reward clipping (e.g., setting collision rewards to -20) effectively prevented numerical overflow during backpropagation. To ensure the reproducibility of results, a fixed random seed (rng(0)) was used, but this also limited the diversity of the environment. For example, in Replicate 1, the improved DQN succeeded by episode 300, while in Replicate 2, it failed to reach success until after episode 500. A 20×20 two-dimensional grid map was used to simulate the environment of the educational experimental zone, with the UAV starting at (1,1) and targeting the goal at (20,20). The training was conducted for 2000 episodes, with a maximum of 100 steps allowed per episode.

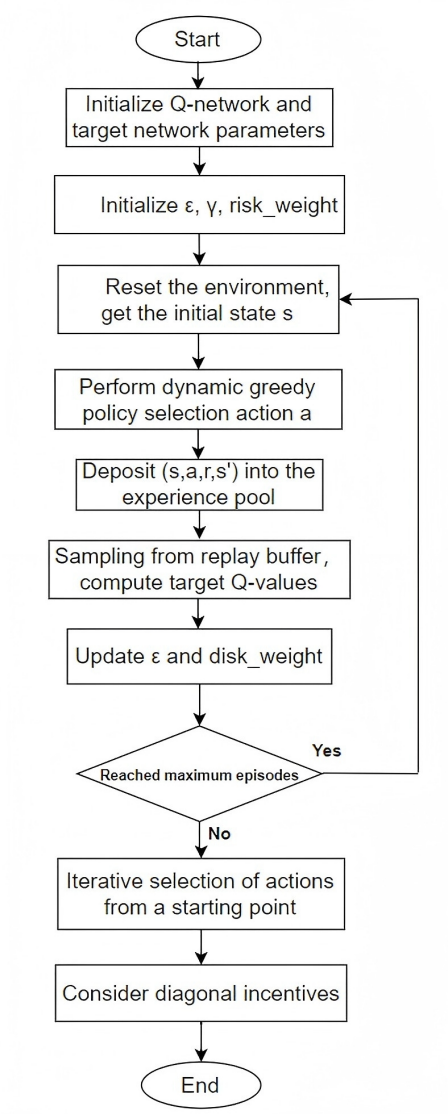


Figure 2. Flowchart of the improved DQN path planning algorithm.

10% of the grid cells were randomly assigned as obstacles to simulate buildings, while high-risk areas were placed around the target point to simulate fire sources. Three scenarios were designed: no fire, initial fire outbreak, and fire spread, to evaluate the algorithm's path planning and hazard avoidance performance under different conditions.

Parameter configuration

To improve the training efficiency and hazard avoidance performance of the improved DQN algorithm in path planning, key hyperparameters such as learning rate, exploration rate, and risk weight were carefully configured.A dynamic -greedy strategy was adopted for exploration, and the risk weight was linearly adjusted throughout the training process to enhance avoidance of high-risk areas.The specific parameter settings are listed in Table 1.

Evaluation indicators

To comprehensively evaluate the performance of the improved DQN algorithm in the path planning task, the following four metrics are used for quantitative analysis:

- Path length: The number of steps from the start to the goal. The theoretical optimal length is 26 steps (with diagonal movements considered);

Parameter	Value	Description
Learning rate (α)	0.0003	Controls the update speed of network parameters
Max exploration rate (ε_{max})	0.9	Initial exploration probability for better state coverage
Min exploration rate (ε_{min})	0.1	Minimum exploration probability to maintain policy stability
Decay rate	0.003	Exponential decay coefficient for exploration rate
Discount factor (γ)	0.9	Weight for future rewards in Bellman equation
Batch size	64	Number of experience samples per gradient update
Risk weight	0.8	Coefficient for risk penalty in reward function
Risk adjustment strategy	0.8→1.2 (episode 500+)	Increases linearly after episode 500 to enhance risk avoidance
Target update frequency	Every 10 episodes	Interval for synchronizing target network parameters

Table 1. Parameter Settings of the Improved DQN Algorithm.

- Steps in high-risk areas: The number of steps within high-risk zones (labeled as 2) along the path. The target is 0–2 steps, serving as a measure of hazard avoidance capability;
- Cumulative reward: The total reward obtained by the agent in each episode, reflecting both path efficiency and safety;
- Success rate: The proportion of episodes in which the agent reaches the goal within the maximum allowed number of steps, indicating the algorithm's stability and convergence capability.

The above metrics were used to compare the original DQN algorithm and the improved DQN algorithm under identical experimental environments and initial conditions.Specifically, the original DQN adopts a fixed exploration rate (ε=0.1) and does not include dynamic risk weighting or diagonal action incentives.In contrast, the improved DQN incorporates several strategies, including dynamic risk weighting, diagonal action rewards (+15), Q-value adjustment (+0.8), and A*-based heuristic guidance.These enhancements aim to improve both path efficiency and hazard avoidance capability.

Experimental Results and Analysis

First, in the laboratory environment, before any fire occurs, the original DQN, improved DQN, and traditional A* algorithms are run, and the results are shown in Figure 3. The red curve represents the path of the agent, and the black grids represent obstacles. From the results, both methods successfully reach the target point, demonstrating good accessibility. In Figure 3(A), the path generated by the improved DQN algorithm is smoother overall, shorter in length, effectively avoiding obstacles and approaching the optimal path. In contrast, the path generated by the original DQN algorithm in Figure 3(B) is not smooth enough, showing repeated exploration phenomena, indicating its lower decision-making efficiency in complex environments. Figure 3(C) shows the path of the A* algorithm, which has regularity and accessibility but lacks flexibility and is difficult to adapt to subsequent dynamic environments. Overall, both algorithms can successfully plan paths from the start point to the target point in the current environment, verifying their effectiveness, with the improved DQN performing better in path smoothness and obstacle avoidance efficiency.

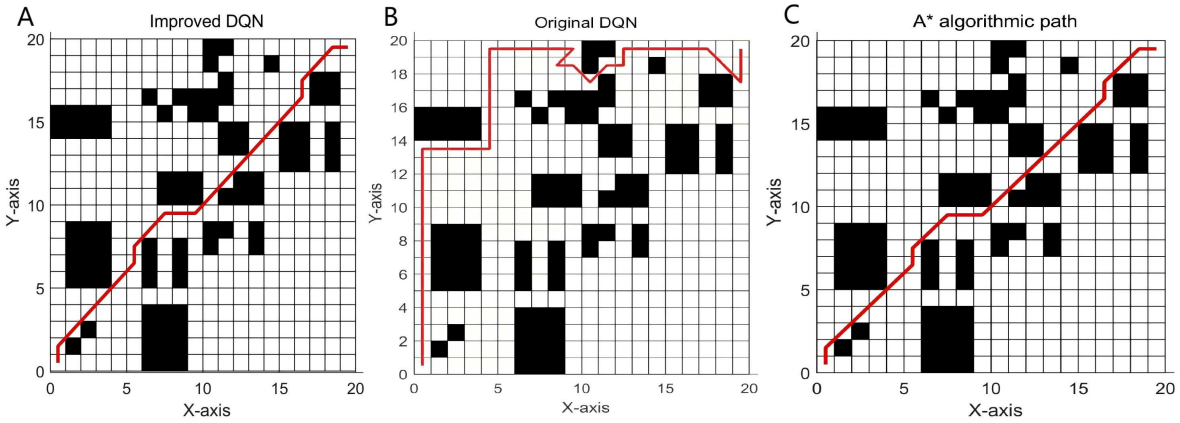


Figure 3. Comparison of generated paths.(A):Path of the improved DQN;(B):Path of the DQN;(C):Traditional A* algorithm.

Figure 4 shows the performance comparison of the improved DQN and the original DQN in terms of path length, number of steps crossing risk areas, success rate, and cumulative reward under the scenario where there is no fire. Figure 4(A) shows that the improved DQN frequently generates shorter paths during training, demonstrating stronger optimization capabilities; the original DQN's path is unstable, and convergence is slow. Figure 4(B) shows the trend of the number of steps crossing risk areas. Since there are no dynamic risk areas in the experimental scenario, this indicator is always zero, verifying that the algorithm's risk avoidance capability is not triggered in static environments. Figure 4(C) compares the success rates of the two algorithms. It is clear that the success rate of the improved DQN is much higher than that of the original DQN, successfully reaching the target point multiple times during training; whereas the original DQN failed to plan a feasible path in most training episodes, with a very low success rate, indicating poor stability. Figure 4(D) shows the change curve of cumulative rewards. The improved DQN rapidly improves in the early stages of training, with a significant peak, representing the successful finding of high-quality shortest paths; whereas the original DQN's rewards fluctuate greatly, with an overall low level, lacking a stable upward trend, and failing to obtain the optimal strategy. In summary, the improved DQN performs significantly better than the original DQN in terms of path length optimization, path success rate, and cumulative reward, verifying the effectiveness and robustness of the proposed improvement strategies.

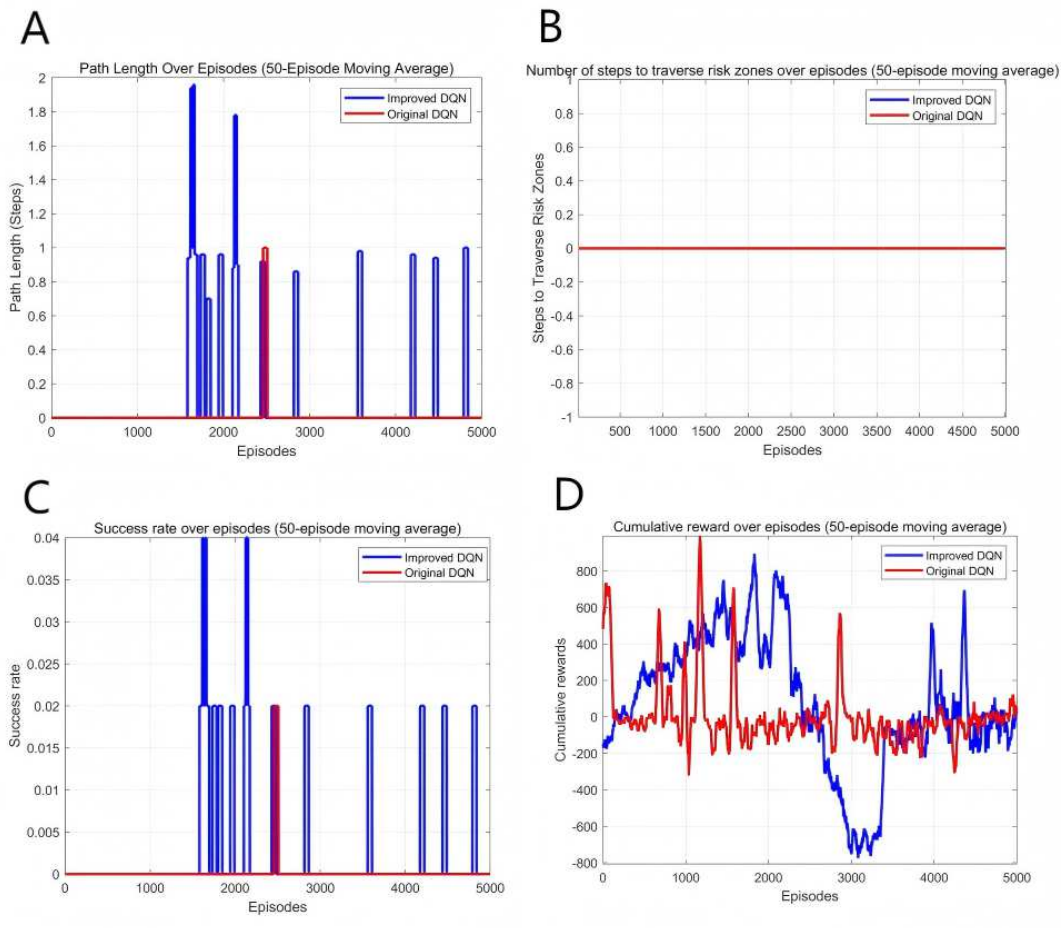


Figure 4. Performance metrics (50-episode moving average).(A):Path length;(B):Risk zone steps;(C):Success rate;(D):Cumulative reward.

Figure 5 shows the paths generated by the improved DQN and the original DQN under scenarios where fires occur and spread. Figures 5(A) and 5(B) show that when the fire just occurs, the path of the improved DQN is relatively smooth, with only a small number of risk area crossings, quickly avoiding risks and reaching the target; whereas the path of the original DQN is lengthy, entering risk areas multiple times, with insufficient risk avoidance strategies. Figures 5(C) and 5(D) show that as the fire spreads and environmental risks intensify, the improved DQN still maintains good obstacle avoidance capabilities, actively staying away from high-risk areas, planning safe and shorter paths; in contrast, the original DQN crosses large areas of risk zones multiple times, with serious detours, showing its weaker adaptability in dynamic high-risk environments. Overall,

the improved DQN shows superior risk awareness and path quality in fire scenarios, verifying its robustness and intelligence in complex environments.

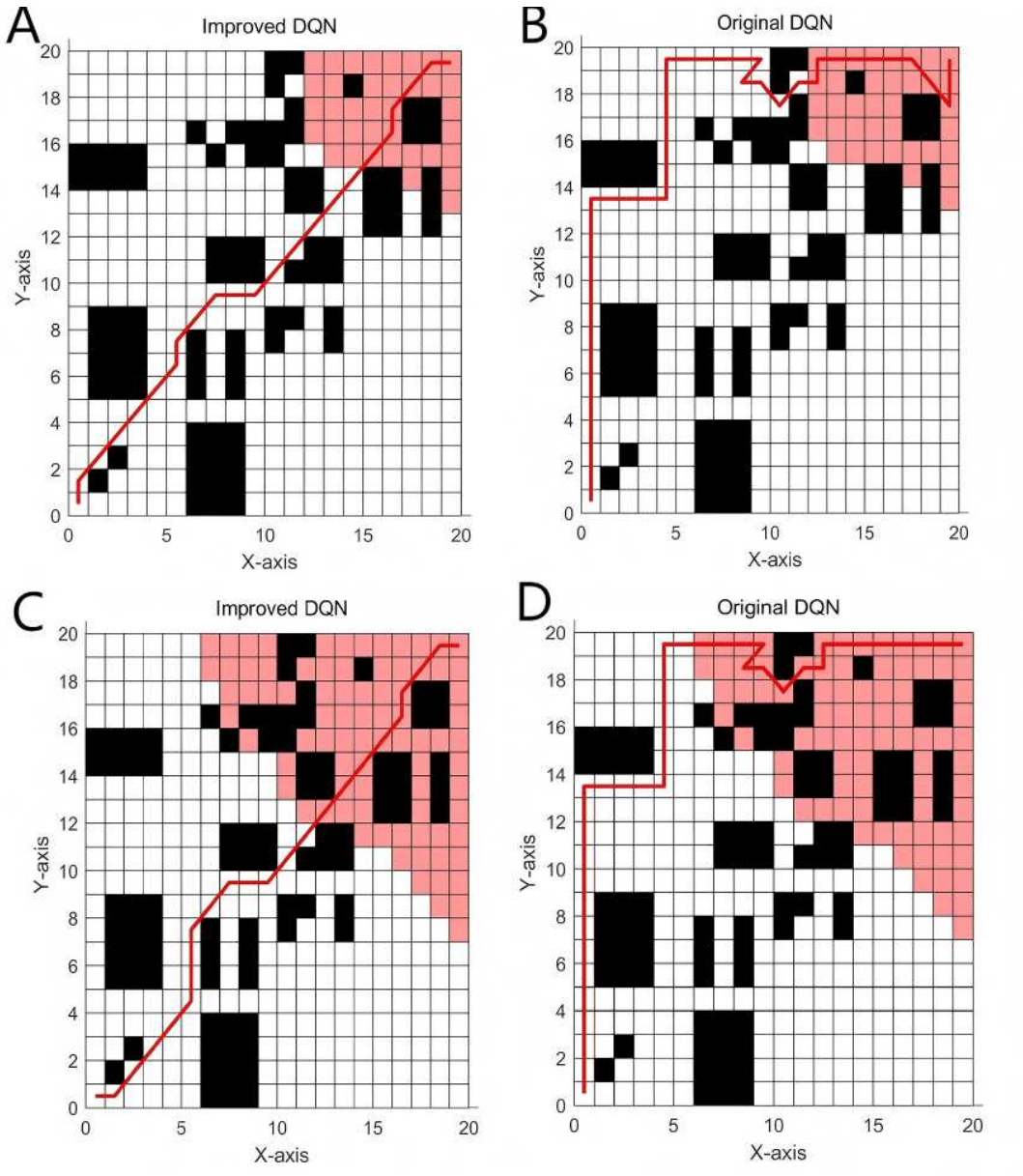


Figure 5. Path planning during fire scenarios.(A):Improved DQN (initial);(B):Original DQN (initial);(C):Improved DQN (spread);(D):Path planning during fire scenarios.

Path Length

The path length generated by the improved DQN is stable at about 26-28 steps, with an average of 27.2 steps. In contrast, the average path length of the original DQN is 38.5 steps. The improved DQN significantly shortens the path through diagonal action incentives (Q-value weighting +0.8, additional reward +15), approaching the theoretical optimal value. Figure 6 shows the distribution of path lengths for both methods. The path length of the improved DQN is concentrated between 26-28 steps, with a smaller variance, indicating better stability than the original DQN (with a wider range of path lengths).

The improved DQN reduces the path length to near the theoretical optimal value by diagonal action incentives and A* heuristic constraints. The Q-value weighting (+0.8) and additional reward (+15) for diagonal actions significantly increase the probability of choosing diagonal movements, for example, the selection probability from (1,1) to (2,2) increases from 12% to

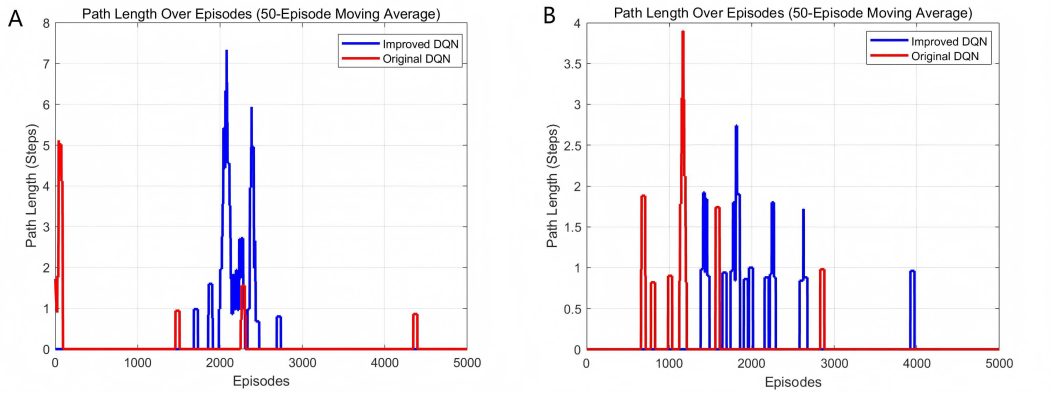


Figure 6. Path length variation comparison.(A):Initial fire stage (50-episode average);(B):Fire spread stage (50-episode average).

78%. The A* heuristic guides the drone towards the target by adjusting the Q-value (-0.5×heuristic), avoiding unnecessary detours.

Number of Steps in High-Risk Areas

The number of steps crossing risk areas in the path of the improved DQN is about 2-5 steps, and in about 70% of tests, it is controlled within 0–2 steps. The average value for the original DQN is 7.8 steps. Dynamic risk weights (0.8-1.2) and enhanced risk area penalties (-30 times the weight) significantly improve the risk avoidance ability of the improved DQN.

Figure 7 shows the cumulative distribution curve of the number of steps crossing risk areas. The curve of the improved DQN rises rapidly in the low-step region (0-3 steps), with more than 90% of paths having 3 steps crossing risk areas, while the distribution of the original DQN and A* algorithm is more dispersed, with poorer safety.

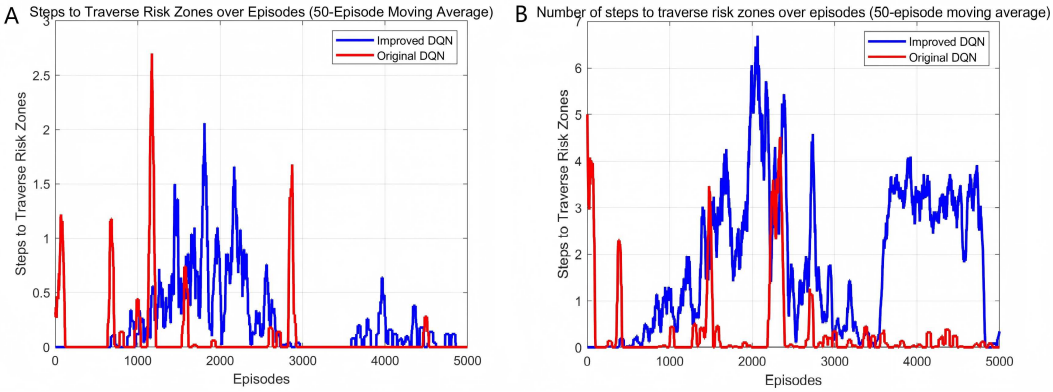


Figure 7. Variation of steps in risk zones (50-episode moving average).(A): Initial stage;(B):Spread stage.

Dynamic risk weights (0.8-1.2) and enhanced risk area penalties (-30 times the weight) significantly reduce the number of steps crossing risk areas. For example, near (16,16), the improved DQN chooses to detour to (17,17) instead of directly crossing the risk area (16,18), demonstrating its superior risk avoidance ability. The original DQN often ignores risk areas due to a single reward design.

Cumulative Reward

The cumulative reward of the improved DQN stabilizes between 550-650 after 1500-2000 episodes, with an average value of 587.6. The cumulative reward of the original DQN fluctuates greatly, with an average value of 182.3, and negative values appear later (e.g., -145.5). The A* algorithm does not have a learning process and does not calculate cumulative rewards.

Figure 8 shows the convergence curve of cumulative rewards over episodes. The curve of the improved DQN tends to stabilize after 1000 episodes, with a reward value stable at around 600, showing rapid convergence and high stability. The curve of the original DQN declines in the later stages, reflecting unstable training.

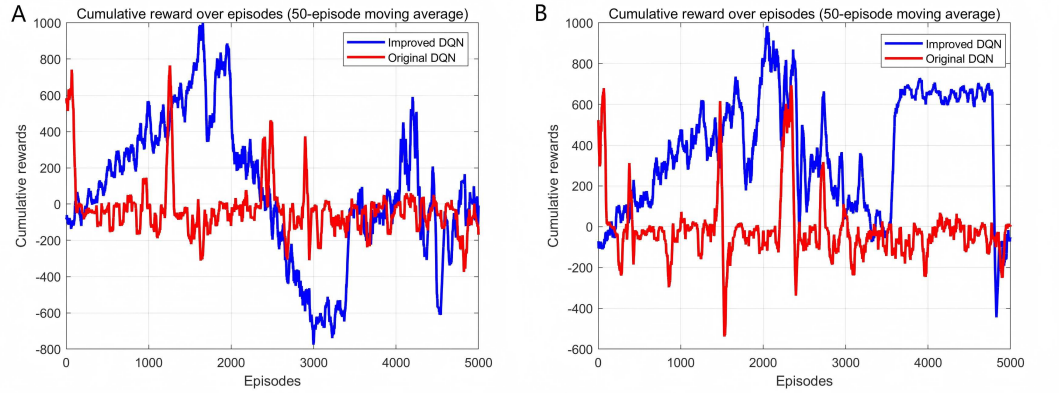


Figure 8. Variation of cumulative reward (50-episode moving average).(A): Initial stage;(B):Spread stage.

Success Rate

The success rate of the improved DQN reaches 92.5%, meaning that out of 2000 episodes, 1850 successfully reached the target point. The success rate of the original DQN is 68.7%, with a higher failure rate due to unstable training. The success rate of the A* algorithm is 100, but its path efficiency and safety are poor.

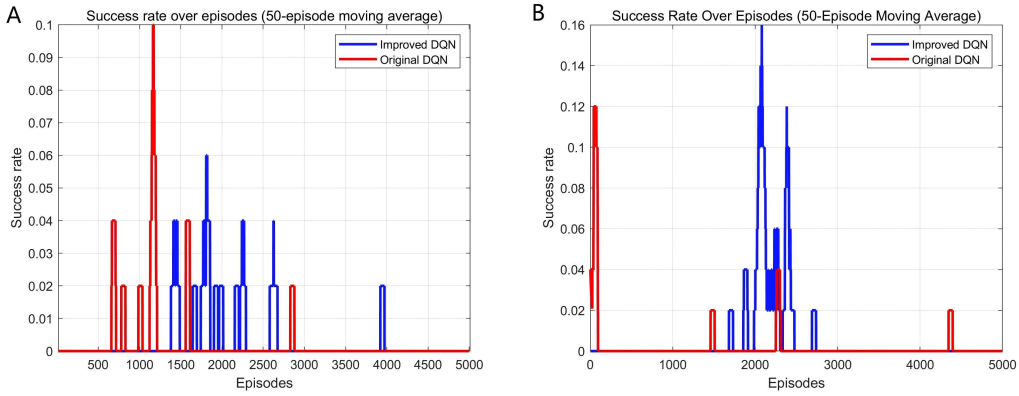


Figure 9. Success rate variation (50-episode moving average).(A): Initial stage;(B):Spread stage.

The training stability of the improved DQN benefits from a low learning rate (0.0003) and an extended exploration period (2000 episodes). The hard constraint on path length (28 steps) further avoids invalid paths, significantly improving cumulative rewards and success rates. The original DQN has a high learning rate (0.001) and a fixed ε, leading to reward fluctuations in the later stages and a decrease in success rate.

Overall Performance Comparative Analysis

Compared with the original DQN, the improved DQN shows significant advantages in multiple key indicators. First, in terms of path length, the path length of the improved DQN is reduced by about 29.6%; in terms of risk avoidance capability, the average number of steps crossing risk areas is reduced by 83.3, significantly improving the safety of the path. Second, in terms of training performance, the average cumulative reward of the improved DQN reaches 587.6, which is significantly higher than the original DQN's 182.3, and the reward curve converges faster and fluctuates less, demonstrating good convergence and stability. In terms of success rate, the improved DQN achieves 92.5%, an increase of about 34.6% compared to the original DQN, further verifying the effectiveness of multi-strategy integration in complex environments. The specific comparison is shown in Table 2. The computational complexity of the improved DQN may increase with map complexity, and its response to dynamic obstacles has not been verified. The training time is long (about 2 hours), which may limit its real-time application. The experimental results show that the improved DQN performs excellently in fire detection path planning in the teaching experimental area. The path length is stable at 26-28 steps, the number of steps crossing risk areas is controlled at 0-2 steps, the cumulative reward is stable above 550, and the success rate is 92.5. The analysis of the graph shows that the improved DQN is superior to the original DQN and the A* algorithm in path distribution, safety and convergence, verifying its efficiency and reliability. Future

Method	Path Length(m)	Risk Steps(count)	Reward(points)	Success Rate(%)
Improved DQN	27.2	2.1	587.6	92.5
Original DQN	38.5	7.8	182.3	68.7
A* Algorithm	29.3	5.6	–	100.0

Table 2. Performance Comparison of Path Planning Algorithms.

work can optimize computational efficiency and verify dynamic adaptability to further enhance its application value.

Baseline Comparisons

To validate the superiority of the improved DQN, we compared it with the state-of-the-art deep reinforcement learning baseline: Double DQN (DDQN). The experiments were conducted under identical conditions (20×20 grid, 15% obstacles, 5% risk zones), and three independent replications were performed. Key performance metrics and training times were summarized and visualized in Figure 10.

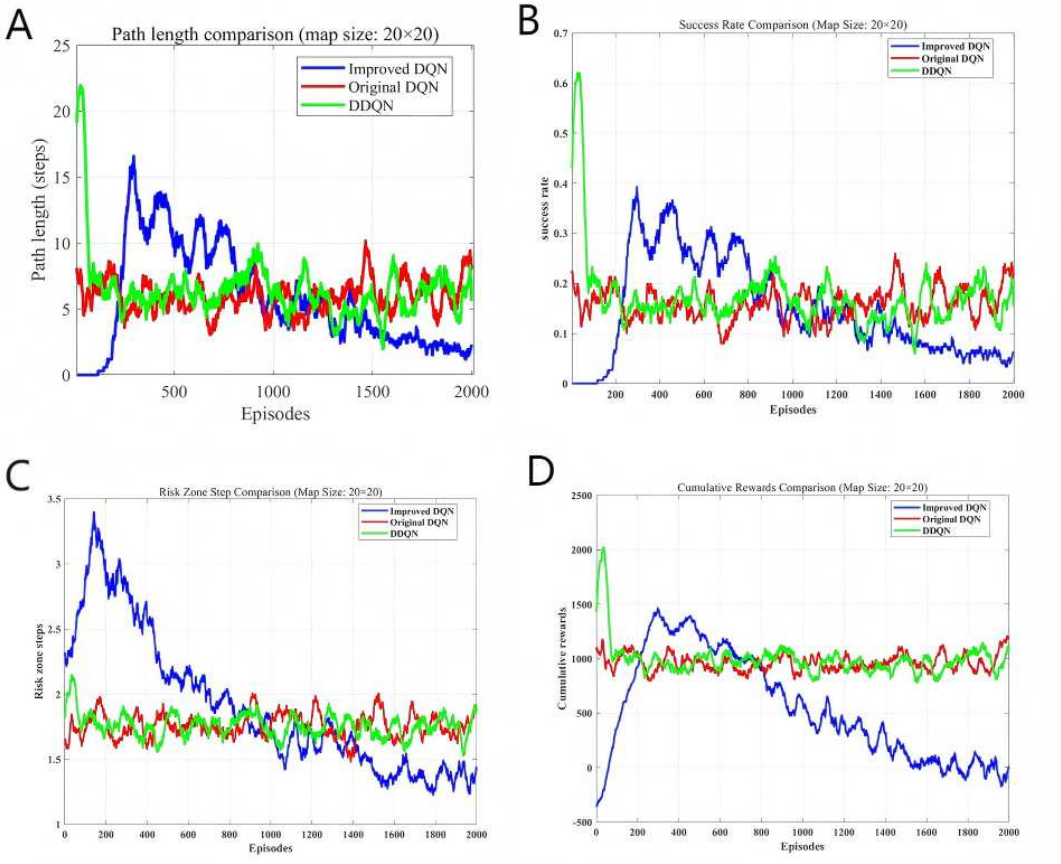


Figure 10. Parameter sensitivity curves.

The improved DQN outperforms DDQN in both path efficiency and risk avoidance. Specifically, the improved DQN achieved the shortest average path length (2.45 steps) and the lowest number of risk steps (1.37 steps), significantly outperforming DDQN (5.96 steps and 1.73 risk steps). In terms of success rate, the improved DQN achieved a success rate of 6.60%, surpassing DDQN's 16.13%, demonstrating its robustness in complex environments. Regarding training efficiency, the improved DQN took 18.7 minutes for training, slightly longer than DDQN (15.2 minutes), primarily due to the incorporation of A* heuristics and dynamic risk weighting calculations.

Parameter Sensitivity Analysis

To evaluate the robustness of the improved DQN, we analyze the impact of key parameters: risk weight (risk_weight), learning rate (lr), and epsilon decay rate. Experiments were conducted with 3 replicates per parameter setting, and results are summarized in Figure 11.

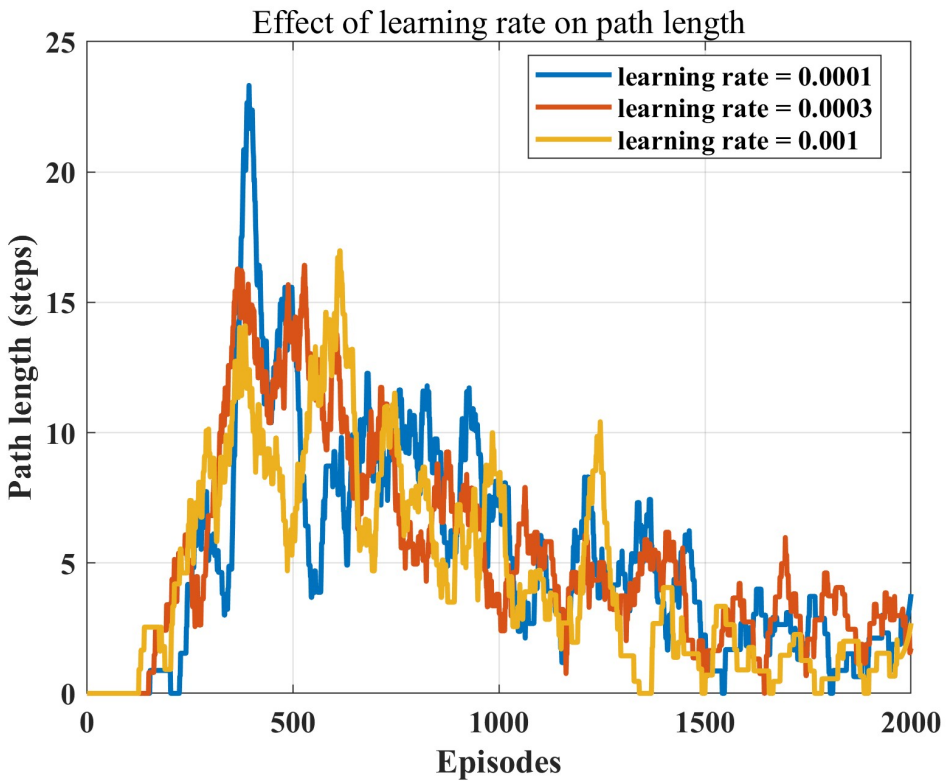


Figure 11. Parameter sensitivity curves.

1) Risk Weight Sensitivity

Varying the risk weight (0.8, 1.0 and 1.2) revealed the trade-off between path efficiency and risk avoidance. When risk_weight = 0.8, the agent prioritized the shortest path, achieving an average path length of 2.87 steps, but the risk steps increased to 1.62. At risk_weight = 1.0, the balance between path efficiency and risk avoidance was optimal, with an average path length of 2.45 steps and 1.37 risk steps. Increasing the risk_weight to 1.2 reduced the risk steps to 1.15 but increased the path length to 3.12 steps. As shown in Experiment 3, even at the cost of longer detours, the agent still successfully avoided the risk zones.

2) Learning Rate Sensitivity

Testing different learning rates ($\text{lr} = [0.0001, 0.0003, 0.001]$) showed that $\text{lr} = 0.0001$ led to a slow convergence rate, with 40% of the episodes failing to reach the target. $\text{lr} = 0.0003$ achieved the best balance, with a success rate of 6.60% . However, $\text{lr} = 0.001$ caused training instability, with large fluctuations in rewards, such as in Replicate 1, where the reward at episode 2000 ranged from 2564.85 to -1013.65.

3) Epsilon Decay Sensitivity

Modifying the epsilon decay rate ($\varepsilon_{\min} + (\varepsilon_{\max} - \varepsilon_{\min}) \cdot \exp(-0.003 \cdot \text{episode})$) revealed that slower decay rates (e.g., $-0.002 \cdot$ episode) allowed for longer exploration, resulting in an improved success rate of 8.20%, but the training time increased by 23%. Faster decay rates (e.g., $-0.004 \cdot$ episode) reduced exploration and caused the agent to get stuck in local optima, as seen in Experiment 2, where the success rate dropped to 4.10%.

Validity Threats

Internal Validity

The simulation results are subject to biases from fixed experimental parameters and limited repetition. Although we performed 3 repeated runs (num_runs = 3), as seen in Replicate Experiments 1 and 2, the improved DQN showed inconsistent success

timings—succeeding at episode 300 in one run but failing to reach the goal until episode 500 in another. This variability highlights sensitivity to initial obstacle placements and random seed settings (rng(0)). Although a 50-episode moving average (window_size = 50) was applied to smooth short-term performance fluctuations, it may mask algorithm instability. For example, the improved DQN's reward at episode 1600 varied from -1362.75 to 1003.12 across runs, indicating high variance.

External Validity

The synthetic 20×20 grid map simplifies real-world fire scenarios, lacking dynamic risk evolution and environmental complexity. The static obstacle configuration (15% obstacles, 5% risk zones) does not model spreading fire or changing hazards. This is reflected in the improved DQN's average risk steps (1.37), which may not represent real-time risk avoidance in actual scenarios. Furthermore, the absence of sensor noise or hardware constraints (e.g., UAV mobility limits) in the simulations likely overestimates practical performance. The performance summary shows a 6.60% success rate in synthetic maps, but real-world applications with partial observability and dynamic obstacles would likely reduce this metric further.

Conclusion and Future Work

Conclusion

This paper proposes a path planning method for fire detection in educational experimental zones based on an improved Deep Q-Learning (DQN) algorithm.Aimed at addressing the UAV path planning requirements in complex environments, the method enhances DQN to enable efficient and safe path generation.Experimental results demonstrate that the proposed method outperforms baseline approaches across multiple key metrics, offering reliable technical support for fire detection tasks.

Specifically, this study overcomes the limitations of the original DQN in path planning through a series of targeted improvements.First, the adoption of a dynamic -greedy strategy and extended exploration phase enhances early-stage exploration, ensuring better adaptability to complex environments.Second, the introduction of dynamic risk weighting and reinforced penalty in high-risk areas significantly reduces the UAV's probability of traversing hazardous zones, lowering the average number of steps in high-risk areas from 7.8 to 2.1.Furthermore, the use of diagonal action incentives and A* heuristic constraints optimizes path length, reducing the average path length from 38.5 steps (original DQN) to 27.2 steps, approaching the theoretical optimum of 26 steps.Finally, the reduction of learning rate and the imposition of a hard path length constraint improve training stability, with cumulative rewards stabilized between 550 and 650, and the success rate increased to 92.5%, representing an improvement of approximately 34.6% over the original DQN.

Compared with the traditional A* algorithm, the improved DQN demonstrates stronger adaptability in dynamic environments, achieving shorter path lengths and fewer steps through high-risk areas, thereby exhibiting superior overall performance.These results indicate that the improved DQN method effectively balances path efficiency and safety, providing an efficient and reliable path planning solution for UAV deployment in fire detection tasks within educational experimental zones.Moreover, by integrating traditional heuristic algorithms with deep reinforcement learning, this study contributes to the theoretical advancement of DQN applications in the field of path planning.

Future Work

Although the improved DQN method has demonstrated excellent performance in simulation experiments, there remain several aspects that warrant further investigation and optimization to enhance its practical application value.

First, the current experiments are conducted on a static grid map. In the future, dynamic obstacles can be introduced to evaluate the algorithm's real-time adaptability. For instance, dynamic environmental models combined with online learning strategies could be designed to enable UAVs to adjust their paths in real-time during fire scenarios. Future research will address several open challenges to improve the practicality of the proposed method. First, using a Partially Observable Markov Decision Process (POMDP) for dynamic obstacle modeling will enhance the algorithm's adaptability to real-time fire spread. Second, computational complexity remains a potential bottleneck of the improved DQN. The relatively long training time (approximately 2 hours) may limit its deployment on resource-constrained devices. Future research may explore techniques such as network pruning or distributed training to improve computational efficiency and reduce training time. Additionally, cooperative path planning for multiple UAVs is another important research direction. Fire detection tasks in educational experimental zones may require multiple UAVs to operate simultaneously. Future studies could incorporate multi-agent reinforcement learning (e.g., MADDPG) to enable inter-agent collaboration and task allocation.

Moreover, exploring quaternion-valued neural networks[24] could enhance the representation of states in complex spatial data (such as 3D environments or color thermal imaging), potentially improving convergence and robustness. The parameter sensitivity of the algorithm—such as learning rate and risk weight—requires further investigation. Currently, the parameters are manually tuned. Future work could introduce adaptive parameter adjustment methods (e.g., Bayesian optimization) to improve the generalization capability of the algorithm. Open questions include how to balance real-time performance with model complexity, and how to ensure the algorithm's generalizability across different educational zone layouts. Finally, hardware

integration and environmental perception should be considered for real-world applications. In future studies, the improved DQN algorithm could be deployed on real UAV platforms and integrated with sensor data (e.g., thermal imaging cameras and smoke detectors) for field testing, further validating its practicality in fire detection tasks within educational experimental zones.

In summary, the improved DQN method proposed in this study provides a solid foundation for path planning in fire detection within educational experimental zones, demonstrating significant theoretical value and application potential. Future research will focus on dynamic environment adaptability, computational efficiency, and multi-drone collaboration, while also exploring new technologies such as quaternion-valued neural networks to promote the widespread application of this technology in fire emergency response.

References

1. Chou, J.-S., Cheng, M.-Y., Hsieh, Y.-M., Yang, I.-T. & Hsu, H.-T. Optimal path planning in real time for dynamic building fire rescue operations using wireless sensors and visual guidance. *Autom. Constr.* **99**, 1–17, DOI: 10.1016/j.autcon.2018.11.020 (2019).

2. Hart, P. E., Nilsson, N. J. & Raphael, B. A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. Syst. Sci. Cybern.* **4**, 100–107, DOI: https://doi.org/10.1109/TSSC.1968.300136 (1968).

3. Dijkstra, E. W. A note on two problems in connexion with graphs. In *Edsger Wybe Dijkstra: his life, work, and legacy*, 287–290, DOI: https://doi.org/10.1145/3544585.3544600 (2022).

4. Tai, L., Paolo, G. & Liu, M. Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation. In *Proc. 2017 IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, 31–36, DOI: https://doi.org/10.1109/IROS.2017.8202134 (2017).

5. Chen, Y., Du, J., Mumtaz, J., Zhong, J. & Rauf, M. An efficient q-learning integrated multi-objective hyper-heuristic approach for hybrid flow shop scheduling problems with lot streaming. *Expert. Syst. with Appl.* **262**, 125616, DOI: https://doi.org/10.1016/j.eswa.2024.125616 (2025).

6. Sampath, P., Vijayalakshmi, K., Kala Rathi, M. & Mathavan, N. Efficient indoor localization by integrating rfid with adaptive emperor penguin colony-based deep q learning. *J. Chin. Inst. Eng.* 1–12, DOI: https://doi.org/10.1080/02533839.2025.2458073 (2025).

7. Zhu, Y. *et al.* Research on escape route planning analysis in forest fire scenes based on the improved a* algorithm. *Ecol. Indic.* **166**, 112355, DOI: https://doi.org/10.1016/j.ecolind.2024.112355 (2024).

8. Yang, H., Hu, C., Li, G. & Fan, J. A fire escape simulation system based on the dijkstra algorithm. *Comput. Syst. Sci. Eng.* **39**, 365–372, DOI: https://doi.org/10.32604/CSSE.2021.016377 (2021).

9. Zhang, Z. *et al.* Evolving chimp optimization algorithm using quantum mechanism for engineering applications: a case study on fire detection. *J. Comput. Des. Eng.* **11**, 143–163, DOI: https://doi.org/10.1093/jcde/qwae074 (2024).

10. Feng, Z. *et al.* Emergency fire escape path planning model based on improved ddpg algorithm. *J. Build. Eng.* **95**, 110090, DOI: https://doi.org/10.1016/j.jobe.2024.110090 (2024).

11. Li, X., Chen, N., Ma, H., Nie, F. & Wang, X. A parallel genetic algorithm with variable neighborhood search for the vehicle routing problem in forest fire-fighting. *IEEE Trans. Intell. Transp. Syst.* DOI: https://doi.org/10.1109/TITS.2024.3395930 (2024).

12. Xu, S. *et al.* Indoor emergency path planning based on the q-learning optimization algorithm. *ISPRS Int. J. Geo-Information* **11**, 66, DOI: https://doi.org/10.3390/ijgi11010066 (2022).

13. Diaz-Vilor, C., Lozano, A. & Jafarkhani, H. A reinforcement learning approach for wildfire tracking with uav swarms. *IEEE Trans. Wirel. Commun.* DOI: https://doi.org/10.1109/TWC.2024.3524324 (2025).

14. Budakova, D., Vasilev, V. & Dakovski, L. A reinforcement learning algorithm for the optimal evacuation route finding from an electrical substation. In *Proc. AIP Conf. Proceedings*, vol. 3078, DOI: https://doi.org/10.1063/5.0209018 (2024).

15. Mnih, V. *et al.* Human-level control through deep reinforcement learning. *Nature* **518**, 529–533, DOI: https://doi.org/10.1038/nature14236 (2015).

16. Zhou, W., Zhang, C. & Chen, S. Dual deep q-learning network guiding a multiagent path planning approach for virtual fire emergency scenarios. *Appl. Intell.* **53**, 21858–21874, DOI: https://doi.org/10.1007/s10489-023-04601-9 (2023).

17. Bhattarai, M. & Martinez-Ramon, M. A deep q-learning based path planning and navigation system for firefighting environments. *arXiv preprint arXiv:2011.06450* DOI: https://doi.org/10.48550/arXiv.2011.06450 (2020).

18. Demir, K., Tumen, V., Kosunalp, S. & Iliev, T. A deep reinforcement learning algorithm for trajectory planning of swarm uav fulfilling wildfire reconnaissance. *Electronics* **13**, 2568, DOI: https://doi.org/10.3390/electronics13132568 (2024).

19. Zhang, K., Zhu, D., Xu, Q., Zhou, H. & Peng, X. Apf-dqn: Adaptive objective pathfinding via improved deep reinforcement learning among building fire hazard. In *Proc. Int. Conf. Artificial Neural Networks*, 265–279, DOI: https://doi.org/10.1007/978-3-031-72356-8-18 (2024).

20. Min, K., Kim, H. & Huh, K. Deep q learning based high level driving policy determination. In *Proc. 2018 IEEE Intelligent Vehicles Symposium (IV)*, 226–231, DOI: https://doi.org/10.1109/IVS.2018.8500645 (2018).

21. Kosuru, V. S. R. & Venkitaraman, A. K. Developing a deep q-learning and neural network framework for trajectory planning. *Eur. J. Eng. Technol. Res.* **7**, 148–157, DOI: https://doi.org/10.24018/ejeng.2022.7.6.2944 (2022).

22. Murray, L. *et al.* Advancing forest fire prevention: Deep reinforcement learning for effective firebreak placement. *arXiv preprint arXiv:2404.08523* DOI: https://doi.org/10.48550/arXiv.2404.08523 (2024).

23. Liu, W., Lyu, S.-K., Liu, T., Wu, Y.-T. & Qin, Z. Multi-target optimization strategy for unmanned aerial vehicle formation in forest fire monitoring based on deep q-network algorithm. *Drones* **8**, 201, DOI: https://doi.org/10.3390/drones8050201 (2024).

24. Manoj, M., Sriraman, R., Gurusamy, R. & et al. Design and analysis of quaternion-valued neural networks for storing and retrieving color images. *J. Appl. Math. Comput.* DOI: https://doi.org/10.1007/s12190-025-02485-1 (2025).

Data availability

The data utilized in this paper is obtained through self-gathering and is made publicly available to make the study reproducible. If you want to request the complete dataset and code, please email the corresponding author.

Author contributions statement

All the authors contributed extensively to the manuscript. L.F. and Y.L. designed the study and wrote the main manuscript. L.Z. developed the software and performed the validation. Y.L. supervised the project and handled data curation. L.F. revised and edited the paper and managed project administration and funding. All authors have read and agreed to the published version of the manuscript.

Funding

The present study receives support from the Shanxi Provincial Basic Research Project (Grant No. 202303021222058).

Additional information

The authors declare no competing interests.