

## Article

# A Hierarchical Path Planning Framework of Plant Protection UAV Based on the Improved D3QN Algorithm and Remote Sensing Image

Haitao Fu <sup>1</sup>, Zheng Li <sup>1</sup>, Jian Lu <sup>2</sup>, Weijian Zhang <sup>1</sup>, Yuxuan Feng <sup>1</sup>, Li Zhu <sup>1</sup>, He Liu <sup>3</sup> and Jian Li <sup>1,\*</sup>

<sup>1</sup> College of Information Technology, Jilin Agricultural University, Changchun 130118, China; fht@jlau.edu.cn (H.F.); 20231308@mails.jlau.edu.cn (Z.L.); 20231257@mails.jlau.edu.cn (W.Z.); fengyuxuan@jlau.edu.cn (Y.F.); zhuli@jlau.edu.cn (L.Z.)

<sup>2</sup> College of Resources and Environment, Jilin Agricultural University, Changchun 130118, China; lujian025@mails.jlau.edu.cn

<sup>3</sup> College of Engineering and Technology, Jilin Agricultural University, Changchun 130118, China; liuhe@jlau.edu.cn

\* Correspondence: lijian@jlau.edu.cn; Tel.: +86-139-4419-5488

## Abstract

Traditional path planning algorithms often fail to simultaneously ensure operational efficiency, energy constraint compliance, and environmental adaptability in agricultural scenarios, thereby hindering the advancement of precision agriculture. To address these challenges, this study proposes a deep reinforcement learning algorithm, MoE-D3QN, which integrates a Mixture-of-Experts mechanism with a Bi-directional Long Short-Term Memory model. This design enhances the efficiency and robustness of UAV path planning in agricultural environments. Building upon this algorithm, a hierarchical coverage path planning framework is developed. Multi-level task maps are constructed using crop information extracted from Sentinel-2 remote sensing imagery. Additionally, a dynamic energy consumption model and a progressive composite reward function are incorporated to further optimize UAV path planning in complex farmland conditions. Simulation experiments reveal that in the two-level scenario, the MoE-D3QN algorithm achieves a coverage efficiency of 0.8378, representing an improvement of 37.84–63.38% over traditional algorithms and 19.19–63.38% over conventional reinforcement learning methods. The redundancy rate is reduced to 3.23%, which is 38.71–41.94% lower than traditional methods and 4.46–42.77% lower than reinforcement learning counterparts. In the three-level scenario, MoE-D3QN achieves a coverage efficiency of 0.8261, exceeding traditional algorithms by 52.13–71.45% and reinforcement learning approaches by 10.15–50.2%. The redundancy rate is further reduced to 5.26%, which is significantly lower than the 57.89–92.11% observed with traditional methods and the 15.57–18.98% reported for reinforcement learning algorithms. These findings demonstrate that the MoE-D3QN algorithm exhibits high-quality planning performance in complex farmland environments, indicating its strong potential for widespread application in precision agriculture.

**Keywords:** precision agriculture; deep reinforcement learning; path planning; Mixture-of-Experts mechanism; remote sensing



Academic Editors: Dongyu Li, Rui Li, Yinxia Cao and Haoyang Yang

Received: 8 June 2025

Revised: 24 July 2025

Accepted: 1 August 2025

Published: 4 August 2025

**Citation:** Fu, H.; Li, Z.; Lu, J.; Zhang, W.; Feng, Y.; Zhu, L.; Liu, H.; Li, J. A Hierarchical Path Planning Framework of Plant Protection UAV Based on the Improved D3QN Algorithm and Remote Sensing Image. *Remote Sens.* **2025**, *17*, 2704. <https://doi.org/10.3390/rs17152704>

**Copyright:** © 2025 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

With the rapid advancement of unmanned aerial vehicle (UAV) technology, the paradigms and technical pathways of modern agricultural production have been pro-

foundly restructured [1,2]. Plant protection UAVs, characterized by exceptional maneuverability and a broad aerial perspective, have overcome operational bottlenecks associated with complex terrains—including plains, hills, and mountainous regions—and have significantly enhanced the flexibility and efficiency of crop protection operations. By alleviating rural labor shortages and mitigating pesticide overuse, the large-scale deployment of UAV platforms also signifies an evolution in the precision-agriculture technology framework, opening new avenues for the efficient utilization of agricultural resources and the sustainable management of ecosystems. As a result, the construction and comprehensive development of integrated rural systems have been powerfully advanced [3–6].

In UAV-driven precision-agriculture systems, the formulation of an effective path planning strategy is paramount to harmonizing the “perception–decision–action” pipeline. Fundamentally, this challenge entails not only achieving comprehensive spatial coverage but also deeply integrating the time-varying characteristics of the agricultural environment with the dynamic requirements of field tasks [7,8]. To date, two primary categories of planning methods have been employed. The first comprises classical approaches—such as heuristic-based A\* search [9], population-inspired genetic and particle-swarm algorithms [10], and sampling-based rapidly exploring random trees [11]—which generally excel in simple, well-defined scenarios but suffer from high computational complexity and limited adaptability when environmental conditions, targets, or obstacles change, necessitating full replanning. The second category encompasses machine-learning techniques, notably Deep Q-Networks (DQN) [12] leveraging experience replay and policy gradient methods such as Proximal Policy Optimization (PPO) [13]. In the context of agriculture—where environments are complex and constraints are numerous—traditional algorithms often struggle to accommodate all operational factors, whereas deep reinforcement-learning (DRL) algorithms, by virtue of their capacity for autonomous representation of high-dimensional state spaces and end-to-end decision optimization, are emerging as a promising avenue for overcoming the limitations of conventional path planning solutions.

In recent years, a variety of advanced path planning approaches have been developed for coverage tasks in agricultural and orchard environments. Zhang et al. proposed a method based on an improved A\* algorithm for orchard-weeding robots [14], in which a weighted heuristic cost function and an additional turning-angle penalty were introduced to optimize coverage-path planning. Wu et al. developed an enhanced genetic algorithm incorporating simulated-annealing-inspired stretch-fitness evaluation tailored for complete coverage path planning [15], the hyperbolic-tangent activation function from neural networks, and fireworks-algorithm operators to mitigate early-stage premature convergence and bolster global search capabilities. Kong et al. presented B-APFDQN [16], a UAV-based coverage-path planning algorithm that fuses DQN learning with an artificial potential field (APF). In this approach, APF-derived priors are used to curtail blind exploration during training, and a novel network architecture simultaneously outputs both Q-values and action-distribution probabilities, thereby accelerating convergence. Zhou et al. introduced a robot-coverage strategy grounded in DRL [17], employing a DDQN with prioritized experience replay (PER) and incorporating a blockage-and-blind-area (BBA) mechanism to guide exploration toward unvisited regions. Lin et al. designed a collision-free coverage method for guava-picking robots by combining a recurrent neural network (RNN) with an enhanced deep deterministic policy gradient (DDPG) algorithm to predict safe trajectories in unknown states [18]. Finally, Ahmad et al. proposed a DRL-based coverage scheme for precision crop-health monitoring in smart agriculture, in which a centralized convolutional neural network (CNN) is integrated with DDQN. A bespoke reward function was devised to incentivize visits to high-value, uninspected areas while penalizing suboptimal actions [19].

Despite the considerable progress made in UAV path planning research, existing approaches still exhibit limited applicability in agricultural scenarios. Many current methods tend to oversimplify farmland as homogeneous areas and focus on global traversal optimization, which fails to meet the integrated demands of precision agriculture in terms of operational accuracy, resource efficiency, and ecological sustainability. On the one hand, prevailing strategies often overlook critical factors such as variations in cropping patterns, spatial heterogeneity of pest and disease distribution, and dynamic changes in UAV payloads. This omission restricts both the precision of pesticide application and the overall energy efficiency. On the other hand, the singular pursuit of globally optimal paths may result in issues such as redundant pesticide application, energy waste, and contamination of ecologically sensitive zones, thereby making it difficult to balance increased agricultural output with environmental sustainability.

To address these challenges, Mukhamediev et al. proposed a genetic algorithm-based heterogeneous UAV swarm coverage planning method [7], which targets irregular farmland operations by jointly optimizing UAV selection, flight route design, and ground platform scheduling, significantly reducing overall system costs. Syed et al. focused on semi-structured orchard environments and developed an improved YOLOv8-CNN model integrating Ghost modules and a SE attention mechanism [20], enabling efficient classification of real versus false obstacles and enhancing the safety and decision-making efficiency of autonomous navigation. In another study, Li et al. addressed multi-region fertilization tasks in large and complex farmland by proposing a path planning method based on an enhanced Double Deep Q-Learning algorithm [21]. By utilizing GEE-GNDVI to extract nitrogen stress distribution and construct multi-region task maps, the approach effectively enabled intelligent optimization of fertilization paths. These studies underscore the inherent complexity and specificity of agricultural environments and offer valuable insights for improving UAV path planning strategies tailored to the needs of precision agriculture.

To address these limitations, a hierarchical coverage path planning (H-CPP) framework is proposed in this study, whereby adaptability to agricultural environments is enhanced via a two-tier optimization structure. Using satellite-derived organic-matter indices, croplands are first subdivided into graded subzones; high-risk areas are then prioritized for scheduling. Concurrently, dynamic energy consumption constraints and a hybrid-expert architecture are integrated to further improve coverage efficiency.

The principal contributions of this study are summarized as follows:

1. To address the limited policy generalization capability of the D3QN algorithm caused by insufficient representational power in complex and dynamic environments, an improved deep reinforcement learning algorithm, termed MoE-D3QN, is proposed in this study. The algorithm integrates the expert mechanism of a Mixture-of-Experts (MoE) structure with a Bi-directional Long Short-Term Memory (Bi-LSTM) module to enhance the model's ability to capture temporal features, improve state perception accuracy, and enable efficient policy generation under complex state conditions.
2. To better simulate real-world agricultural scenarios and enhance the applicability of the proposed algorithm, a hierarchical coverage path planning (H-CPP) framework is developed in this study. This framework utilizes Sentinel-2 satellite data to extract spectral bands for the calculation of the CCC index, which is employed to detect chlorophyll content in crops. A hierarchical approach is then applied to segment multi-level task regions, while a grid-based method is used to simulate the agricultural UAV mapping environment.
3. This study designs a progressive composite reward function that synergistically combines hierarchical reward guidance—including priority coverage reward, target distance guidance reward, and energy efficiency reward—with dynamic penalty mechanisms, such as invalid exploration penalty and safety violation penalty. This approach

aims to balance coverage efficiency and energy consumption while ensuring adherence to priority constraints.

## 2. Materials and Methods

### 2.1. Designed Planning Area Description

Canopy chlorophyll content serves as a pivotal indicator of crop photosynthetic physiology and is thus of paramount importance for modern agricultural monitoring. As the principal pigment driving photosynthesis, chlorophyll concentration directly governs the efficiency of material and energy exchange between the plant and its environment, while simultaneously reflecting the crop's photosynthetic capacity, nutritional status, and growth trajectory [22,23]. By leveraging spectral remote-sensing techniques to obtain three-dimensional distributions of canopy chlorophyll, quantitative linkages can be established between crop phenotypic traits and underlying ecological processes. Such data furnish essential support for real-time growth diagnostics, fertility management, and yield forecasting in precision-agriculture systems, representing a critical technological enabler for scientific crop monitoring and management.

The study area selected for this investigation is Changchun, the capital city of Jilin Province, China, geographically bounded by latitudes 43°05′ to 45°15′N and longitudes 124°18′ to 127°05′E. Situated at the core of the Songliao Plain within the Northeast China Plain, this region is underlain by the characteristic “black soil” belt. The predominant soil types—highly fertile Chernozem and meadow soils—exhibit elevated surface organic-matter content, classifying the area as a globally scarce high-yield arable land resource [24]. The extensive, flat terrain further provides an ideal physical foundation for the implementation of precision-agriculture technologies [25].

This study was implemented on the Google Earth Engine platform, employing the LEAF-Toolbox inversion routines developed by Sentinel-2 Level-2A surface reflectance data were integrated with SRTM terrain elevations to drive a PROSAIL radiative-transfer-model-based retrieval workflow [26–28]. First, the S2MaskClear algorithm was used to exclude pixels exhibiting over 10% cloud cover, and atmospheric correction was applied via Sen2Cor. Thereafter, topographic illumination and radiometric distortions were corrected by merging the SAIL canopy model with SRTM-derived terrain information. Finally, multispectral inversion was performed using the SL2P module—synergizing visible, red-edge, and near-infrared bands—to generate a canopy chlorophyll-content (CCC) estimate for the Changchun region (125.17°E–125.82°E, 43.45°N–44.15°N), as illustrated in Figure 1.

After obtaining the CCC index map, the area was partitioned into multi-level task regions and non-task regions, which were subsequently converted into a grid map format suitable for computational processing, as illustrated in Figure 2. To ensure that the task regions accurately reflect the actual crop distribution, a CCC lower threshold of 100,000 was established. Areas below this threshold were classified as non-task zones and excluded from the path planning process.

Following this, a quantile-based method was employed to categorize task priorities: regions with CCC values exceeding 410,000 were assigned Priority I, representing vigorously growing crop areas; values between 230,000 and 410,000 were designated Priority II; and those between 100,000 and 230,000 were classified as Priority III, corresponding to low-density or potentially stressed crop zones.

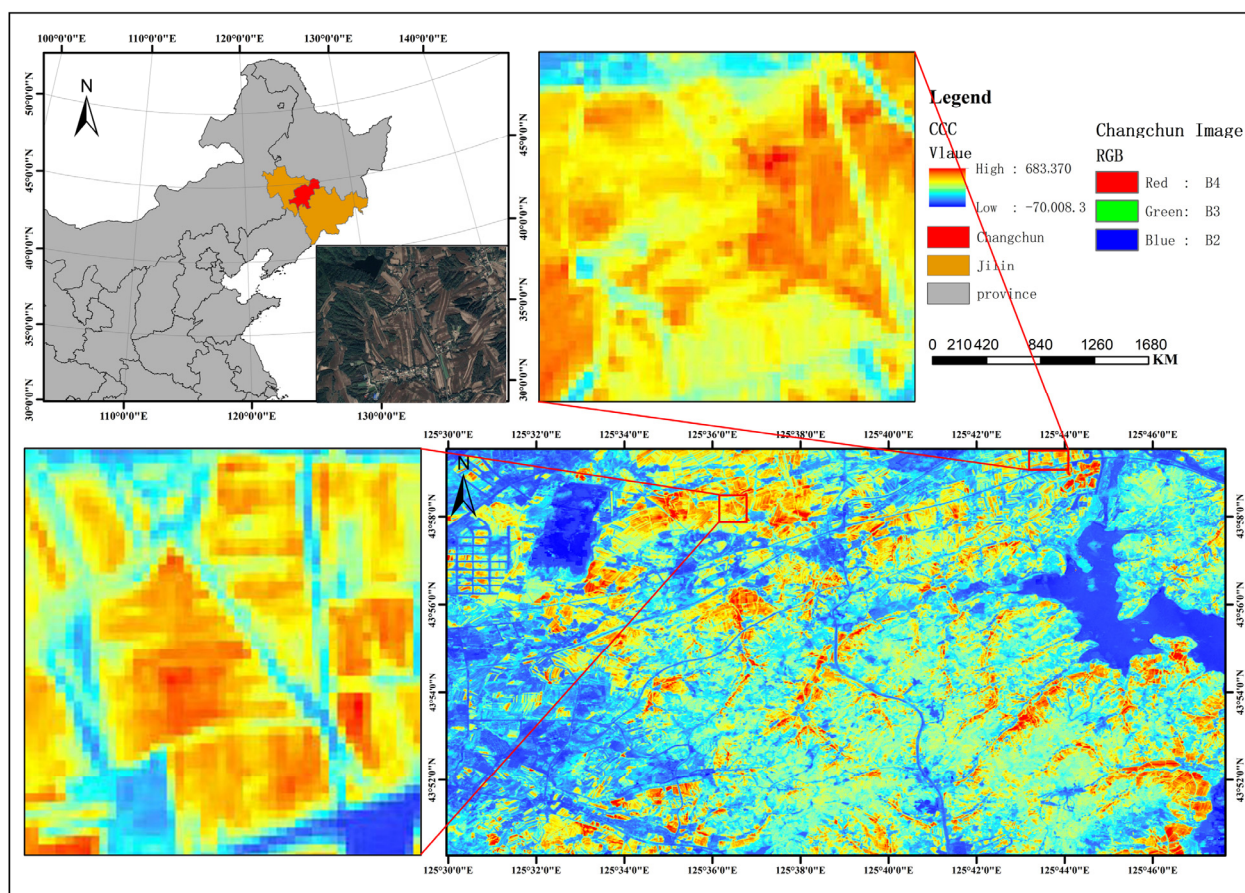


Figure 1. Canopy chlorophyll-content inversion results derived by remote sensing for the study area.

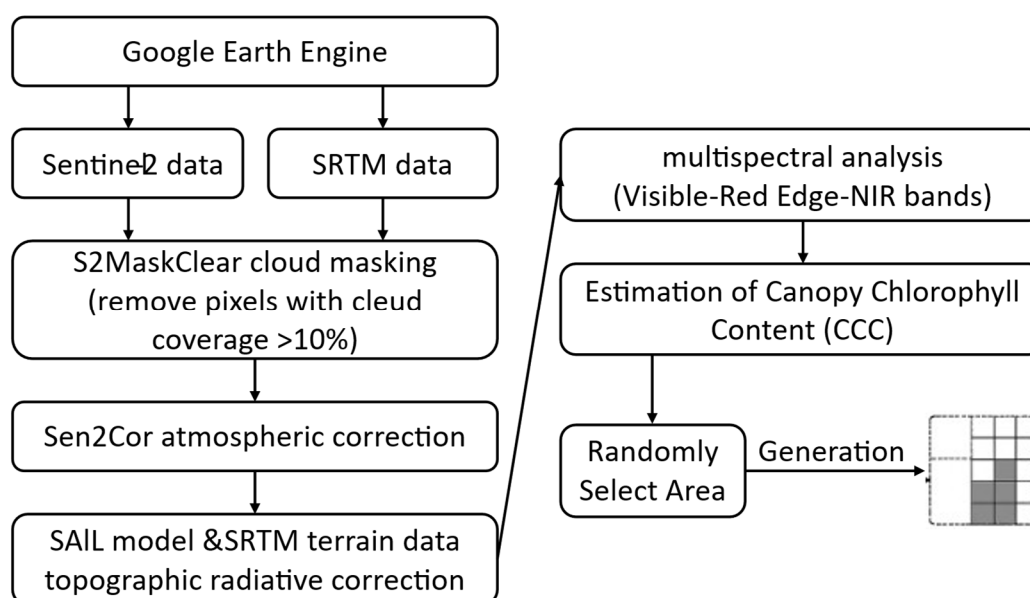


Figure 2. Task map processing flow chart.

This prioritization scheme is grounded in the principle of maximizing operational resource efficiency by focusing coverage on high-yield potential areas. Such prioritization helps consolidate the healthy growth of major production zones and prevents the spread of pests and diseases, thereby enhancing the operational value and responsiveness of the path planning.

## 2.2. State Space

The current temporal characteristics of the mission environment were defined as the state space. Using a grid-based discretization approach, the study area was partitioned into a series of discrete cells. Each cell was assigned a unique identifier corresponding to a specific environmental state—such as task zone, obstacle region, or energy-resupply area—to facilitate subsequent decision-making and path planning processes. The meaning of environmental unit is shown in Table 1.

**Table 1.** Map state corresponding to different values.

| Value ( $Env_{(x,y)}$ ) | Description                    |
|-------------------------|--------------------------------|
| 9                       | Priority Level I area          |
| 8                       | Priority Level II area         |
| 7                       | Priority Level III area        |
| 4                       | Energy consumption supply area |
| 3                       | Obstacle area                  |
| 2                       | Current position of UAV        |
| 1                       | Non-task area                  |
| 0                       | Covered task area              |

## 2.3. Action Space and Action Selection Strategy

The action space was defined as the set of all feasible commands available to the UAV during the path planning process. Although the UAV's motion model theoretically supports a continuous action set capable of omnidirectional pose transformations, the accompanying increase in degrees of freedom has been shown to markedly impede training efficiency. To alleviate the computational burden associated with continuous control dimensions, an action policy comprising eight discrete commands was devised. Upon execution of any given command, the state space is immediately updated and the current state  $S_t$  transitions to the subsequent state  $S_t + 1$ . The detailed state-transition mechanism is illustrated in Table 2.

**Table 2.** Action space.

| Action <sub>(i)</sub> | State Transition Process | Description    |
|-----------------------|--------------------------|----------------|
| 0                     | (x, y + 1)               | North (N)      |
| 1                     | (x, y − 1)               | South (S)      |
| 2                     | (x − 1, y)               | West (W)       |
| 3                     | (x + 1, y)               | East (E)       |
| 4                     | (x − 1, y + 1)           | Northwest (NW) |
| 5                     | (x + 1, y + 1)           | Northeast (NE) |
| 6                     | (x − 1, y − 1)           | Southwest (SW) |
| 7                     | (x + 1, y − 1)           | Southeast (SE) |

The balance between exploration and exploitation is central to the action-selection strategy. To pursue higher rewards, the UAV must attempt various actions during exploration; however, excessive exploration can degrade algorithmic efficiency and slow convergence, whereas excessive exploitation may lead to entrapment in local optima and hinder discovery of a global optimum. To address this issue, an  $\epsilon$ -greedy algorithm was adopted for action selection: with probability  $\epsilon$  the action yielding the maximum estimated return is chosen, and with probability  $1 - \epsilon$  a random action is selected, as shown in Equation (1).

$$Action = \begin{cases} \operatorname{argmax} Q(s, a), & \text{if } RN > \epsilon \\ \text{random action}, & \text{Otherwise} \end{cases} \quad (1)$$

Meanwhile, to avoid the meaningless exploration and slow convergence caused by a fixed  $\varepsilon$  in the traditional  $\varepsilon$ -greedy algorithm, a dynamic  $\varepsilon$  schedule was employed. In this scheme, exploration is initially prioritized, and, as the number of training episodes increases, the exploration probability is gradually reduced while the exploitation weight is correspondingly increased, as shown in the following Equation (2). Here,  $\varepsilon_{min}$  denotes the minimum exploration probability,  $\varepsilon_{max}$  the maximum probability, *Decay rate* the rate at which exploration probability decays, and *currunt episode* the index of the current episode.

$$\varepsilon = \varepsilon_{min} + (\varepsilon_{max} - \varepsilon_{min}) \times e^{-\text{Decay rate} \times \text{currunt episode}} \quad (2)$$

## 2.4. Environmental Constraints

### 2.4.1. Priority Constraints

In hierarchical coverage path planning tasks, priority constraints constitute the core logic of task planning and resource scheduling. A well-designed constraint directly influences operational efficiency. Task types are arranged in descending order of priority, as illustrated in Equation (3), where  $P_1$  represents the highest priority and  $P_n$  the lowest. The symbol “>” denotes a strict priority relationship, indicating that all tasks of type  $P_1$  must be completed before proceeding to those of type  $P_2$ .

$$\mathcal{P} = \{P_1, P_2, \dots, P_n\} (P_1 > P_2 > \dots P_n) \quad (3)$$

At the end of each time step, the remaining number of tasks in each priority region is calculated, as defined in Equation (4).  $N_k^{(t)}$  denotes the number of remaining tasks of priority level  $P_k$  at time step  $t$ . This value is obtained by traversing all positions  $(i,j)$  in the environment and counting the locations that satisfy the corresponding conditions.

$$N_k^{(t)} = \sum_{i,j} [m_{ij}^{(t)} = P_k] (k = 1, 2, \dots, n) \quad (4)$$

As the UAV continues to operate, the state is updated dynamically according to the state transition function defined in Equation (5). Specifically,  $m_{x_t y_t}^{(t)} = P_k$  is used to determine the task type at the current location, identifying whether the agent is positioned at a task point of priority  $P_k$ . The term  $\prod_{i=1}^{k-1} [N_i^{(t)} = 0]$  serves to verify whether all higher-priority tasks have been completed.  $N_k^{(t+1)}$  denotes the number of uncompleted tasks of priority level  $k$  remaining at time step  $t + 1$ .

$$N_k^{(t+1)} = N_k^{(t)} - [m_{x_t y_t}^{(t)} = P_k] \cdot \prod_{i=1}^{k-1} \mathbb{I}[N_i^{(t)} = 0] \quad (5)$$

To enhance the agent’s understanding of task constraints, a nearest-distance function is introduced to guide the UAV toward uncompleted high-priority regions. The agent is encouraged to navigate preferentially toward the nearest area with higher task priority, as

defined in Equation (6). Here,  $\mathcal{T}_k(t)$  denotes the set of coordinates corresponding to the remaining unprocessed tasks of priority level  $k$ .

$$d^*(s_t) = \begin{cases} \min_{(i,j) \in \mathcal{T}_1(t)} \| (i,j) - (x_t, y_t) \|_1, & |\mathcal{T}_1(t)| > 0 \\ \min_{(i,j) \in \mathcal{T}_2(t)} \| (i,j) - (x_t, y_t) \|_1, & |\mathcal{T}_1(t)| = 0 \wedge |\mathcal{T}_2(t)| > 0 \\ \vdots & \\ \min_{(i,j) \in \mathcal{T}_n(t)} \| (i,j) - (x_t, y_t) \|_1, & \bigwedge_{m=1}^{n-1} |\mathcal{T}_m(t)| = 0 \wedge |\mathcal{T}_n(t)| > 0 \\ 0, & otherwise \end{cases} \quad (6)$$

## 2.4.2. Energy Consumption Constraint

During agricultural operations, unmanned aerial vehicles (UAVs) are susceptible to disturbances from natural environmental factors and their own payload. Therefore, path planning for UAVs should not be based solely on minimizing time or distance, while neglecting energy consumption. To address this, a dynamic energy consumption constraint for UAVs is proposed in this study, as shown in Equation (7). This constraint is developed based on a mechanism-guided black-box approach, taking into account both payload variations and environmental influences.

$$C_t = C_{move} + K * C_{load} + e \quad (7)$$

In this context,  $C_t$  denotes the energy consumption incurred by the UAV at time step  $t$ . Specifically,  $C_{move}$  represents the baseline energy consumption during unloaded flight, while  $C_{load}$  accounts for the additional energy required due to payload. The coefficient  $K$  refers to the environmental coupling factor, which integrates external influences such as headwind velocity  $w$ . The term  $e$  denotes Gaussian white noise, capturing unmodeled disturbances including sensor inaccuracies and atmospheric turbulence.

## 2.5. Improved Algorithm

### 2.5.1. Basic Theory

Reinforcement learning primarily functions through real-time interactions between an agent and its environment. At time step  $t$ , the agent perceives the environmental state  $s$  and selects an action  $A$  accordingly. The environment then provides a reward  $R$  based on the selected action, and the agent's state is updated to the next state  $S_{t+1}$ . Through repeated iterations of this process, the agent aims to maximize cumulative rewards.

In 2015, the Deep Q-Network algorithm was proposed [29], which employed deep neural networks to approximate the Q-value function in high-dimensional continuous state spaces. To improve learning stability and efficiency, the algorithm introduced an experience replay mechanism and a target network. However, since a single neural network was used to estimate the Q-values, estimation bias could occur, leading to the systematic overestimation of Q-values for certain actions. As training progresses, this estimation error may accumulate and be further amplified, potentially impairing the optimization of the policy.

The Double Denoising Deep Q-Network (D3QN) algorithm adopts the dual-network mechanism of the Double Deep Q-Network [30], employing two structurally identical but parameter-independent networks: the policy network and the target network. The policy network is used for training and action selection, while after a fixed number of training iterations, its parameters are copied to the target network. The target network

then computes the Q-values to decouple action selection from value evaluation, thereby reducing overestimation bias. The target Q-value is calculated according to Equation (8),

$$y = r + \gamma \cdot Q' \left( s_{t+1}, \underset{a}{\operatorname{argmax}} Q(s_{t+1}, a; \theta); \theta' \right) \quad (8)$$

where  $y$  denotes the target Q-value,  $R$  represents the immediate reward,  $Q'$  refers to the target network,  $\underset{a}{\operatorname{argmax}} Q(s_{t+1}, a; \theta)$  is the optimal action that maximizes the Q-value in state  $s_{t+1}$ , and  $\theta'$  represents the parameters updated with delay.

In addition, to further enhance the network's ability to distinguish between state value and action advantage, D3QN incorporates the Dueling DQN architecture [31]. By introducing the state-value function  $V(s_t)$  and the action-advantage function  $A(s_t, a_t)$ , which represent the average value of state  $s_t$  independent of specific actions and the relative advantage of action  $a_t$  compared to the average of all actions in that state, respectively, the policy evaluation capability of the reinforcement learning algorithm is significantly enhanced. The calculation formula is shown in Equation (9).

$$Q(s_t, a_t) = V(s_t) + A(s_t, a_t) - \frac{1}{|A|} \sum_{a'} A(s_t, a') \quad (9)$$

In practical applications such as precision agriculture, environmental states are neither static nor isolated; rather, they exhibit pronounced temporal continuity and undergo constant changes. Therefore, when making decisions, agents are required not only to perceive the current state but also to incorporate trends in environmental dynamics over time, in order to form more forward-looking and informed judgments.

To address this need, researchers have increasingly incorporated recurrent neural network (RNN) architectures—particularly Long Short-Term Memory (LSTM) networks—owing to their capacity for temporal sequence modeling. Through the use of a gating mechanism comprising forget, input, and output gates, LSTM networks are capable of selectively retaining relevant information while suppressing noise, thereby enhancing the model's ability to capture and interpret long-term dependencies.

Building upon conventional LSTM models, Bi-directional Long Short-Term Memory (Bi-LSTM) networks introduce two parallel LSTM layers that process the input sequence in both forward and backward directions. This bi-directional structure enables the simultaneous capture of past and future contextual information within a sequence. Each LSTM unit dynamically regulates information flow via its gating mechanisms, effectively modeling long-range dependencies while mitigating the vanishing gradient problem. By integrating information from both directions, Bi-LSTM networks substantially improve the model's representational capacity for complex temporal patterns.

### 2.5.2. MOE-D3QN

The Mixture-of-Experts architecture has emerged as a prominent paradigm in deep learning, offering notable advantages in enhancing model capacity and computational efficiency through the dynamic routing and integration of multiple expert sub-models. This architecture adaptively allocates computational resources based on the characteristics of input data and employs a gating network to aggregate the outputs of selected experts.

In recent years, with the rapid development of large-scale models, the MoE framework has demonstrated significant potential in fields such as natural language processing. Representative implementations include Google's GShard [32] and the subsequent Switch Transformer [33], both of which activate only a subset of experts during training. This selective activation mechanism substantially improves training efficiency and reduces

computational cost, thereby laying a solid foundation for the practical deployment of MoE-based models.

To address the dual optimization challenges inherent in the conventional D3QN algorithm—namely, the need for synchronous updates of the policy network and the target network, as well as the joint optimization of the state-value function and action-advantage function, which lead to increased parameter counts and reduced training efficiency—this study proposes an improved scheme based on the Mixture-of-Experts framework combined with a Bi-directional Long Short-Term Memory (Bi-LSTM). The enhanced MoE-D3QN model establishes a multi-expert collaboration mechanism where the gating network dynamically allocates parameters, improving model expressiveness under computational constraints. Additionally, the integration of the Bi-LSTM module strengthens temporal feature extraction by capturing long-range dependencies through bi-directional state propagation.

To address the dual challenges faced by the traditional D3QN algorithm—namely, the need for synchronized updates of the policy and target networks, and the parameter redundancy and training inefficiency arising from the joint optimization of the state-value and action-advantage functions—an improved framework, termed MoE-D3QN, is proposed in this study. By incorporating MoE architecture with a gating network, the model enables dynamic parameter allocation and adaptive feature selection. This design significantly enhances policy representation and training efficiency while maintaining constraints on computational resources.

In addition, to further improve the model's capability for temporal feature modeling, Bi-LSTM module is integrated as an auxiliary component. Leveraging its bi-directional state propagation mechanism, the Bi-LSTM effectively captures long-term dependencies during task execution. In the context of the UAV-based precision spraying task addressed in this study, where path planning depends on the dynamic evolution of environmental states, the integration of Bi-LSTM improves the model's ability to perceive historical changes. This, in turn, reduces redundant path coverage and resource waste, while enhancing the continuity and robustness of the learned policies.

As illustrated in Figure 3, the model architecture comprises three core components:

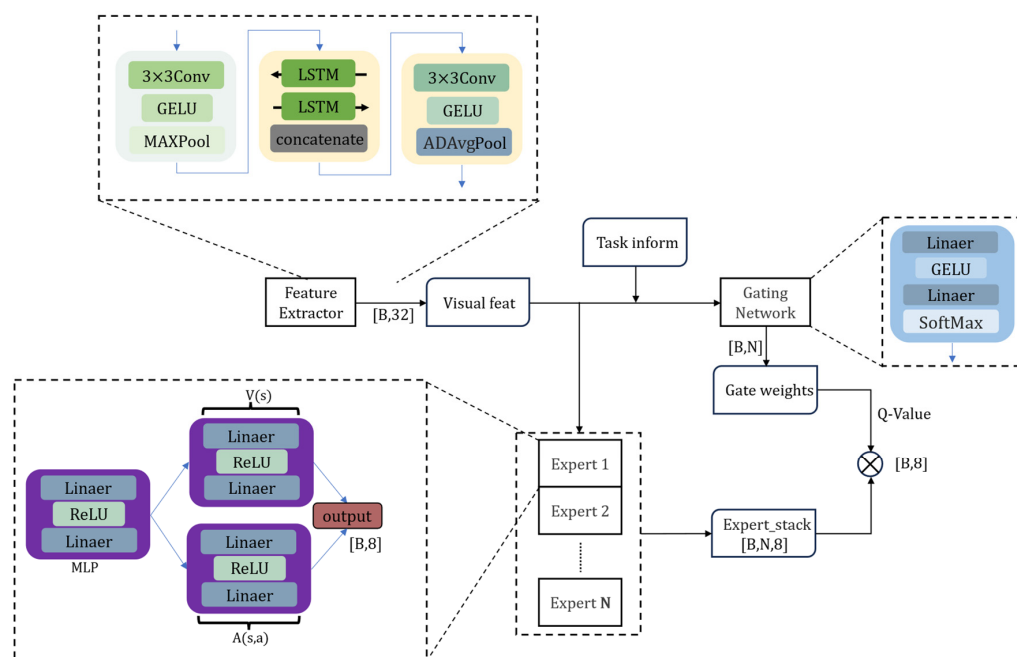


**Figure 3.** Policy and target network model architecture diagram.

(1) A CNN-LSTM cascaded feature extraction module, which adopts depthwise separable convolution and channel attention mechanisms to achieve a lightweight design, with the computation defined in Equation (10).

$$\begin{cases} H, (h_t, c_t) = BiLSTM(Reshape(MaxPool(GELU(W_{conv1} * X + b_{conv1})))) \\ v = Flatten(AdaptiveAvgPool(GELU(W_{conv2} * H + b_{conv2}))) \end{cases} \quad (10)$$

(2) An expert network ensemble, where each expert includes a feature transformation layer as well as independent computation units for the state-value function and the advantage function, as defined in Equation (11).

$$\begin{cases} V_j = W_{jV2} \cdot GELU(W_{jV1} \cdot W_{j2} \cdot GELU(W_{j1} \cdot v + b_{j1}) + b_{jV1}) \\ A_j = W_{jA2} \cdot GELU(W_{jA1} \cdot W_{j2} \cdot GELU(W_{j1} \cdot v + b_{j1}) + b_{jA1}) \\ Q_j = V_j + \left( A_j - \sum_{k=1} A_{j,k} \right) \end{cases} \quad (11)$$

(3) A dynamic gating network, which achieves sparse expert weight distribution through low-rank matrix decomposition and temperature coefficient adjustment. Specifically, the input observation data are first transformed into a high-dimensional representation via the feature extraction module. Based on the complexity of the current state, the gating network generates activation weights for the experts. Each expert network processes the feature vector in parallel, and the outputs are then fused through a weighted aggregation to produce the final action-value function prediction, as defined in Equation (12).

$$\begin{cases} g = Softmax\left(\frac{W_{g2} \cdot GELU(W_{g1} \cdot v + b_{g1}) + b_{g2}}{\tau}\right) \\ Q_{final} = \sum_{j=1}^N g_j \odot Q_j \end{cases} \quad (12)$$

$\tau$  represents the temperature coefficient, which is used to control the smoothness of the Softmax distribution output by the gating network. A linear annealing strategy is adopted to dynamically adjust the temperature coefficient during training, as shown in Equation (13). At the early stages of training, a higher temperature value is assigned to encourage the participation of multiple experts, thereby enhancing the model's responsiveness to complex environmental states encountered in coverage path planning tasks. As training progresses,  $\tau$  is gradually annealed to a lower value, increasing the determinism of expert selection and thereby improving policy stability and convergence speed.

$$\tau(t) = \tau_{max} - \frac{t}{T}(\tau_{max} - \tau_{min}) \quad (13)$$

However, if the temperature coefficient decreases excessively, the Softmax output tends to approximate a one-hot distribution, activating only a single expert. This over-sparsity may hinder the model's generalization ability and lead to inefficient path selection in local regions, ultimately resulting in convergence to suboptimal solutions. To mitigate this issue, a lower bound is imposed on the temperature coefficient, ensuring a balance between determinism and diversity in expert selection. This constraint helps maintain a dynamic equilibrium between exploration and exploitation throughout the training process.

The process of the MOE-D3QN algorithm is shown in Figure 4.

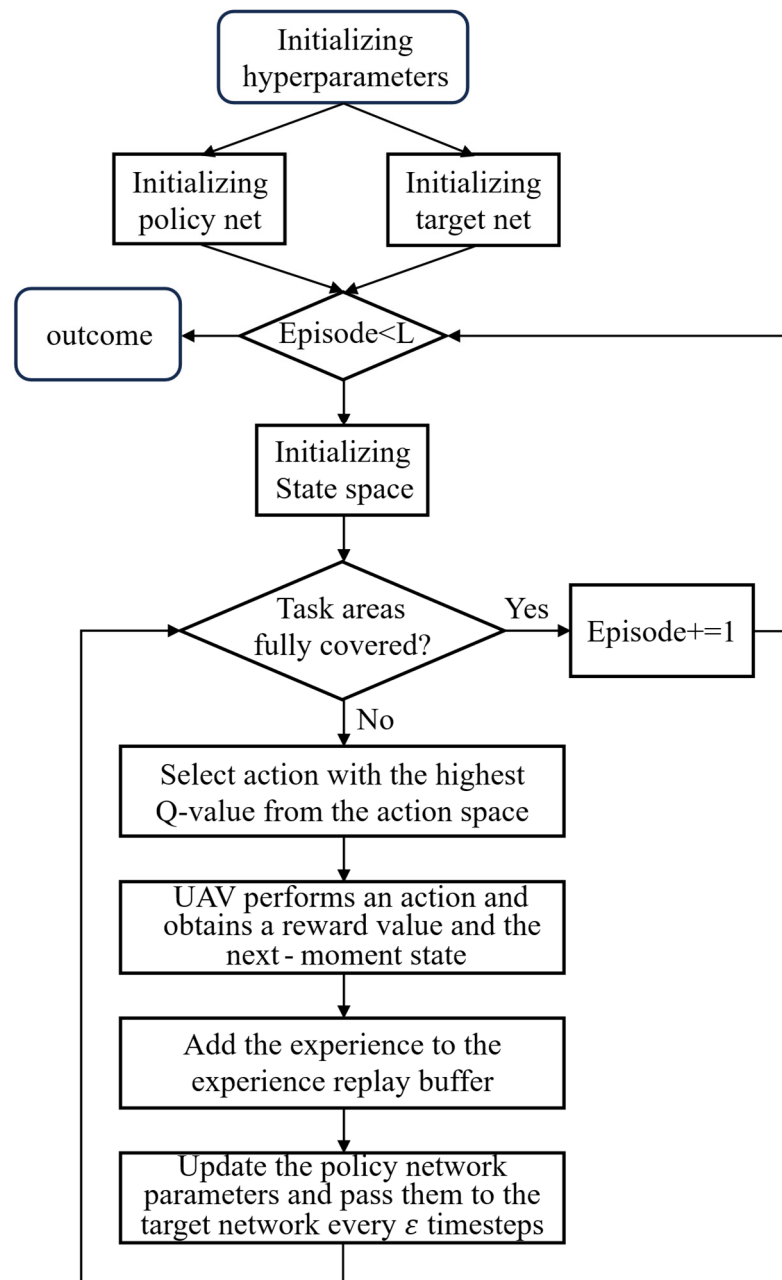


**Figure 4.** Flow chart of the MOE-D3QN algorithm.

### 2.5.3. Reward

To address the challenge faced by traditional path planning algorithms in agricultural environments—namely, the difficulty in simultaneously optimizing task priority, energy efficiency, and safety constraints—this study proposes a reward function designed to guide the UAV. The function encourages the UAV to approach the target coverage area promptly and prioritizes the coverage of high-priority regions in descending order, as defined in Equation (14).

$$r_t = \begin{cases} r_{Nontask}, & \text{Non-task area} \\ r_{overlay}, & \text{Task area coverage} \\ r_{accomplish}, & \text{Complete the coverage task} \end{cases} \quad (14)$$

The term  $r_{Nontask}$  represents the reward assigned to non-task areas, as defined in Equation (15). The variable  $Env_{(UAV \text{ position})}$  denotes the environmental state at the UAV's current location on the map, where 0 indicates a covered task area and 1 denotes a non-

task area. A penalty is applied when the UAV enters an already covered area, while a base penalty is also imposed when the UAV moves into a non-task area. Additionally, a distance-based reward is provided to encourage the UAV to navigate toward task regions, thereby guiding it effectively toward completing mission objectives.

$$r_{Nontask} = \begin{cases} -0.5, & \text{if } Env_{(UAV \text{ position})} = 0 \\ -0.2 + R_{target}, & \text{if } Env_{(UAV \text{ position})} = 1 \end{cases} \quad (15)$$

The term  $r_{overlay}$ , defined as in Equation (16), denotes the coverage reward. A positive reward is assigned whenever the UAV moves into a region that corresponds to the current priority level. In contrast, if the UAV enters a region that does not match the required priority—such as covering a lower-priority area prematurely—a penalty is applied. This reward mechanism is designed to guide the UAV toward prioritizing the coverage of higher-priority areas.

$$r_{overlay} = \begin{cases} 1, & \text{if } Env_{(UAV \text{ position})} = \text{Current task priority} \\ -0.8, & \text{Other} \end{cases} \quad (16)$$

The term  $r_{accomplish}$ , as defined in Equation (17), represents the task completion reward. After each step, a check is performed: if all priority levels in the task sequence have been fully covered, the task is considered complete, and a substantial positive reward is granted. If the sequence is not yet empty, no completion reward is given, and the UAV proceeds to execute the remaining tasks.

$$r_{accomplish} = \begin{cases} 20, & \text{if } \text{Current task priority} = \text{None} \\ 0, & \text{Other} \end{cases} \quad (17)$$

To prevent ineffective exploration, improve training efficiency, avoid local optima or infinite loops, and enhance the safety of the path planning process, this study introduces a task termination constraint, as defined in Equation (18). Let  $S_{episode}$  denote the number of steps taken in the current episode,  $S_{max}$  represent the maximum allowable steps,  $S_{not \text{ reward count}}$  denote the number of consecutive steps without a reward, and  $S_{max \text{ not reward}}$  indicate the maximum allowed consecutive non-rewarded steps. The episode is terminated under any of the following conditions: (1) the total steps taken exceed the maximum allowable steps; (2) the consecutive non-rewarded steps exceed the predefined limit; or (3) the UAV enters an obstacle region.

$$\text{Termination condition} \begin{cases} S_{episode} \geq S_{max} \\ S_{not \text{ reward count}} \geq S_{max \text{ not reward}} \\ Env_{(UAV \text{ position})} = \text{Obstacle area} \end{cases} \quad (18)$$

#### 2.5.4. Sensitivity Analysis of Reward Function Parameters

To further evaluate the influence of key parameters in the reward function and gating mechanism on path planning performance, a sensitivity analysis was conducted from the perspectives of coverage efficiency, redundancy rate, and task completion rate. For each parameter, a univariate perturbation approach was employed: the target parameter was individually adjusted while keeping all other parameters fixed. Specifically, the parameter was varied by  $\pm 30\%$  around its baseline value, and the resulting impact on path quality was systematically observed and analyzed.

As shown in Table 3, under the default configuration (Our), the algorithm achieved superior coverage performance with a low redundancy rate of only 3.23% and a cov-

erage efficiency of 0.8378. In comparative experiments 1a, b, when the non-task area penalty  $r_{Nontask}$  was increased, a reduction in redundant paths was observed. However, due to the absence of guidance from target-directed rewards, the agent struggled to understand the priority coverage relationships. This resulted in an increased number of steps taken in non-task areas and a consequent decline in overall efficiency. Furthermore, as the penalty was reduced, the base movement cost decreased, which inadvertently caused the agent to receive positive rewards for covering non-task regions, leading to significant ineffective coverage in these areas.

**Table 3.** Parameter sensitivity analysis table.

| Serial Number | $r_{Nontask}$ | $r_{overlay}$ | $r_{accomplish}$ | $\tau_{max} \setminus \tau_{min}$ | Repeat (%) | Coverage Efficiency |
|---------------|---------------|---------------|------------------|-----------------------------------|------------|---------------------|
| Our           | −0.5/−0.2     | 1/−0.8        | 20               | 1.0/0.7                           | 3.23%      | 0.8378              |
| 1a            | −0.7/−0.3     | 1/−0.8        | 20               | 1.0/0.7                           | 7.14%      | 0.7                 |
| 1b            | −0.3/−0.1     | 1/−0.8        | 20               | 1.0/0.7                           | 15.38%     | 0.5778              |
| 2a            | −0.5/−0.2     | 1.3/−1        | 20               | 1.0/0.7                           | 3.33%      | 0.8108              |
| 2b            | −0.5/−0.2     | 0.7/−0.5      | 20               | 1.0/0.7                           | 10.71%     | 0.6829              |
| 3a            | −0.5/−0.2     | 1/−0.8        | 26               | 1.0/0.7                           | 6.45%      | 0.7949              |
| 3b            | −0.5/−0.2     | 1/−0.8        | 14               | 1.0/0.7                           | 3.33%      | 0.75                |

In experiments 2a,b, adjustments were made to the priority area reward  $r_{overlay}$ . Increasing this parameter did not yield substantial improvements, indicating that priority relationships were already reasonably well handled. Conversely, decreasing the reward, as demonstrated in 2b, noticeably degraded the prioritization of coverage paths, resulting in increased redundancy and reduced efficiency. Additionally, experiments 3a,b tested the impact of the full coverage task reward. It was found that setting this reward too low or too high disrupted the original planning rhythm, slightly decreasing path efficiency, although the overall effect remained limited.

### 3. Results and Discussion

#### 3.1. Experimental Setup

All simulation experiments in this study were conducted on a high-performance desktop workstation configured with dual NVIDIA GeForce RTX 3090 GPUs (24 GB VRAM each) operating in parallel under the Ubuntu OS. The algorithms were developed and implemented using the Python 3.11.5 programming environment. The experimental parameters are configured as shown in Table 4.

**Table 4.** Experimental parameters table.

| Parameter        | Value              | Description                       |
|------------------|--------------------|-----------------------------------|
| $E_{max}$        | 60,000             | Maximum episode                   |
| $\gamma$         | 0.95               | Discount rate                     |
| $B$              | 128                | Batch size                        |
| $M$              | 60,000             | Experience replay buffer capacity |
| $LR$             | $1 \times 10^{-4}$ | Learning rate                     |
| $\epsilon_{max}$ | 0.9                | Initial exploration rate          |

**Table 4.** *Cont.*

|                  |      |                                                             |
|------------------|------|-------------------------------------------------------------|
| $\epsilon_{min}$ | 0.1  | Minimum exploration rate                                    |
| $\tau_{max}$     | 1    | Maximum temperature coefficient                             |
| $\tau_{min}$     | 0.7  | Minimum temperature coefficient                             |
| Decay rate       | 5000 | Controlling the speed at which $\epsilon$ epsilon decreases |
| $n$              | 10   | Network update frequency                                    |
| $N_{actions}$    | 8    | Action space size                                           |
| Optimizer        | Adam | Optimizer                                                   |

### 3.2. Experimental Results

To systematically validate the performance advantages of the proposed algorithm in multi-level agricultural task scenarios, representative traditional path planning algorithms, namely A-Star and the genetic algorithm (GA), as well as mainstream deep reinforcement learning models including DQN, DDQN, D3QN, and Dueling DQN, were selected as baseline references for comparative experiments. Considering the spatial complexity characteristics of agricultural operational environments, this study constructed representative two-level and three-level hierarchical task regions based on satellite remote sensing imagery data. After 60,000 training iterations, the algorithms' performances were quantitatively evaluated in terms of coverage repetition rate, coverage rate, and coverage efficiency.

In this study, dark blue was assigned to obstacle areas, while task region priorities were represented using progressively lighter shades of blue. White areas indicated non-task regions. Charging stations were marked with red stars, UAV starting points with green circles, and UAV task termination points with pink circles. Yellow lines depicted the initial outbound paths, green lines represented routes taken after returning to charging stations for battery replenishment, and red lines indicated repeated paths.

As shown in Figures 5 and 6, the MOE-D3QN algorithm demonstrated superior coverage efficiency in both two-level and three-level hierarchical task maps, successfully completing the coverage tasks without any confusion in task hierarchy. However, from a global perspective, some route redundancies were observed, with partial path repetitions occurring. Among the comparison algorithms, only the traditional A-Star and genetic algorithms (GA) completed the coverage tasks. The A-Star algorithm, relying on heuristic functions to prioritize exploration in the “optimal” direction, encountered challenges in coverage tasks where the globally optimal path may require temporary deviations from the target direction to satisfy energy consumption and priority constraints. This caused repeated attempts along the same paths. The GA algorithm, when faced with complex constraints, tended to have its population dominated by a few individuals meeting basic constraints but exhibiting similar structures. Subsequent crossover and mutation operations could only slightly adjust path sequences, preventing escape from local optima and resulting in numerous redundant paths differing only in minor details. Furthermore, algorithms such as DDQN and Dueling DQN exhibited limitations due to inherent algorithmic shortcomings, which led to difficulties in convergence during training and entrapment in local optima.

To further evaluate the performance of the algorithms, quantitative analyses were conducted by calculating the coverage repetition rate, coverage rate, and coverage efficiency. A detailed comparison of the results is presented in Table 5.

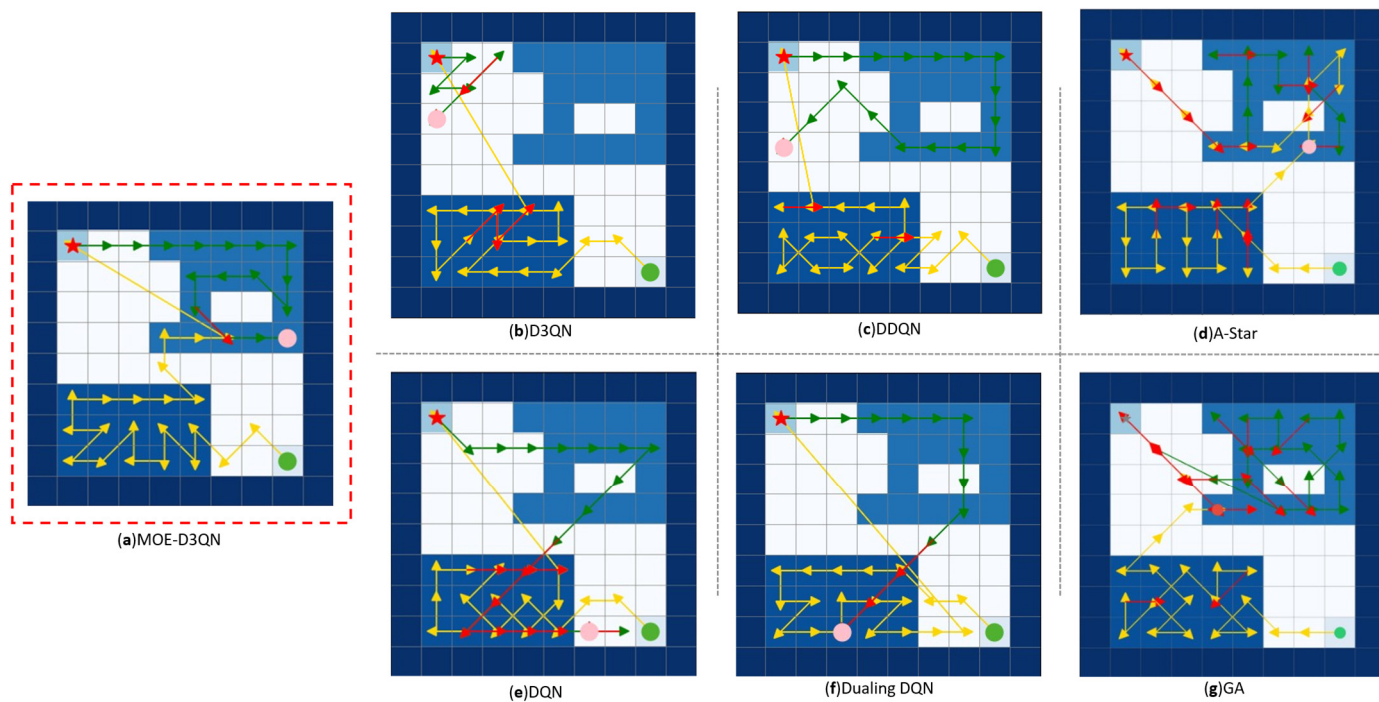


Figure 5. Two-level task planning result diagram.

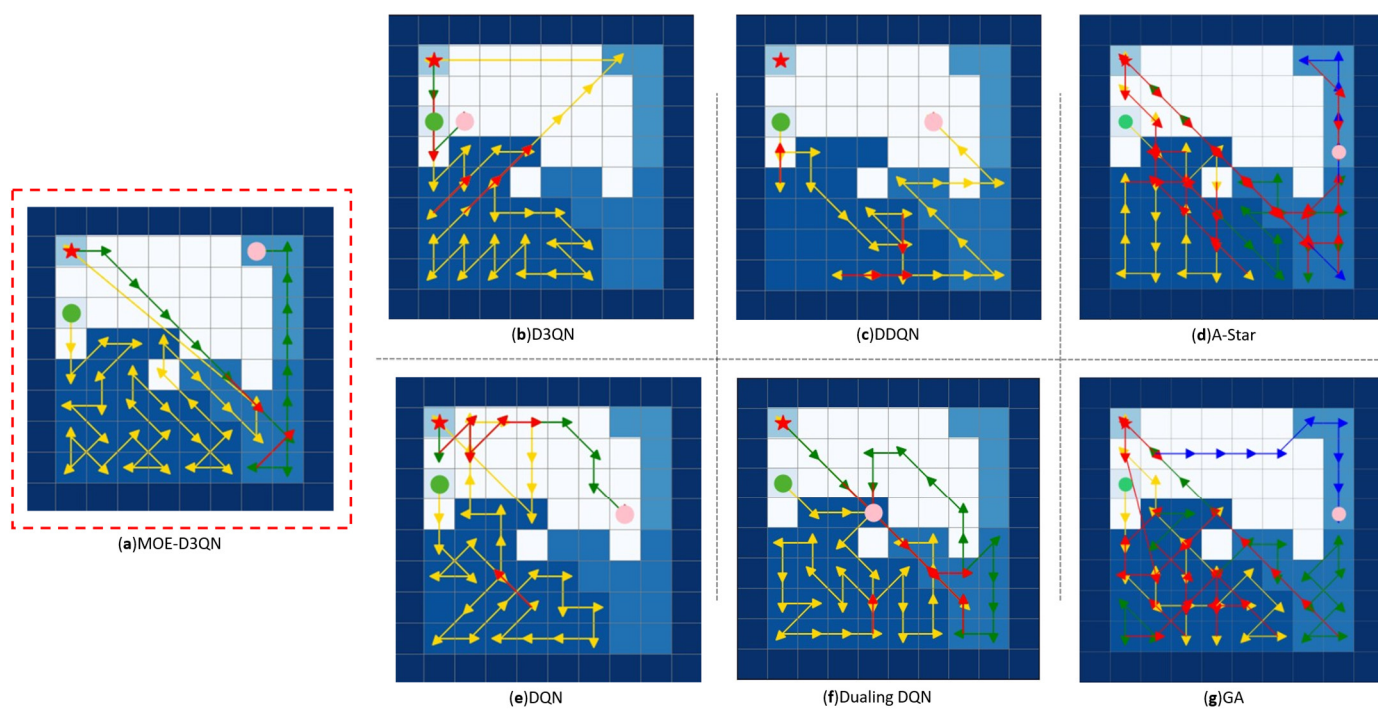


Figure 6. Three-level task planning result diagram.

**Table 5.** Experimental data statistics.

| Number of Task Areas | Algorithms  | Step | $Area_{repeat}$ | $Area_{cover}$ | Repeat (%) | Coverage Efficiency | Precedence Constraints |
|----------------------|-------------|------|-----------------|----------------|------------|---------------------|------------------------|
| 2                    | MOE-D3QN    | 37   | 1               | 31             | 3.23%      | 0.8378              | Yes                    |
|                      | D3QN        | 26   | 4               | 15             | 26.67%     | 0.5769              | Yes                    |
|                      | DQN         | 39   | 9               | 20             | 45.00%     | 0.5128              | Yes                    |
|                      | DDQN        | 37   | 2               | 26             | 7.69%      | 0.7027              | Yes                    |
|                      | Dueling DQN | 32   | 3               | 21             | 14.29%     | 0.6563              | Yes                    |
|                      | A-Star      | 51   | 13              | 31             | 41.94%     | 0.6078              | Yes                    |
|                      | GA          | 51   | 12              | 31             | 38.71%     | 0.6078              | Yes                    |
| 3                    | MOE-D3QN    | 46   | 2               | 38             | 5.26%      | 0.8261              | Yes                    |
|                      | D3QN        | 32   | 5               | 24             | 20.83%     | 0.75                | Yes                    |
|                      | DQN         | 40   | 5               | 22             | 22.73%     | 0.55                | Yes                    |
|                      | DDQN        | 27   | 4               | 19             | 21.05%     | 0.7037              | NO                     |
|                      | Dueling DQN | 50   | 8               | 33             | 24.24%     | 0.66                | Yes                    |
|                      | A-Star      | 79   | 35              | 38             | 92.11%     | 0.481               | Yes                    |
|                      | GA          | 70   | 22              | 38             | 57.89%     | 0.5429              | Yes                    |

(a) Path length: UAV's movement steps.

(b) Coverage efficiency: ratio of effectively covered area to total target region area, as shown in Equation (19).

$$E = \frac{Area_{cover}}{Step} \times 100\% \quad (19)$$

(c) Redundancy rate: ratio of repeatedly visited areas to total covered regions in the planned path, as formalized in Equation (20)

$$R_{repeat} = \frac{Area_{repeat}}{Area_{cover}} \times 100\% \quad (20)$$

### 3.3. Experimental Analysis

The reward curves presented in Figures 7 and 8 illustrate the training process, where the MOE-D3QN algorithm initially exhibited a gradual upward trend. Due to the exploration-promoting action selection strategy employed in this study, the overall reward curve demonstrated a spiraling increase. During the early to mid-training phases, the model progressively learned to understand environmental constraints and developed effective action strategies. By the mid-training stage, the model had achieved a satisfactory comprehension of the task requirements, successfully completing the planning task and obtaining the final reward. However, in the later stages, due to the chosen training strategy, the model showed signs of overfitting, resulting in fluctuations in the reward values.

In contrast, the DQN algorithm suffered from policy update bias accumulation caused by Q-value overestimation, leading to premature convergence in complex state spaces. Although DDQN mitigated some of these biases by decoupling action selection from value evaluation, its fixed-interval parameter update mechanism limited its adaptability to dynamic environmental changes. The dueling network architecture of Dueling DQN, while advantageous in some contexts, was prone to advantage function estimation errors in sparse reward scenarios, which increased the likelihood of becoming trapped in local optima.

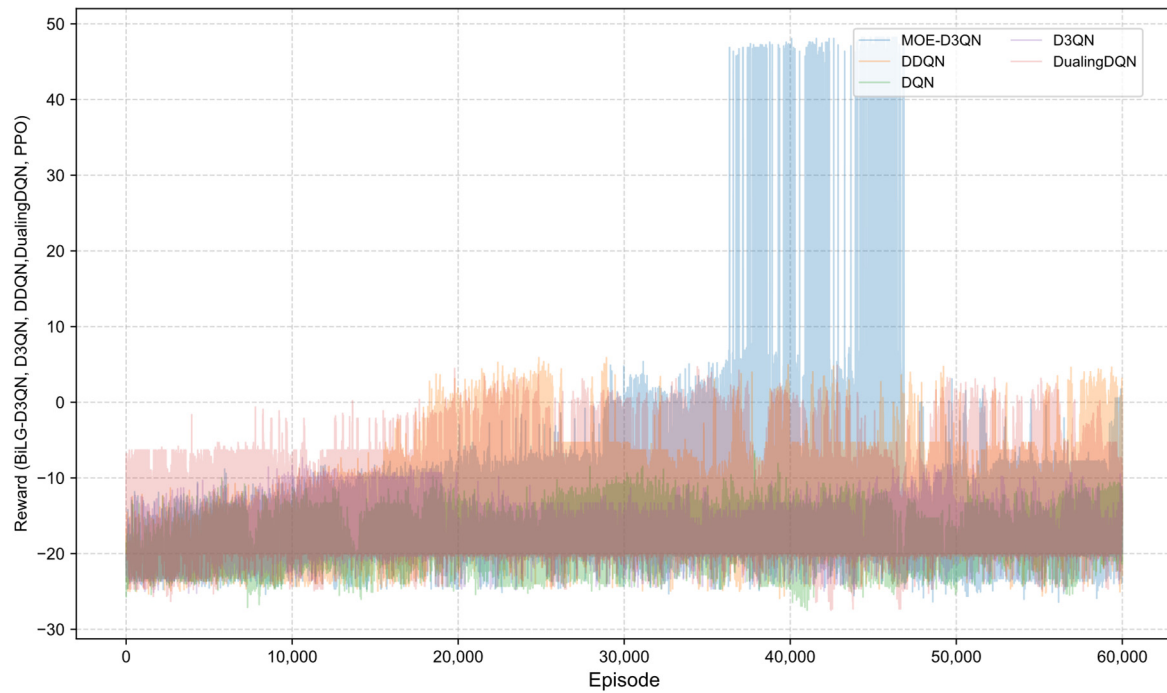


**Figure 7.** The two-level task planning algorithm reward value comparison diagram.

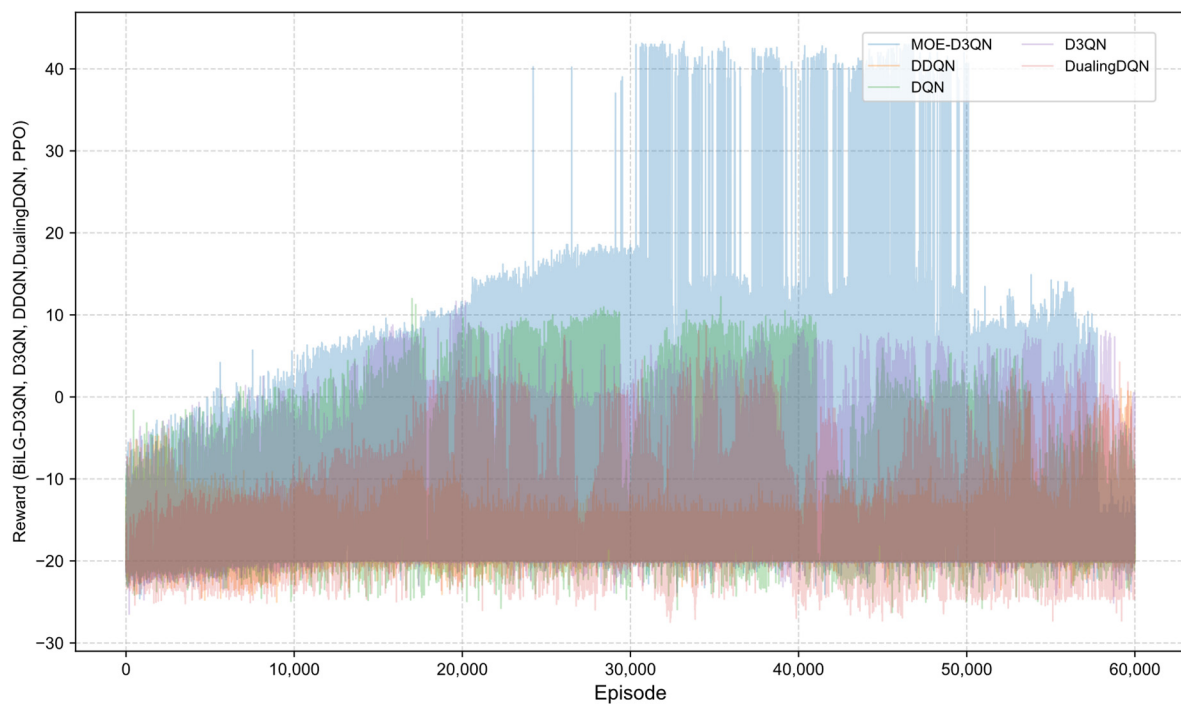


**Figure 8.** The three-level task planning algorithm reward value comparison diagram.

#### 4. Discussion

In terms of experimental metrics, MOE-D3QN demonstrated significantly higher coverage efficiency and lower redundancy rates compared to traditional algorithms and mainstream reinforcement learning methods. This not only reflects the optimization advantages of the path planning strategy but also holds profound practical significance for agricultural applications. Firstly, the high coverage efficiency ensures comprehensive and omission-free treatment of target areas for operations such as pesticide spraying and

fertilization, effectively minimizing blind spots and coverage gaps. This is particularly crucial under conditions of high pest incidence or uneven crop growth, where the risk of “over-treatment and under-protection” or “localized loss of control” can be largely avoided. Secondly, the enhanced coverage efficiency markedly improves operational productivity. For plant protection UAVs, this translates into the ability to complete larger-area tasks within a single flight, thereby reducing the need for multiple flights and lowering the frequency and cost associated with battery recharging or replacement. Consequently, labor input and time consumption are further reduced.

Simultaneously, the low redundancy rate indicates a reduced frequency of path overlap during mission execution, facilitating precise control of agricultural inputs such as pesticides, fertilizers, and irrigation water. Traditional path planning methods often suffer from significant path overlap issues, leading to excessive pesticide or fertilizer concentrations in certain areas, which not only increase production costs but may also cause crop phytotoxicity and soil accumulation pollution, posing ecological risks. Conversely, the low-redundancy path planning approach ensures that each target area is uniformly and efficiently covered, promoting relative controllability of agricultural resources. This approach balances both the “precision” and “uniformity” of operational outcomes, safeguarding crop health while minimizing environmental burden—an essential factor for advancing green agriculture and resource-efficient development.

From a technical implementation perspective, this study utilized Sentinel-2 Level-2A satellite remote sensing imagery to perform spectral inversion of vegetation status in the study area, extracting the Canopy Chlorophyll Content index. Subsequently, multi-level thresholds were established using quantile methods to rasterize the imagery into a digital task map, where different grid values represented varying operational priorities (e.g., high, medium, and low risk zones). These were uniformly fed into the state space of the deep reinforcement learning model. This processing approach effectively transforms remote sensing perception data into structured inputs recognizable by the reinforcement learning agent, thereby simplifying the linkage between remote sensing information and path decision making.

Regarding the specific implementation, the rasterization method employed offers strong generalizability and operability, capable of accommodating task partitioning requirements across diverse crop scenarios. Although the current experiments were conducted in flat farmland areas, the CCC index used is derived from red-edge and near-infrared bands, possessing a degree of crop non-specificity. Theoretically, it is applicable to physiological status estimation for various crops such as rice, wheat, and soybean, thus exhibiting cross-crop adaptability potential. Moreover, the MOE-D3QN model employs a Mixture-of-Experts architecture, composed of multiple policy subnetworks dynamically fused through a gating mechanism based on input states. This architecture offers advantages in modeling heterogeneous input states and is expected to enhance the model’s responsiveness to environmental heterogeneity and task complexity.

Despite the current experiments focusing primarily on relatively flat terrain, the model’s structural design does not rely on specific landform types or single-crop characteristics, indicating certain scalability. Preliminary assessments suggest that, with appropriate adjustments to state inputs and task configurations, the model may possess adaptability for more complex terrains and multi-crop structured scenarios.

## 5. Conclusions

This study proposes an improved deep reinforcement learning algorithm, MoE-D3QN, which integrates D3QN, a Mixture-of-Experts mechanism, and a Bi-directional Long Short-Term Memory, aiming to enhance path planning efficiency for agricultural unmanned aerial

vehicles operating in dynamic and complex environments. The algorithm leverages the Bi-LSTM module's capability to capture temporal features, thereby strengthening contextual awareness within task sequences. Moreover, through the dynamic selection capability of expert networks and gating mechanisms in the MoE architecture, it enables adaptive decision making in response to variable states within complex environments. In addition, a hierarchical coverage path planning framework is developed, utilizing Sentinel-2 remote sensing data to invert canopy chlorophyll content and construct multi-level task maps via thresholding and rasterization techniques. A dynamic energy consumption modeling mechanism is introduced to quantify the coupling effects of payload and environmental factors. Furthermore, a progressive composite reward function is designed to achieve synergistic optimization of energy efficiency and task priority.

Simulation experiments demonstrate that MOE-D3QN achieves superior performance compared to six other algorithms in hierarchical coverage path planning tasks with load-related energy considerations. In the secondary task scenario, the MOE-D3QN algorithm achieved a coverage efficiency of 0.8378, representing improvements of 45.26% over D3QN, 63.38% over DQN, 19.19% over DDQN, 27.66% over Dueling DQN, and 37.84% over A-Star and GA. The redundancy rate was only 3.23%, far lower than D3QN at 26.67%, DQN at 45.00%, DDQN at 7.69%, Dueling DQN at 14.29%, A-Star at 41.94%, and GA at 38.71%. In the tertiary task scenario, the MOE-D3QN algorithm achieved a coverage efficiency of 0.8261, with improvements of 10.15% over D3QN, 50.2% over DQN, 17.4% over DDQN, 25.17% over Dueling DQN, 71.45% over A-Star, and 52.13% over GA. The redundancy rate was 5.26%, significantly better than D3QN at 20.83%, DQN at 22.73%, DDQN at 21.05%, Dueling DQN at 24.24%, A-Star at 92.11%, and GA at 57.89%.

These results validate the algorithm's capability to comprehend task constraints and handle temporal information, providing a low-redundancy and highly adaptive path planning solution for precision agricultural UAV operations. The proposed approach not only enhances spraying accuracy and resource utilization but also reduces pesticide overuse and environmental pollution risks, contributing to the advancement of intelligent and sustainable agriculture.

In future research, efforts will be directed towards further expanding the practical applicability of the proposed algorithm to better support UAV systems in precision agriculture, thereby promoting agricultural intelligence and sustainable development. Specifically, the following aspects will be addressed:

1. This study will focus on the collaborative mechanisms of UAV path planning among various types of agricultural intelligent devices. Emphasis will be placed on the information interaction and operational coordination between UAVs and field sensing equipment—such as soil moisture sensors and microclimate stations—as well as variable-rate pesticide application systems. By constructing a unified crop state perception and task collaboration framework, the coupling among multi-device platforms will be enhanced, facilitating the evolution of UAV operation systems from independent execution towards intelligent collaborative operation. This will provide crucial support for the integration and intelligent application of multiple agricultural devices in precision farming.
2. To meet the demands for efficient operations across large areas and multi-crop distributions, future work will consider coordinated path planning for multi-UAV systems. By integrating plot characteristics, crop requirements, and operational time window constraints, dynamic scheduling of multiple UAVs will be realized. This approach will offer algorithmic support for UAV swarm formation and zonal operations, thus advancing the large-scale deployment of UAVs in precision agriculture.
3. To further enhance the algorithm's generalizability across diverse scenarios, subsequent research plans include incorporating real orchard point cloud data or digital elevation

models (DEM) to embed terrain factors such as slope and undulation into the state space construction. This will enable three-dimensional path planning environments that more closely align with practical operational needs. Experimental evaluations will be conducted in irregular terrains such as hilly areas and orchards, providing more comprehensive empirical support for the model's applicability in varied agricultural settings.

**Author Contributions:** Conceptualization, H.F., H.L., Z.L. and J.L. (Jian Li); methodology, W.Z. and Y.F.; software, H.F. and L.Z.; investigation, L.Z., W.Z. and J.L. (Jian Lu); resources, Z.L., H.L. and Y.F.; writing—original draft, Z.L.; writing—review and editing, H.F. and J.L. (Jian Li); visualization, Z.L.; supervision, H.F., H.L. and W.Z.; funding acquisition, L.Z.; validation, L.Z., Y.F. and J.L. (Jian Lu); data curation, H.F. and J.L. (Jian Lu); project administration, J.L. (Jian Li). All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Jilin Provincial Department of Education Science and Technology Research Project (JJKH20250567KJ), the Scientific Research Project of the Jilin Provincial Department of Education (JJKH20250574BS), and the Demonstration Project for the Construction of a Provincial-Level Modern Agricultural Industry Technology System (JLARS-2025-010216).

**Data Availability Statement:** Data is contained within the article.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Maes, W.H.; Steppe, K. Perspectives for Remote Sensing with Unmanned Aerial Vehicles in Precision Agriculture. *Trends Plant Sci.* **2019**, *24*, 152–164. [\[CrossRef\]](#) [\[PubMed\]](#)
2. Shakhathreh, H.; Sawalmeh, A.H.; Al-Fuqaha, A.; Dou, Z.; Almaita, E.; Khalil, I.; Othman, N.S.; Khreishah, A.; Guizani, M. Unmanned Aerial Vehicles (UAVs): A Survey on Civil Applications and Key Research Challenges. *IEEE Access* **2019**, *7*, 48572–48634. [\[CrossRef\]](#)
3. Rejeb, A.; Abdollahi, A.; Rejeb, K.; Treiblmaier, H. Drones in Agriculture: A Review and Bibliometric Analysis. *Comput. Electron. Agric.* **2022**, *198*, 107017. [\[CrossRef\]](#)
4. Guebsi, R.; Mami, S.; Chokmani, K. Drones in Precision Agriculture: A Comprehensive Review of Applications, Technologies, and Challenges. *Drones* **2024**, *8*, 686. [\[CrossRef\]](#)
5. Canicatti, M.; Vallone, M. Drones in Vegetable Crops: A Systematic Literature Review. *Smart Agric. Technol.* **2024**, *7*, 100396. [\[CrossRef\]](#)
6. Hafeez, A.; Husain, M.A.; Singh, S.P.; Chauhan, A.; Khan, M.T.; Kumar, N.; Chauhan, A.; Soni, S.K. Implementation of Drone Technology for Farm Monitoring & Pesticide Spraying: A Review. *Inf. Process. Agric.* **2023**, *10*, 192–203. [\[CrossRef\]](#)
7. Mukhamediev, R.I.; Yakunin, K.; Aubakirov, M.; Assanov, I.; Kuchin, Y.; Symagulov, A.; Levashenko, V.; Zaitseva, E.; Sokolov, D.; Amirgaliyev, Y. Coverage Path Planning Optimization of Heterogeneous UAVs Group for Precision Agriculture. *IEEE Access* **2023**, *11*, 5789–5803. [\[CrossRef\]](#)
8. Radoglou-Grammatikis, P.; Sarigiannidis, P.; Lagkas, T.; Moscholios, I. A Compilation of UAV Applications for Precision Agriculture. *Comput. Netw.* **2020**, *172*, 107148. [\[CrossRef\]](#)
9. Tang, G.; Tang, C.; Claramunt, C.; Hu, X.; Zhou, P. Geometric A-Star Algorithm: An Improved A-Star Algorithm for AGV Path Planning in a Port Environment. *IEEE Access* **2021**, *9*, 59196–59210. [\[CrossRef\]](#)
10. Wang, D.; Tan, D.; Liu, L. Particle Swarm Optimization Algorithm: An Overview. *Soft Comput.* **2018**, *22*, 387–408. [\[CrossRef\]](#)
11. Qi, J.; Yang, H.; Sun, H. MOD-RRT\*: A Sampling-Based Algorithm for Robot Path Planning in Dynamic Environment. *IEEE Trans. Ind. Electron.* **2021**, *68*, 7244–7251. [\[CrossRef\]](#)
12. Zhao, Y.; Qi, X.; Ma, Y.; Li, Z.; Malekian, R.; Sotelo, M.A. Path Following Optimization for an Underactuated USV Using Smoothly-Convergent Deep Reinforcement Learning. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 6208–6220. [\[CrossRef\]](#)
13. Wen, S.; Wen, Z.; Zhang, D.; Zhang, H.; Wang, T. A Multi-Robot Path-Planning Algorithm for Autonomous Navigation Using Meta-Reinforcement Learning Based on Transfer Learning. *Appl. Soft. Comput.* **2021**, *110*, 107605. [\[CrossRef\]](#)
14. Zhang, M.; Li, X.; Wang, L.; Jin, L.; Wang, S. A Path Planning System for Orchard Mower Based on Improved A\* Algorithm. *Agronomy* **2024**, *14*, 391. [\[CrossRef\]](#)
15. Wu, G.; Wang, M.; Guo, L. Complete Coverage Path Planning Based on Improved Genetic Algorithm for Unmanned Surface Vehicle. *J. Mar. Sci. Eng.* **2024**, *12*, 1025. [\[CrossRef\]](#)
16. Kong, F.; Wang, Q.; Gao, S.; Yu, H. B-APFDQN: A UAV Path Planning Algorithm Based on Deep Q-Network and Artificial Potential Field. *IEEE Access* **2023**, *11*, 44051–44064. [\[CrossRef\]](#)

17. Zhou, Y.; Yang, J.; Guo, Z.; Shen, Y.; Yu, K.; Lin, J.C.-W. An Indoor Blind Area-Oriented Autonomous Robotic Path Planning Approach Using Deep Reinforcement Learning. *Expert Syst. Appl.* **2024**, *254*, 124277. [\[CrossRef\]](#)
18. Lin, G.; Zhu, L.; Li, J.; Zou, X.; Tang, Y. Collision-Free Path Planning for a Guava-Harvesting Robot Based on Recurrent Deep Reinforcement Learning. *Comput. Electron. Agric.* **2021**, *188*, 106350. [\[CrossRef\]](#)
19. Din, A.; Ismail, M.Y.; Shah, B.; Babar, M.; Ali, F.; Baig, S.U. A Deep Reinforcement Learning-Based Multi-Agent Area Coverage Control for Smart Agriculture. *Comput. Electr. Eng.* **2022**, *101*, 108089. [\[CrossRef\]](#)
20. Syed, T.N.; Zhou, J.; Lakhari, I.A.; Marinello, F.; Gemechu, T.T.; Rottok, L.T.; Jiang, Z. Enhancing Autonomous Orchard Navigation: A Real-Time Convolutional Neural Network-Based Obstacle Classification System for Distinguishing ‘Real’ and ‘Fake’ Obstacles in Agricultural Robotics. *Agriculture* **2025**, *15*, 827. [\[CrossRef\]](#)
21. Li, J.; Zhang, W.; Ren, J.; Yu, W.; Wang, G.; Ding, P.; Wang, J.; Zhang, X. A Multi-Area Task Path-Planning Algorithm for Agricultural Drones Based on Improved Double Deep Q-Learning Net. *Agriculture* **2024**, *14*, 1294. [\[CrossRef\]](#)
22. Qu, L.; Zhang, P.; Jing, C.; Du, M.; Wang, J.; Zhao, Q.; Li, J. Estimating the Fractional Cycle Biases for GPS Triple-Frequency Precise Point Positioning with Ambiguity Resolution Based on IGS Ultra-Rapid Predicted Orbits. *Remote Sens.* **2021**, *13*, 3164. [\[CrossRef\]](#)
23. Clevers, J.G.P.W.; Kooistra, L.; Van den Brande, M.M.M. Using Sentinel-2 Data for Retrieving LAI and Leaf and Canopy Chlorophyll Content of a Potato Crop. *Remote Sens.* **2017**, *9*, 405. [\[CrossRef\]](#)
24. Li, D.; Chen, G.; Xi, Z.; He, P.; Xin, X.; Chen, J.; Yu, H.; Kang, G. Spatiotemporal Changes of Cultivated Land Utilization in Black Soil Region, China Based on Geo-Information Tupu. *Sci. Rep.* **2025**, *15*, 10306. [\[CrossRef\]](#)
25. Wu, W.; Feng, X.; Lu, C. The rise of smart agriculture in China: Current situation and suggestions for further development. *Exp. Agric.* **2024**, *60*, e28. [\[CrossRef\]](#)
26. Brown, L.A.; Ogutu, B.O.; Dash, J. Estimating Forest Leaf Area Index and Canopy Chlorophyll Content with Sentinel-2: An Evaluation of Two Hybrid Retrieval Algorithms. *Remote Sens.* **2019**, *11*, 1752. [\[CrossRef\]](#)
27. Brown, L.A.; Fernandes, R.; Djamaï, N.; Meier, C.; Gobron, N.; Morris, H.; Canisius, F.; Bai, G.; Lerebourg, C.; Lanconelli, C.; et al. Validation of Baseline and Modified Sentinel-2 Level 2 Prototype Processor Leaf Area Index Retrievals over the United States. *ISPRS J. Photogramm. Remote Sens.* **2021**, *175*, 71–87. [\[CrossRef\]](#)
28. Validation of the Sentinel Simplified Level 2 Product Prototype Processor (SL2P) for mapping cropland biophysical variables using Sentinel-2/MSI and Landsat-8/OLI data | Request PDF. *Remote Sens. Environ.* **2024**, *225*, 416–430. [\[CrossRef\]](#)
29. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-Level Control through Deep Reinforcement Learning. *Nature* **2015**, *518*, 529–533H. [\[CrossRef\]](#)
30. Hasselt, H.; van Guez, A.; Silver, D. Deep Reinforcement Learning with Double Q-Learning. *arXiv* **2015**, arXiv:1509.0646. [\[CrossRef\]](#)
31. Wang, Z.; Schaul, T.; Hessel, M.; Hasselt, H.; van Lanctot, M.; Freitas, N. de Dueling Network Architectures for Deep Reinforcement Learning. *arXiv* **2016**, arXiv:1511.06581.
32. Lepikhin, D.; Lee, H.; Xu, Y.; Chen, D.; Firat, O.; Huang, Y.; Krikun, M.; Shazeer, N.; Chen, Z. GShard: Scaling Giant Models with Conditional Computation and Automatic Sharding. *arXiv* **2022**, arXiv:2006.16668.
33. Fedus, W.; Zoph, B.; Shazeer, N. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *J. Mach. Learn. Res.* **2022**, *23*, 5232–5270.

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