UCAV Path Planning Algorithm Based on Deep Reinforcement Learning

Kaiyuan Zheng, Jingpeng Gao(✉), and Liangxi Shen

College of Information and Communication Engineering, Harbin Engineering University,
Heilongjiang 150001, China
gaojingpeng@hrbeu.edu.cn

Abstract. In the field of the Unmanned Combat Aerial Vehicle (UCAV) confrontation, traditional path planning algorithms have slow operation speed and poor adaptability. This paper proposes a UCAV path planning algorithm based on deep reinforcement learning. The algorithm combines the non-cooperative game idea to build the UCAV and radar confrontation model. In the model, the UCAV must reach the target area. At the same time, in order to complete the identification of the radar communication signal based on ResNet-50 migration learning, we use the theory of Cyclic Spectrum(CS) to process the signal. With the kinematics mechanism of the UCAV, the radar detection probability and the distance between the UCAV and center of the target area are proposed as part of the reward criteria. And we make the signal recognition rate as another part of the reward criteria. The algorithm trains the Deep Q-Network(DQN) parameters to realize the autonomous planning of the UCAV path. The simulation results show that compared with the traditional reinforcement learning algorithm, the algorithm can improve the system operation speed. The accuracy reaches 90% after 300 episodes and the signal recognition rate reaches 92.59% under 0 dB condition. The proposed algorithm can be applied to a variety of electronic warfare environment. It can improve the maneuver response time of the UCAV.

Keywords: UCAV · Signal recognition · Path planning · Cyclic spectrum · Reward criteria · Deep Q-Network

1 Introduction

The appearance of new system radar brings new requirements to the UCAV on electronic warfare, thus the UCAV path planning has become an urgent problem. Good path planning can improve the safety performance of the UCAV and help UCAV accomplish their tasks well.

Currently, the main methods of UCAV path planning include intelligent algorithms, and neural networks [1]. Sun proposes quantum genetic algorithm for mobile robot path planning [2]. He guides and realizes path optimization by introducing genetic operators including quantum crossover operator and quantum gate mutation operator with the essential characteristics of quantum. However, the algorithm is easy to fall into local extreme points and the convergence speed is slow. To achieve a fast search of the path,

Y. Zhao et al. (Eds.): ICIG 2019, LNCS 11902, pp. 702–714, 2019.
https://doi.org/10.1007/978-3-030-34110-7_59

Wang uses the fuzzy neural network to plan the path of the mobile robot [3]. But in electronic warfare, there is a lack of training samples, resulting in poor applicability. In order to overcome the shortcomings of small samples, Peng proposes the 3-D path planning with Multi-constrains [4]. He takes multi-constrains into account in the planning scheme. A path is generated by searching in the azimuth space using genetic algorithm and geometry computation. It causes the algorithm to converge slowly.

With the development of artificial intelligence, more path planning algorithms have been proposed [5–8]. In [9], Q-learning-based path planning algorithm is presented to find a target in the maps which are obtained by mobile robots. Q-learning is a kind of reinforcement learning algorithm that detects its environment. It shows a system which makes decisions itself that how it can learn to make true decisions about reaching its target. However, because the limitations of the Q table of the algorithm [10], its calculation accuracy is poor. Although he uses the reward criteria, the reward values is too single, which makes it less accurate.

Under the UCAV and radar confrontation model, this paper proposes a UCAV path planning algorithm based on deep reinforcement learning to solve the UCAV signal recognition and path planning problem. In order to enable the UCAV to complete the task of identifying the radar signal while reaching the target area, the proposed method combines the neural network of deep learning with the reward criteria of reinforcement learning. We set the reasonable reward values, state values and action values of the UCAV. Then, we use ResNet migration learning to improve the recognition rate of radar signals. The Deep Q-Network is trained to realize adaptive generation of UCAV path. Finally, simulation experiments verify the effectiveness of the proposed method.

The structure of this paper is organized as follows: The model of deep reinforcement learning is described in the Sect. 2. The Sect. 3 introduces the system structure of the path planning. It includes confrontation model and training constraints. The Sect. 4 shows the simulation results. The conclusion will be discussed in Sect. 5.

2 Deep Reinforcement Learning

On one hand, the traditional reinforcement learning algorithms rely on human-involved feature design [11], on the other hand they rely on approximations of values functions and strategy functions. Deep learning, especially Convolutional Neural Networks (CNN), can extract high-dimensional features of images [12]. The Google technical team combines the CNN in deep learning with the Q-learning algorithm in reinforcement learning. They propose the Deep Q-Network (DQN) algorithm [13]. As a pioneering work of deep reinforcement learning, the DQN can finish end-to-end learning from perception to action.

Figure 1 shows the structure of the DQN algorithm. DQN effectively removes the instability and divergence caused by neural network nonlinear action values approximator. It greatly improves the applicability of reinforcement learning. First, the experience replay in the figure randomizes the data. Thereby, it can remove the correlation between the observed data, smooth the data distribution and increase the utilization of historical data. Secondly, the CNN is used to replace the traditional reinforcement learning table mechanism. By using two networks to iteratively update, the algorithm uses the DQN loss function to adjust the direction of the current network values toward the target values network. It is also periodically updated to reduce the correlation with the target network. In addition, through truncating the rewards and regularizing the network parameters, the gradient is limited to the appropriate range, resulting in a more robust training process.

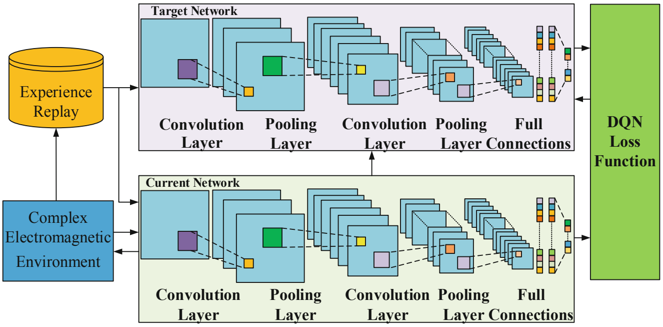


Fig. 1. The structure of DQN model.

3 UCAV Path Planning Research Program

Aiming at the problem that the UCAV avoids the ground radar detection to effectively break through the defense area and accurately identify radar signals, the deep reinforcement learning algorithm is used to train the model to realize the automatic generation of the UCAV path. This paper considers how to plan a reasonable path for a UCAV to arrive at the target area, which makes the UCAV not detected by the two radar.

3.1 Path Planning System Structure

The structure of the path planning system of the UCAV is shown in Fig. 2. Under the background of complex electromagnetic environment, we build a model of the UCAV and radar confrontation. On the one hand, we use the radar signal-to-noise ratio to calculate the radar detection probability. On the other hand, we analyze the UCAV kinematic constraints to study the state values and action values of the UCAV in intelligent decision. These two factors are combined as the reward criteria of the UCAV path planning, which is used to train the intelligent decision network parameters of the UCAV. Finally, a real and reliable path of the UCAV is obtained according to the result.

3.2 Confrontation Model and Training Constraints

In order to simulate the real confrontation environment, it is assumed that there are two random search radars. The distance between the radars satisfies:

$$0.5R_{\max} \le d < 3R_{\max} \tag{1}$$

where $R_{\max}$ is the maximum working distance of the radar.

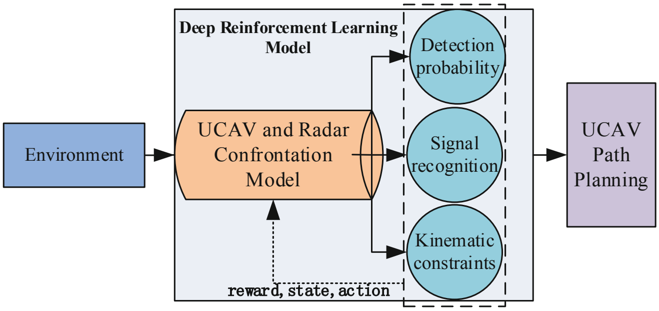


Fig. 2. The structure of path planning system.

The initial position of UCAV is fixed, and the target area is the ground hemisphere with a radius of d_0. After establishing the battlefield environment, the training constraints for the path planning include the reward values, the state values, and the action values constraints. These three types of constraints are the basis for training the parameters of the UCAV intelligent decision network.

Kinematic constraints of the UCAV

Set the random speed vector of the UCAV to:

$$A = [v_x, v_y, v_z] \tag{2}$$

where: v_x is the velocity component of the north direction of the UCAV; v_y is the velocity component of the east direction of the UCAV; v_z is the velocity component of the UCAV facing upward.

The UCAV uses uniform motion:

$$\sqrt{v_x^2 + v_y^2 + v_z^2} = C_0 \tag{3}$$

where C_0 is the UCAV speed values.

The coordinate position $[x, y, z]$ of the UCAV is taken as the state values S, where x represents the coordinates of the north of the UCAV, y represents the coordinates of the east, and z represents the height of the UCAV, and satisfies $0 \leq z \leq 2.5\,\text{km}$.

After moving for Δt time, according to the action constraints and the current state values, the next state values of the UCAV is:

$$S^{'} = [x + v_x \Delta t, y + v_y \Delta t, z + v_z \Delta t] \tag{4}$$

The action values of UCAV after Δt time is:

$$A' = [v_x', v_y', v_z'] \tag{5}$$

The action values also needs to satisfy (6):

$$\sqrt{v_x'^2 + v_y'^2 + v_z'^2} = C_0 \tag{6}$$

When the state values of the UCAV satisfies (7), it can be determined that the UCAV has reached the target area:

$$\sqrt{(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2} \le d_0 \tag{7}$$

where:

$[x_0, y_0, z_0]$ is the target point coordinate;

d_0 is the target area radius.

Signal recognition

The traditional signal recognition based on radar characteristic parameters can not meet the recognition requirements in complex electromagnetic environment in modern electronic countermeasures. In recent years, the development of deep learning, especially the widespread use of neural networks, has provided new ideas and methods for signal processing and recognition. The recognition based on neural network image features can be well applied in the field of radar signal recognition. However, the reasonable conversion of radar signals into images is the key to reliable identification.

First, generating a reliable signal is the basis for signal processing:

$$x(t) = ae^{j\varphi(t)} + n(t) \tag{8}$$

where a is the amplitude, we suppose $a = 1$ in this paper; $\varphi(t)$ is the instantaneous phase of the radar signal; $n(t)$ is the white Gaussian noise. The radar communication signal are BPSK, QPSK, 8PSK, ASK, OQPSK, QAM16, QAM32, QAM64, QAM256.

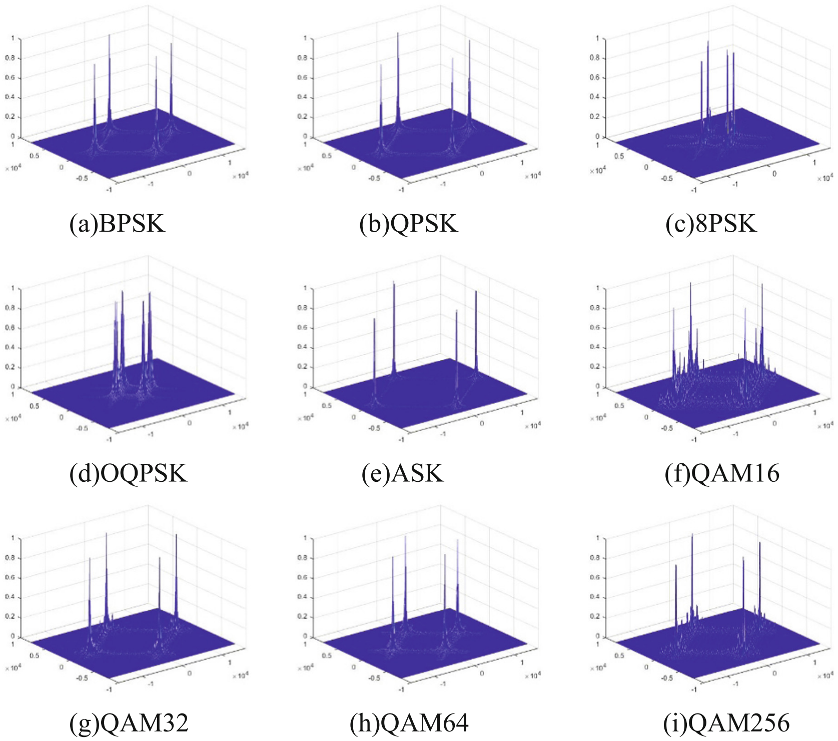
The theory of Cyclic Spectrum(CS) is established based on the cyclic and stationary characteristics. The signal processing of the communication signal by the cyclic spectrum can obtain good results. Therefore, after receiving the communication modulation signal, the signal processing of the cyclic spectrum analysis is performed:

$$S_x^\alpha(f) = \int_{-\infty}^{\infty} R_x^\alpha(\tau)e^{-j2\pi f\tau}d\tau \tag{9}$$

where α is the cyclic frequency and $R_x^\alpha(\tau)$ is the cyclic autocorrelation of the signal $x(t)$:

$$R_x^\alpha(\tau) = \lim_{T\to\infty}\frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} x(t + \frac{\tau}{2})x^*(t - \frac{\tau}{2})e^{-j2\pi\alpha t}dt \tag{10}$$

Figure 3 shows nine types of radar communication signals according formula (10).



(a)BPSK (b)QPSK (c)8PSK

(d)OQPSK (e)ASK (f)QAM16

(g)QAM32 (h)QAM64 (i)QAM256

Fig. 3. Nine types of radar communication signals

Compared with the common convolutional neural network, the ResNet network mainly adds a shortcut connection between the input and the output, so that the network can make the subsequent layer can directly learn the residual. When the traditional convolutional layer or fully connected layer is used for information transmission, there will be problems of information loss due to inconvenient connection between input and output. ResNet solves this problem to some extent, and the ResNet network passes the input information. Therefore, we use the ResNet network to extract features from the three-dimensional spectrum of the cyclic spectrum of the communication signal. Migration learning is a new machine learning method. It has good adaptability and can improve the quality of feature extraction. Therefore, the above network application migration learning is adapted to the communication field. Figure 4 shows the migration process of the ResNet. We use pre-training model ResNet to process the signal. Finally, the system uses the Support Vector Machine(SVM) to reach the purpose of classification and identification and get the communication signal recognition rate η.

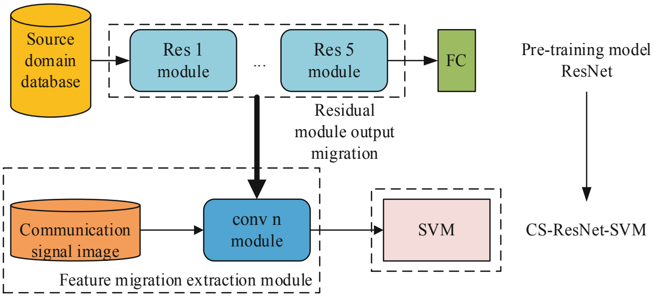

Source domain database
Res 1 module
...
Res 5 module
FC
Pre-training model ResNet
Residual module output migration
Communication signal image
conv n module
SVM
CS-ResNet-SVM
Feature migration extraction module

Fig. 4. The migration process of the ResNet.

3.3 Reward Criteria

In order to obtain a reasonable UCAV trajectory in the network training of the model, we set the appropriate radar detection probability and the distance between the UCAV and the center of the target area as the reward values of the intelligent decision network. On this basis, we will use the processed signal recognition rate as a supplement to the reward mechanism to achieve the purpose of the multitasking of the UCAV.

The detection probability P_d of the radar is an important indicator for the effective penetration of the UCAV. When $P_d \leq 0.1$, the radar does not find the UCAV; when $P_d > 0.1$, the radar finds the UCAV. In order to train the UCAV to reach the target area in the shortest time without being detected by the radar, the reward values of the UCAV path planning is:

$$R(\boldsymbol{\omega}) = P_d(\omega_1) + D(\omega_2) \tag{11}$$

where: D is the distance between the UCAV and the center of the target area; $\boldsymbol{\omega}= [\omega_1, \omega_2]$ are the weights of the detection probability and the flight time of the UCAV respectively. Different weights can get different reward trends. In this paper, we set the reward as:

$$R = \frac{\omega_1}{\log_2 P_d^{-1}} - \omega_2(D - 250)^3 + 2 \tag{12}$$

where: $\omega_1 = 1$, $\omega_2 = 0.001$.

However, in order to enable the UCAV to better identify the radar communication signal. We improve the reward with communication signal recognition rate η:

$$R = \eta(\frac{\omega_1}{\log_2 P_d^{-1}} - \omega_2(D - 250)^3 + 2) \tag{13}$$

3.4 Path Planning Algorithm

Intelligent decision making is the core of the UCAV path planning algorithm. In the traditional reinforcement learning algorithm, the state and action space are discrete and the dimension is low. Q table can be used to store the Q values of each state action. In solving the problem of UCAV path planning, the state and action of the UCAV are highly dimensionally continuous, and the data is also very large. Due to the limitations of the Q table, it is very difficult to store data, which makes the traditional reinforcement learning algorithm cannot solve the problem of path planning. The DQN algorithm replaces the Q table by fitting a loss function so that similar states get similar action outputs. We propose that the path planning algorithm based on deep reinforcement learning, which can effectively solve the problem that the data is too large to be trained. Besides, it can break the correlation between data and improve the training efficiency. We have discussed the three elements of the DQN algorithm: state, action, and reward. Figure 4 shows the DQN algorithm flow.

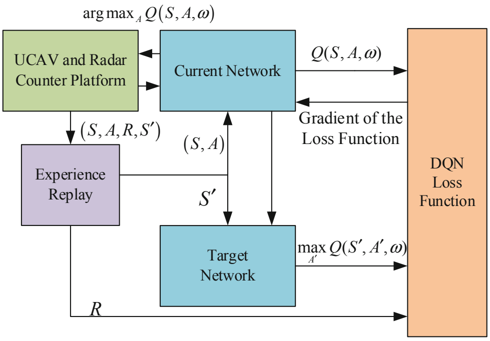

$\arg\max_A Q(S,A,\omega)$
UCAV and Radar Counter Platform
Current Network
$Q(S,A,\omega)$
Gradient of the Loss Function
(S,A,R,S')
(S,A)
Experience Replay
S'
DQN Loss Function
Target Network
$\max_{A'} Q(S',A',\omega)$
R

Fig. 5. DQN algorithm flow.

In Fig. 5, the experience replay is used to learn the previous experience. We send (S, A, R, S') to the experience replay for learning. At the same time, we send the output reward values R to the DQN loss function. The current state action pair is sent to the current values network, and the next state S' is sent to the target values network. The DQN loss function can be expressed by (14):

$$L(\omega) = E[(r + \gamma \max_{A'} Q(S', A', \omega) - Q(S, A, \omega))^2] \tag{14}$$

where: γ is attenuation coefficient of the reward values; ω is the weight of the loss function; $r + \gamma \max_{A'} Q(S', A', \omega)$ is the values of the target values network; $Q(S, A, \omega)$ is the Q values of the current values network.

It can be seen that the loss function is calculated by the mean square error of the difference between the target values and the current values. After the loss function is derived, we can calculate the gradient of the loss function:

$$\frac{\partial L(\omega)}{\partial \omega} = E[(r + \gamma \max_{A'} Q(S', A', \omega) - Q(S, A, \omega))\frac{\partial Q(S, A, \omega)}{\partial \omega}] \tag{15}$$

Therefore, we use stochastic gradient descent to update parameters to obtain an optimal values $Q(S, A, \omega)$. The current values network in DQN uses the latest parameters, which can be used to evaluate the values function of the current state action pair. But the target values network parameters are a long time ago. After the current values network is iterated, the UCAV takes action on the environment to update the target values network parameters according to (15). In this way, a learning process is completed. The optimal Q values is stored in the network to realize the practical application of the optimal track (Table 1).

Table 1. The process described by algorithm 1

```
Algorithm1 DQN for path planning
Input: $(S, A, R, S')$
Output: $Q(S, A, \omega)$
Initialize replay memory $D$ to capcity $N$ ;
Initialize state-action values function $Q$ with random weights $\omega$;
for episode $= 1, M$ do:
    Initialize the state of UCAV with fixed the coordinate $[x, y, z]$.
    for $t = 1, T$ do:
        With probability $\varepsilon$ select a random the action $A$.
        Otherwise select $A = \arg\max_a Q(S, A, \omega)$.
        UCAV executes action according to (2).
        Calculate the next state according to (4).
        Observe reward according to (13).
        Store transition $(S, A, R, S')$ in $D$.
        Sample random the minibatch from $D$.
        Update the Q-network with the sample according to (15).
    end for.
end for.
```

4 Simulation Experiment and Results

After analyzing the UCAV motion constraints and the path planning algorithm, UCAV and radar confrontation model parameters are set in Table 2. The simulation experiment parameters are shown in Table 3.

Table 2. UCAV and Radar Confrontation Platform parameters

Flight Altitude Of UCAV	0–2.5 (km)
Speed Of UCAV	120–180 (km/h)
Flight Area	500 * 500 (km)
Radar Position	Random

Combined with the UCAV and radar confrontation model parameters in Table 3 and the simulation experiment parameters in Table 3, we use the DQN algorithm to achieve simulation training. Then we normalize the data of the reward values. The normalization function is:

$$f(x) = 1/(1 + e^{-x}) \tag{16}$$

Table 3. Simulation experiment parameters

Network Update Times	200
Episodes Of Training	300
Time Interval	$\Delta t = 0.5\,\mathrm{s}$

On this basis, in order to observe the relation between the reward values and the episodes, we take the reward values of 10 times to do an average. Finally, we can obtain the following simulation results by combining the above parameters and conditions.

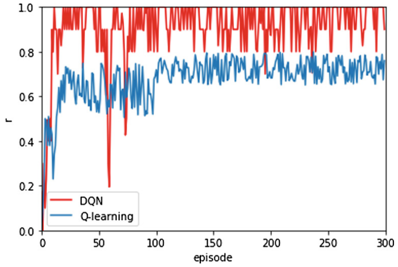


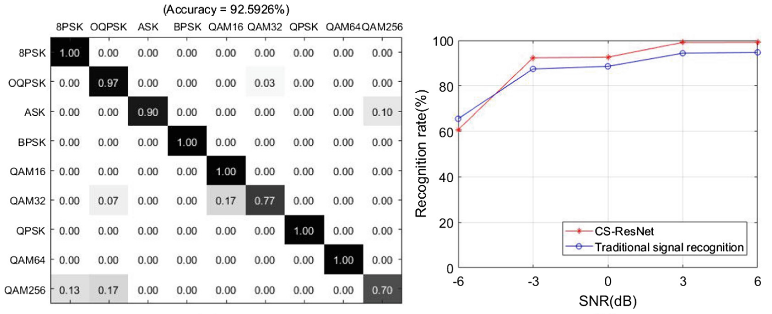
Fig. 6. The reward values of 300 episodes conversion curve.

Figure 6 shows the variation of the overall reward values for each episode of the UCAV. The abscissa represents the number of episode and the ordinate is the reward values after normalization. The reward values 0 and 1 represent the minimum and maximum values after normalization respectively. Figure 6 shows that compared with Q-learning, the DQN overall reward values gradually increases, and eventually stabilizes after the episodes reaches 80, and the average reward values is close to 0.9. This is because the DQN network uses the gradient descent method to correct the loss function, so that the Q values of the intelligent decision network is optimized. When the episode is 60, the reward values is abrupt. This is because after a certain number of trainings, the algorithm will re-randomly search for the optimal solution to avoid falling into local optimum.

When the episodes are 190 and 240, due to the random sampling of the experience replay in the DQN algorithm, the correlation between data is cut off. The simulation shows that the accuracy of the algorithm reaches 90%.

In each modulation category, 100 tests are implemented. It is clear from Fig. 7(a) that the classification accuracy of BPSK, QAM16, QAM64, QPSK and 8PSK is 100%.

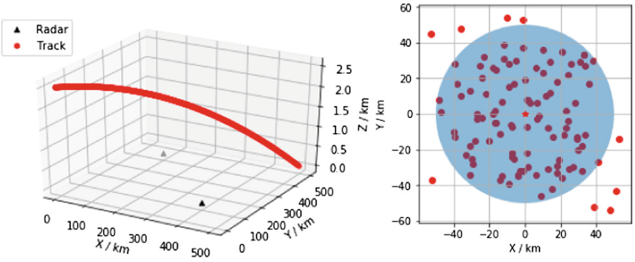
The classification accuracy of OQPSK, ASK, QAM256 and QAM64 becomes worse, especially for QAM32 and QAM256. Figure 7(b) shows the recognition rate curve based on the CS method signal recognition and the recognition rate curve based on the traditional signal recognition. This indicates that the residual neural network signal recognition based on cyclic spectrum has a good recognition rate. The proposed method is lower than the traditional method at -6 dB, but the proposed method is superior to the traditional method with the improvement of signal-to-noise ratio. This is because by transmitting the input information directly to the output, the ResNet network only needs to learn the difference between input and output, which simplifies the learning goal and difficulty, protects the information integrity to a certain extent.



	8PSK	OQPSK	ASK	BPSK	QAM16	QAM32	QPSK	QAM64	QAM256
8PSK	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OQPSK	0.00	0.97	0.00	0.00	0.00	0.03	0.00	0.00	0.00
ASK	0.00	0.00	0.90	0.00	0.00	0.00	0.00	0.00	0.10
BPSK	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00
QAM16	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
QAM32	0.00	0.07	0.00	0.00	0.17	0.77	0.00	0.00	0.00
QPSK	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00
QAM64	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
QAM256	0.13	0.17	0.00	0.00	0.00	0.00	0.00	0.00	0.70

(a) (b)

Fig. 7. (a) Signal recognition confusion matrix under 0 dB condition; (b) The recognition rate of proposed algorithm.



(a) (b)

Fig. 8. (a) The 3D renderings of the UCAV single-shot to the target; (b) The 2D renderings of the UCAV multiple times to the target. (Color figure online)

Figure 8(a) shows the effect of the UCAV reaching the target area. The black point represents the radar and the red curve represents the path of the UCAV. This shows that the UCAV can independently plan a path to the target area without being detected by the radar.

In Fig. 8(b), a light blue circle represents the target area, a red star represents the center ground projection of the target area, and a red projected point represents a of the UCAV approaching the target area. Figure 8(b) shows that after 300 episodes, the success rate of the UCAV reaching the target reaches 90%. This is because the deep reinforcement learning algorithm has good self-learning and correction capabilities. The simulation results show that the UCAV path planning algorithm based on deep reinforcement learning can independently plan a reasonable path in the unknown space.

5 Conclusion

This paper proposes a UCAV path planning algorithm based on deep reinforcement learning, which solves the problem of poor adaptability and slow calculation speed in traditional track planning algorithm. Besides, it realizes the aircraft signal recognition. The simulation results show that the proposed algorithm achieves 90% accuracy under the conditions of compromise planning time and flight quality. Besides, the signal recognition rate reaches 92.59% under 0 dB condition. It has good convergence, which can be applied in the field of modern electronic warfare.

Acknowledgment. This paper is funded by the International Exchange Program of Harbin Engineering University for Innovation-oriented Talents Cultivation, the Fundamental Research Funds for the Central Universities (HEUCFG201832), the Key Laboratory Foundation Project of National Defense Science and Technology Bureau (KY10800180080) and the China Shipbuilding Industry Corporation 722 Research Institute Fund Project (KY10800170051).

References

1. Zou, A.M., Hou, Z.G., Fu, S.Y., Tan, M.: Neural networks for mobile robot navigation: a survey. In: Wang, J., Yi, Z., Zurada, J.M., Lu, B.L., Yin, H. (eds.) Advances in Neural Networks - ISNN 2006. Lecture Notes in Computer Science, vol. 3972, pp. 1218–1226. Springer, Berlin (2006). https://doi.org/10.1007/11760023_177
2. Sun, Y., Ding, M.: Quantum genetic algorithm for mobile robot path planning. In: Fourth International Conference on Genetic and Evolutionary Computing, pp. 206–209 (2010)
3. Wang, H., Duan, J., Wang, M., Zhao, J., Dong, Z.: Research on robot path planning based on fuzzy neural network algorithm. In: IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), pp. 1800–1803 (2018)
4. Peng, J., Sun, X., Zun, F., Zhang, J.: 3-D path planning with multi-constrains. In: IEEE. Chinese Control and Decision Conference, pp. 3301–3305 (2008)
5. Challita, U., Saad, W., Bettstetter, C.: Interference management for cellular-connected UAVs: a deep reinforcement learning approach. IEEE Trans. Wireless Commun. 1–32 (2019)
6. Beomjoon, K., Pineau, J.: Socially adaptive path planning in human environments using inverse reinforcement learning. Int. J. Soc. Robot. **8**(1), 51–66 (2016)
7. Wang, C., Wang, J., Shen, Y.: Autonomous navigation of UAVs in large-scale complex environments: a deep reinforcement learning approach. IEEE Trans. Vehicular Technol. **68**(3), 2124–2136 (2019)
8. Wu, J., Shin, S., Kim, C.: Effective lazy training method for deep q-network in obstacle avoidance and path planning. In: IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 1799–1804 (2017)

9. Çetin, H., Durdu, A.: Path planning of mobile robots with Q-learning. In: 22nd Signal Processing and Communications Applications Conference (SIU), pp. 2162–2165 (2014)
10. Richard, S., Andrew, G.: Reinforcement Learning: An Introduction. MIT press, Cambridge (2018)
11. Lei, T., Ming, L.: A robot exploration strategy based on q-learning network. In: IEEE International Conference on Real-time Computing and Robotics, pp. 57–62 (2016)
12. Mnih, V., et al.: Human-level control through deep reinforcement learning. Nature **518**(7540), 529–533 (2015)
13. Mnih, V., Kavukcuoglu, K., Silver, D., et al: Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602, pp. 1–9 (2013)