

PAPER • OPEN ACCESS

An Automatic Parking Model Based on Deep Reinforcement Learning

To cite this article: Li Junzuo and Long Qiang 2021 *J. Phys.: Conf. Ser.* **1883** 012111

View the [article online](#) for updates and enhancements.



IOP | ebooks™

Bringing together innovative digital publishing with leading authors from the global scientific community.

Start exploring the collection—download the first chapter of every title for free.

An Automatic Parking Model Based on Deep Reinforcement Learning

Li Junzuo^{1, a}, Long Qiang^{1, b*}

¹School of Science, Southwest University of Science and Technology, Mianyang, SiChuan, 621000, China

^aemail: junzuoli@163.com

^{*}Corresponding author's e-mail: longqiang@swust.edu.cn

Abstract. When parking a car, it is crucial to ensure the car constantly approaches the parking point, gets an excellent heading angle, and avoids significant losses caused by line pressure. An automatic parking model based on deep reinforcement learning is proposed. A parking kinematics model is built to calculate the different states of its movement. Steering angle and displacement are used to achieve interaction as actions; A comprehensive reward function is designed to consider the focus of action and safety in different parking stages. Through training, the car's automatic parking is realized, and a comprehensive analysis of the various stages and situations in the parking process is given. Besides, it is showed by a further generalization experiment that the model has good generalization.

1. Introduction

Car parking strategy planning is the key technology to realize automatic parking. In response to this problem, a large number of scholars have conducted research. Mainly can be divided into fuzzy logic-based controller [1] and fuzzy controller-based parking algorithm [2], the improvement of the controller based on the approximate iterative strategy of the fractional-order controller parameter self-tuning method [3], the path-based Tracking, graph search techniques [4-5] and neural network-based methods [6]. Due to the heuristic rules in the graph search algorithm, the large amount of high-quality data required by fuzzy logic and neural networks cannot guarantee the optimal parking effect, thus limiting the improvement of automated parking technology. In recent years, with the rapid development of reinforcement learning in various fields, its method of self-learning according to rules makes it possible to ensure that the agent completes more complex and efficient operations without the influence of subjective human experience. Scholars have studied parking motion planning methods based on the Monte Carlo tree search algorithm [7]. However, this method may cause many excellent actions in the complete set of steps to be unsampled and miss a good choice of activities. There are also scholars based on the EKF path tracking model [8], using the DDPG algorithm to study the automatic parking strategy, and this method has higher requirements for the car's path tracking sensor. Simultaneously, some scholars realize the parking process by learning the reward function based on the complex trajectory and the guidance function of Coulomb's law [9]. This paper proposes an automatic parking model based on the car's parking kinematics model and deep reinforcement learning. Compared with the Monte Carlo tree search algorithm and the EKF path tracing model, this model avoids unsampling of good actions and reduces the requirements for automotive sensors. Besides, compared with the complex trajectory



function and guiding function, this paper proposes a simple and effective reward function, which considers the car's heading angle, the distance of the parking point, and the line pressure penalty.

2. Establishment of parking environment and car kinematics model

2.1 Parking environment

The parking environment simulated in this paper is a vertical parking environment, and there are two obstacles around the parking position. Regarding the parking space's center position as the origin, the parking environment coordinate system's establishment is shown in Figure 1.

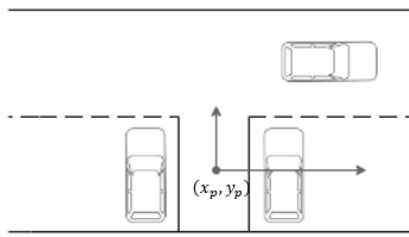


Figure 1. Vertical parking environment.

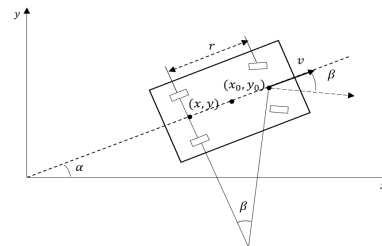


Figure 2. Vertical parking environment.

In the Figure 1, (x_p, y_p) is the parking point coordinates in the parking environment.

2.2 Car kinematics model

Since the car is in a low-speed motion process during parking, the car's speed is controlled within 10 km/h. Therefore, this article regards the car in the low-speed motion state as a rigid body and abstracts the car shape as a rectangle. At this time, the schematic diagram of the car's movement in the parking process is shown in Figure 2. Where, (x, y) are the coordinates of the center point of the rear axle of the car, (x_0, y_0) is the coordinates of the center point of the car; α is the car heading angle formed by the vehicle and the x-axis; β is the steering angle of the car; r is the wheelbase of the vehicle.

In the case of low-speed motion, it can be considered that the wheels of the car only roll, and there is no slip, and the influence of the lateral dynamics of the tire can be ignored. Under the above conditions, a car kinematics model based on the rear wheel as the driving wheel is established [10]:

$$\begin{cases} \dot{x} = v \cdot \cos \alpha \cdot \cos \beta \\ \dot{y} = v \cdot \sin \alpha \cdot \cos \beta \\ \dot{\alpha} = \frac{v \sin \beta}{r} \end{cases} \quad (1)$$

Where v is the vehicle speed. Therefore, in a given time, according to formula (1), the calculation formula for the amount of vehicle state change can be obtained as:

$$\begin{cases} \Delta x = \dot{x} \cdot \Delta t = \Delta s \cdot \cos \alpha \cdot \cos \beta \\ \Delta y = \dot{y} \cdot \Delta t = \Delta s \cdot \sin \alpha \cdot \cos \beta \\ \Delta \alpha = \dot{\alpha} \cdot \Delta t = \frac{\Delta s \cdot \sin \beta}{r} \\ \Delta s = v \cdot \Delta t \end{cases} \quad (2)$$

Where Δs is the vehicle displacement under the steering angle β , the negative value is the reverse displacement, and the positive value is the forward displacement. To ensure the car's actions are in line with reality and to prevent the agent's exploration range from being too large, the steering angle β of the

car is restricted to $-30^\circ \sim 30^\circ$, and the displacement Δs is restricted to $-1 \sim 0.2$ (m). Therefore, after inputting the actions to the vehicle, the heading angle α' and the coordinate of the rear axle center point of the vehicle (x', y') after moving can be obtained by formula (3):

$$\begin{cases} x' = x + \Delta x \\ y' = y + \Delta y \\ \alpha' = \alpha + \Delta \alpha \end{cases} \quad (3)$$

3. Establishment of automatic parking model

Because the car's state of motion in the parking process is continuous, aiming at such state characteristics, this paper adopts the deep deterministic policy gradient algorithm (DDPG) that learns more stable on the continuous action space [11]. Its composition framework is as shown in Figure 3.

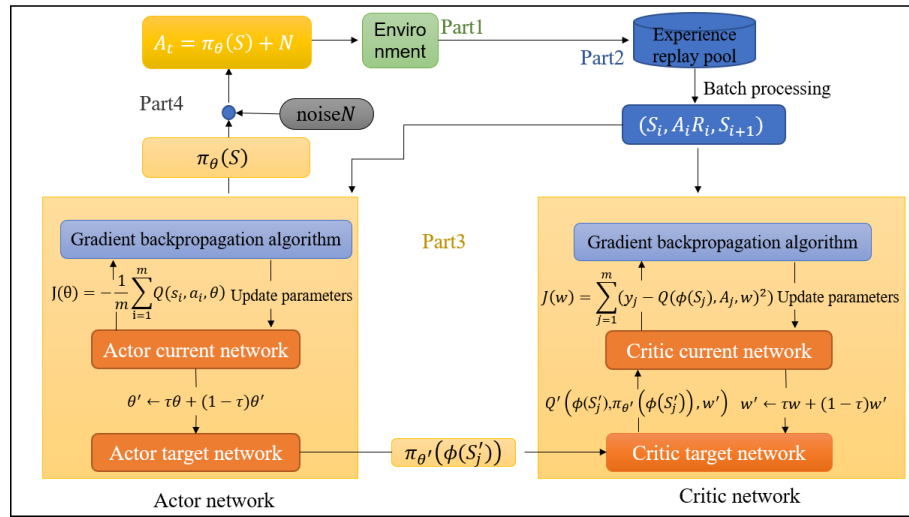


Figure 3. DDPG algorithm framework.

It comprises four neural networks: Actor current network, Actor target network, Critic current network, and Critic target network. The functions' detail in [11].

3.1 Action and state

According to the iterative formula of Eqs. (2) (3), we take the two-tuple $(\beta, \Delta s)$ of the car's steering angle β and the car's displacement Δs under β as the input action of the DDPG. The triplet (x_0, y_0, α) formed by the car's center point coordinates (x_0, y_0) and the car's heading angle α in the coordinate system shown in Figure 1 is used as the output state of the DDPG.

3.2 Reward function

To ensure that the car is constantly approaching the parking center during parking, and the heading angle α of the car when it reaches the parking point is close to 90° . In addition, it is also required that there will be no car collision (ie, line pressure) during parking. In this paper, a simple and effective reward function calculation formula is obtained through many experiments:

$$R = 10 \cdot [(1-w)R_d + wR_\alpha] + R_p \quad (4)$$

Where, R_d is the distance reward; R_α is the heading angle reward value; R_p is the line pressure penalty value; w is the weight of the distance index and the heading angle index, and the value is $[0,1]$.

It is adjusted according to the distance from the parking point during the parking process to adapt to the main movement tendency at different stages.

3.2.1 Distance reward

The distance index is an important index to ensure that the car can keep close to the parking point through learning. R_d is the normalized distance index with a value range of $[0,1]$. $R_d = 0$ indicate the car has reached the parking point. $R_d = -1$ indicates the distance between the car and the parking point is the largest.

$$R_d = -\frac{\sqrt{(x_0 - x_p)^2 + (y_0 - y_p)^2}}{d_{\max}} \quad (5)$$

3.2.2 Course angle reward

R_α ensures the standardization of car parking during the parking process, and the value range is $[0,1]$. $R_\alpha = 0$ indicates the car is in a vertical parking state at the parking point. $R_\alpha = -1$ means the car is in the reverse vertical parking state at the parking point.

$$R_\alpha = \begin{cases} -|\alpha - \pi/2| / \pi & \alpha \in [0, \pi] \\ -|\alpha - 3\pi/2| / \pi - 1/2 & \alpha \in [\pi, 2\pi] \end{cases} \quad (6)$$

3.2.3 Line pressure penalty

We set a penalty value to give a more significant penalty when car pressure the line:

$$R_p = -10 \quad (7)$$

3.3 Parking conditions

To ensure that when the car is automatically parking, it can stop when it reaches the parking point, the parking conditions set in this paper are:

$$\begin{cases} d = \sqrt{(x_0 - x_p)^2 + (y_0 - y_p)^2} \leq 0.1m \\ \alpha \leq 10^\circ \end{cases} \quad (8)$$

Where, d is the distance between the car and the parking spot.

3.4 Neural Network

For the Actor-network responsible for interacting with the environment and the sampling and updating of the experience playback pool and the Critic network responsible for updating the value function.

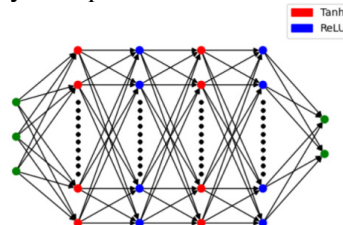


Figure 4. Actor network structure

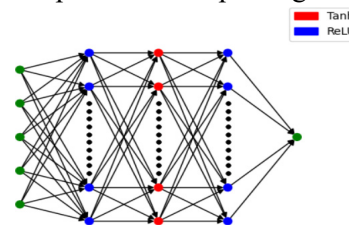


Figure 5. Critic network structure

As shown in Figure 4, the Actor-network structure uses four fully connected layers in this article. The fully connected layer 1 has 30 neurons, and the fully connected layer 3 has 20 neurons. Both use the Tanh activation function to ensure that the network model does not fluctuate too much during the gradient descent calculation [9]. The fully connected layer 2 has 45 neurons, and the fully connected layer 4 has ten neurons, both of which are ReLU activation functions that make the neurons have sparse activation [9]. As shown in the Critic network structure in Figure 5, this article uses a three-layer, fully

connected layer. Fully connected layer 1 has 30 neurons; fully connected layer 3 has 25 neurons, all using ReLu activation function; fully connected layer 2 has 45 neurons, using Tah activation function.

4. Model training and experimental analysis

4.1 DDPG algorithm parameter setting of automatic parking model

Table 1. Model algorithm parameters.

Parameters	Values	Parameters	Values
Actor network learning rate	0.002	Critic network learning rate	0.002
Reward decay factor	0.92	Sample pool capacity	100000
Batch training size	140	Soft update parameters	0.01
Maximum simulation step per round	200	Number of training rounds	2000
Vehicle heading angle action noise	0.99	Displacement noise of car	0.95

4.2 Automatic parking model training

This paper combines the deep reinforcement learning algorithm DDPG with the car parking process's kinematics model to train the automatic parking model proposed. The overall training framework is:

Algorithm 1. Training framework

Step1: Initialize state $S(x, y, \alpha)$, reward in each round $ep_r = 0$
 Step2: Determine whether to complete an iteration. If yes, go to Step3 instead ending.
 Step3: Use the DDPG algorithm to determine the action A that the car should take in the current state S
 Step4: Increase motion noise, and restrain α , Δs
 Step5: A enter the parking environment, get the next state $S'(x', y', \alpha')$ and reward R
 Step6: Record the data and store S' in the experience replay pool
 Step7: Judge whether the experience return visit pool is full. If yes, go to Step8 instead Step 9
 Step8: Gradually reduce noise, use DDPG algorithm to learn
 Step9: $S = S'$, $ep_r += R$, then stop.

4.3 Analysis of training results

The excellent parking actions and states in the experience replay pool are sampled with 50 rounds as the step size, and the overall convergence of the reward value in 2000 different rounds in the entire DDPG algorithm training process is as follows:

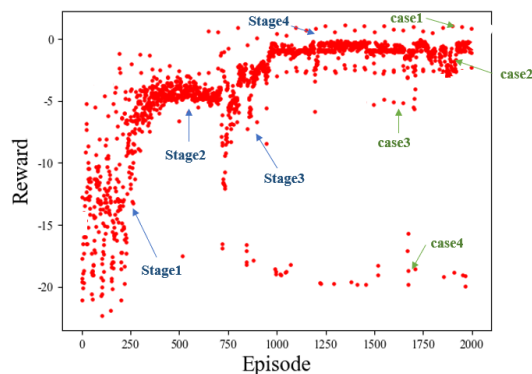


Figure 6. The overall convergence of the reward value

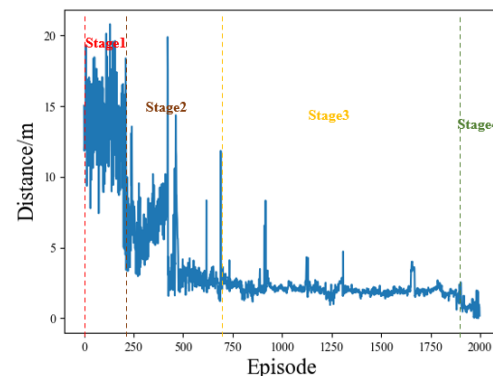


Figure 7. Change in distance within a certain 2000 rounds

As shown in Figure 6 above, the car parking process's reward value gradually converges from -20 to near 0, indicating that the algorithm converges. During the parking process, four types of situations appeared, namely:

Table 2. Situations and explanations

Situations	Explanations
Case1	Realization of the expected parking the effect;
Case2	Car moves to the parking point, and its angle does not meet expectations. Therefore, it adjusts the angle and distance of the car so that the reward value will fluctuate;
Cse3	A certain distance between the car and the parking point exists, reward value is mainly determined by the distance between the car and the parking point;
Case4	Car presses the line when it is in motion to be given a more significant penalty.

The above results show that the car continuously learns the strategy of implementing case1 in the process of realizing parking to realize automatic parking.

In Figure 7, the change of the distance between the car and the parking center of a training round whose initial position is far from the parking point is demonstrated. As shown in Figure 7 above, the four different stages in the parking process are:

Table 3. Stages and explanations

Stages	Explanations
Stage1	The car is constantly exploring the surrounding environment to achieve the goal of being close to the parking point;
Stage2	The car acts close to the parking point after the car has completed the exploration. When approaching the parking point, the car also conducts environmental exploration in a smaller range.
Stage3	The re-exploration stage of the car. In which, the car is closer to the parking point, so the heading angle is mainly considered;
Stage4	The car arrives near the parking center and is constantly adjusted in order to get a better car angle and finally meet the parking conditions;

5. Generalization experiment

Modify the starting position of the car and conduct four sets of generalization experiments. The modified starting position state is:

Table 4. Generalization experiment group with different starting states.

Groups	Positions	Headings
a	(4,6)	0
b	(4,6)	80°
c	(-4,4)	0
d	(-4,4)	-80°

Implement the automatic parking model established in this paper. The car's distance change during the parking process is shown in Figure 8 under each group's initial position states shown in Table 4.

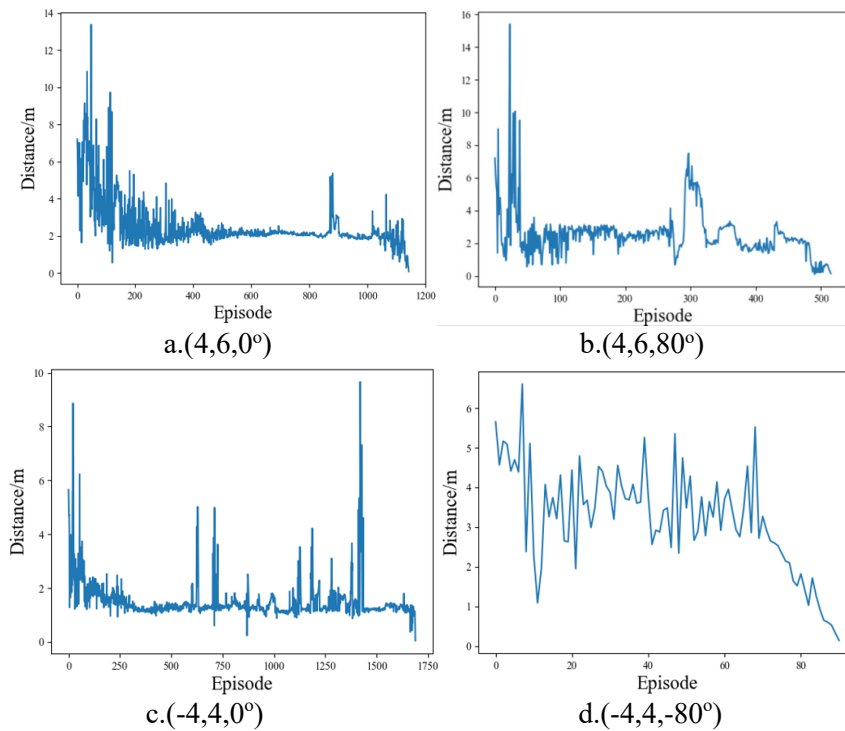
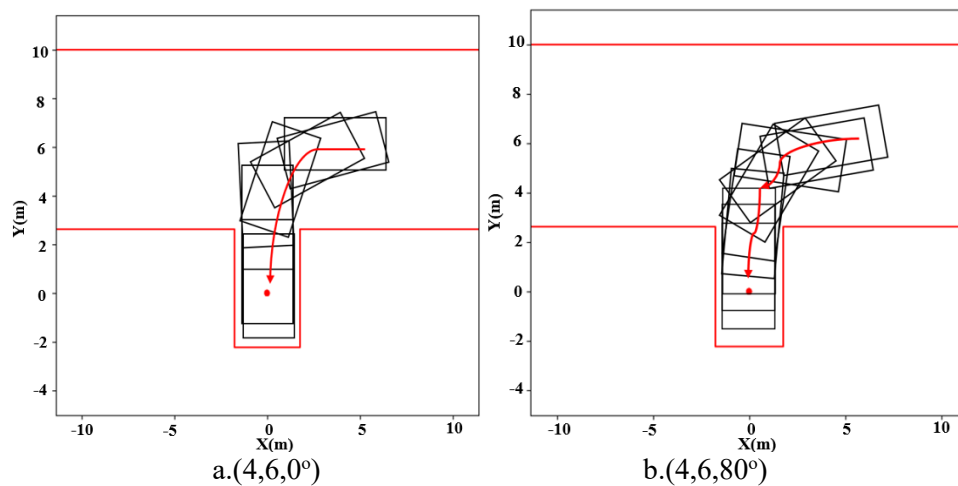


Figure 8. Changes in car parking distance of 4 generalization experiments

As shown in Figure 8, the car can complete automatic parking at different initial positions at different times. Therefore, the model proposed in this paper has a specific generalization and applicability. Besides, the car presents an exploratory parking process during the parking process. That is, the car will first complete the action of exploring the surrounding environment at a particular position and then proceed to the action of approaching the parking point. The trend characteristics of the car movement process under different generalization experiments are shown in Figure 9, the rectangle is the car, and the solid line with the arrow is the car movement trend:



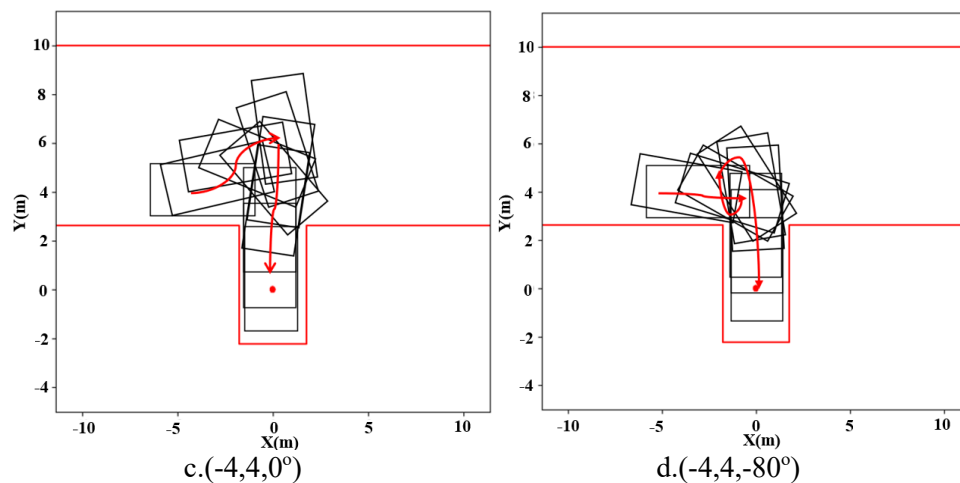


Figure 9. Trend characteristics of car movement in four generalization experiments

6. Conclusions

First, this paper establishes a kinematics model as a simulation environment based on the car's low-speed motion characteristics in the parking process. It builds an automatic parking model based on deep reinforcement learning, which reduces the requirements for sensors for automatic parking technology and increases the application cost low. Besides, based on the deep reinforcement learning algorithm DDPG, we used the neural network to generate learning data. Then, a simple and effective reward function is constructed to ensure that the car is constantly approaching the parking center and get the optimal car angle during the parking process. Furthermore, we penalize the line pressing action to ensure the convergence of the strategy and the parking process's safety and the excellent parking data Learning to avoid unsampling of data.

Acknowledgments

This research was supported by the Southwest University of Science and Technology College of Science Talent Cultivation and Subject Competition Innovation Fund Project (LX2020013).

References

- [1] Guo K, Jiang H, Zhang J, et al. Automatic parallel parking steering controller based on fuzzy logic[C]//Proceedings of the Journal of Jilin University (Engineering and Technology Edition). 2009.
- [2] Ryu Y W, Oh S Y, Kim S Y. Robust automatic parking without odometry using enhanced fuzzy logic controller[C]//2006 IEEE International Conference on Fuzzy Systems. IEEE, 2006: 521-527.
- [3] Jinze, S. (2009) Research on Key Technology for Auto-parking system. National University of Defense Technology.
- [4] Müller B, Deutscher J. Orbital tracking control for car parking via control of the clock using a nonlinear reduced order steering-angle observer[C]//2007 European Control Conference (ECC). IEEE, 2007: 1917-1924.
- [5] ZHOU H, WANG W. Core algorithm research in automatic parking system[J]. Journal of Circuits and Systems, 2012, 3.
- [6] Li T H S, Yeh Y C, Wu J D, et al. Multifunctional intelligent autonomous parking controllers for carlike mobile robots[J]. IEEE transactions on industrial electronics, 2009, 57(5): 1687-1700.
- [7] Zhang J, Chen H, Song S, et al. Reinforcement Learning-Based Motion Planning for Automatic Parking System[J]. IEEE Access, 2020, 8: 154485-154501.
- [8] Zhang P, Xiong L, Yu Z, et al. Reinforcement learning-based end-to-end parking for automatic parking system[J]. Sensors, 2019, 19(18): 3996.

- [9] He, H., Weifeng, G., Weiwei, M., Run, Z., Jin, C., Bingli, Z. (2020) Automatic Parking Based on Deep Reinforcement Learning Control Strategy Research. China Society of Automotive Engineers. 2020:9.
- [10] Hailong, Z., Wei, C., Panpan Y. (2019) Research on parking track tracking of automatic parking system. Automotive Practical Technology.
- [11] Lillicrap T P, Hunt J J, Pritzel A, et al. Continuous control with deep reinforcement learning[J]. arXiv preprint arXiv:1509.02971, 2015.