UAV visual flight control method based on deep reinforcement learning

Shuangxia Bai
School of Electronics and Information, Northwestern Polytechnical University
Xi'an, China
nwpu18734760639@163.com

Bo Li
School of Electronics and Information, Northwestern Polytechnical University
Xi'an, China
libo803@nwpu.edu.cn

Zhigang Gan
School of Electronics and Information, Northwestern Polytechnical University
Xi'an, China
ganzhigang@mail.nwpu.edu.cn

Daqing Chen
School of Engineering, London South Bank University
London, UK
chend@lsbu.ac.uk

***Abstract*—Aiming at an intelligent perception and obstacle avoidance of UAV in an environment, a UAV visual flight control method based on deep reinforcement learning is proposed in this paper. The method employs Gate Recurrent Unit (GRU) to the UAV flight control decision network, and uses Deep Deterministic Policy Gradient (DDPG), a deep reinforcement learning algorithm to train the network. The special gates structure of GRU is utilized to memorize historical information, and acquire the variation law of the environment of UAV from the time series data including image information of obstacles, UAV position and speed information to realize a dynamic perception of obstacles. Moreover, the basic framework and training method of the network are introduced, and the generalization ability of the network is tested. The experimental results show that the proposed method has better generalization ability and better adaptability to the environment.**

Keywords—GRU, DDPG, Flight Control, generalization

I. INTRODUCTION

As a machine leaning paradigm, deep reinforcement learning has found many applications due to its unique mechanism to enable an agent to automatically determine its most meaningful behavior within a certain context, in order to maximum its performance. In the area of Unmanned Aerial Vehicle (UAV) flight control, the essential problem to tackle is to enable UAV to deal with unfamiliar environments, and to make real-time, dynamic and autonomous decisions on their ideal behavior according to their ever-changing environments. The problem facing UAV flight fits the reinforcement learning framework well, and therefore deserves further research. This work presents such a research effort.

At present, methods for UAV flight control based on deep reinforcement learning usually rely on a UAV's flight parameters of a single moment [1-3], and as a consequence, lack of temporal and 3D space information analysis in the process of UAV flight. Actually, the flight process of UAV has strong temporal dependence, so mining sequential feature of environment changes in the flight process of UAV is important to control the flight of UAV.

This paper proposes a UAV intelligent flight control method which combines image information of obstacles and UAV state information as input, and continuously generates UAV linear velocity as outputs to control the movement of UAV to realize dynamic obstacle avoidance and flight. The image information refers to the gray value of the grayscale image of the obstacle in front of UAV. In the image processing, Gate Recurrent Unit (GRU) is added to control the input and memory information to make a prediction for the UAV linear velocity in the current time step. Therefore, based on GRU, an UAV flight control decision network is designed in this paper. The network is trained by Deep Deterministic Policy Gradient algorithm (DDPG). By using four consecutive images as input, mining the sequential variation features. This network is superior to tradition CNN in terms of sequential features extraction. The effectiveness of the proposed model for UAV flight control and obstacle avoidance is verified by experiments.

The remainder of this paper is organised as follows. In Section II, this paper defines a UAV visual flight control method based on DDPG, and details the structure of UAV flight control decision network and reward function. In Section III, this paper trains the UAV flight control decision network and completes the test of the generalization ability of the network. Finally, conclusions are presented in Section Ⅳ.

II. UAV VISUAL FLIGHT CONTROL METHOD BASED ON DEEP REINFORCEMENT LEARNING

A. Task Specification

This paper is concerned with flight control for a UAV to fly to a designated destination through obstacle avoidance flight, and a UAV visual flight control method is proposed based on Deep Reinforcement Learning. The flight decision making is based on the image information of obstacles and the UAV state information. The distance change between the UAV and the obstacle is perceived through monitoring the change of continuous frame images of obstacle with time. The state information (of the UAV's position and speed) can reflect the relative position relationship between the UAV and the target to determine the direction of the UAV's movement. Accordingly, the UAV can make an obstacle avoidance action to realize the vision-based autonomous obstacle avoidance of UAV.

In this paper, the agent outputs the linear velocity of the UAV as a command. After receiving the command, the UAV executes the action, obtains the corresponding reward in the environment, and updates its state. The interaction process between the agent and environment is shown in Figure 1. The task to be achieved by the UAV in this paper is as follows:

Set the target, the UAV judges the obstacle information and the relative position between the UAV and the target

based on the image information, its own position and speed information, makes decision to bypass the obstacle and reach the designated target.

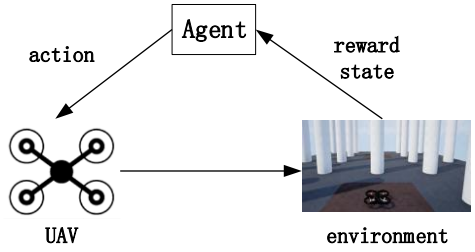

Figure 1. Agent interaction process.

B. Related Theory

1) Deep Deterministic Policy Gradient Algorithm

Deep Deterministic Policy Gradient algorithm (DDPG)[4]is an actor-critic, model-free algorithm based on deterministic policy gradient that can operate over continuous action spaces. DDPG adopts network simulation policy function and Q function, and introduces replay memory to update network parameters. The training process of DDPG is shown in Figure 2.

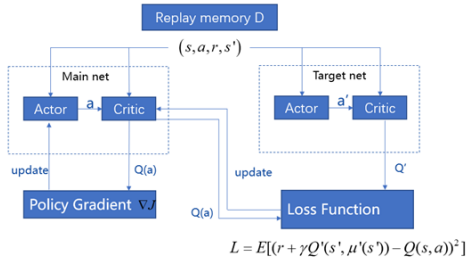

Figure 2. The training process of DDPG.

DDPG algorithm creates a copy of the actor and critic networks, $Q'\left(s,a\middle|\theta^{Q'}\right)$ and $\mu'\left(s,a\middle|\theta^{\mu'}\right)$, respectively, and they are further used for calculating the target values. The weights of the target networks are then updated by having them slowly track the learned network: $\theta' \leftarrow \tau\theta+(1-\tau)\theta'$ with $\tau \ll 1$. That means that the target values are constrained to change slowly, in order to greatly improve the stability of learning.

The actor is updated by applying the chain rule to the expected return from the start distribution J with respect to the actor parameters to have higher reward:

$$\begin{aligned}\nabla_{\theta^\mu} J &\approx \mathrm{E}_{s_t\sim\rho^\beta}[\nabla_{\theta^\mu} Q(s,a\middle|\theta^\mu\Big|_{s=s_t,a=\mu(s_t|\theta^\mu)}] \\ &= \mathrm{E}_{s_t\sim\rho^\beta}[\nabla_a Q(s,a\middle|\theta^\mu\Big|_{s=s_t,a=\mu(s_t)}\nabla_{\theta^\mu}\mu(s_t\middle|\theta^\mu)\Big|_{s=s_t}]\end{aligned} \tag{1}$$

The loss of policy network can be approximated as:

$$L\left(\theta^\mu\right) = -\frac{1}{N}\sum_i Q\left(s_i,\mu\left(s_i\right)\middle|\theta^Q\right) \tag{2}$$

Optimize the Q network by minimizing the loss:

$$L(\theta^Q) = \frac{1}{\mathrm{N}}\sum_i[(y_i - Q(s_i,a_i\middle|\theta^Q))^2] \tag{3}$$

where $$y_i = r_i + \gamma Q'\left(s_{i+1},\mu'(s_{i+1})\middle|\theta^{Q'}\right) \tag{4}$$

where y_t is also dependent on θ^Q, this is typically ignored.

2) Gate Recurrent Unit

In terms of the network structure, the Recurrent Neural Network (RNN) is the same as the traditional neural network (CNN), except the hidden layer neurons in the RNN are interconnected. As such RNN can memorize the previous information and use this information to influence the output of the following successive nodes in the network.

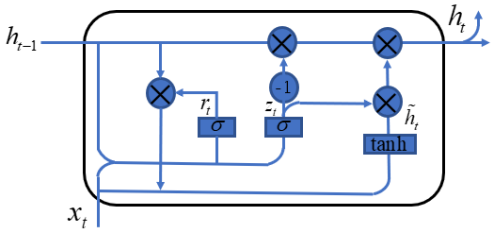

Figure 3. Structure of GRU.

Gate Recurrent Unit (GRU)[5] is a type of RNN, and it has also been proposed to solve the problems of long-term dependencies in RNN, the same as Long-Short Term Memory (LSTM)[6]. The principle of GRU is that the gating mechanism is used to control input, memory and other information to make predictions at the current time step, so that the information can selectively affect the state of the current time in the RNN. The structure of GRU is shown in Figure 3.

The GRU can be expressed as follows:

$$\begin{aligned} z_t &= \sigma\left(W_z \bullet \left[h_{t-1}, x_t\right]\right) \\ r_t &= \sigma\left(W_r \bullet \left[h_{t-1}, x_t\right]\right) \\ \tilde{h}_t &= \tanh\left(W \bullet \left[r_t, h_{t-1}, x_t\right]\right) \\ h_t &= \left(1-z_t\right) * h_{t-1} + z_t * \tilde{h}_t \end{aligned} \tag{5}$$

where x_t is current input, h_{t-1} is last output, h_t is current output.

GRU has two gates, namely reset gate and update gate. The reset gate determines the fusion of input information and memory information. The update gate controls the amount of data that memory information is saved to the current time step.

C. Data Processing

We obtain data from the simulation system that simulates the real flight process of the UAV. During the process of flight maneuvering, all the data about the image, speed and position are generated with successive time tags. These data describe the state and trend of the UAV at a certain time and were collected to form the historical dataset. In this research, the data to be obtained includes image information from the front camera of the UAV and the position and velocity of the UAV. Each image needs to be processed into a gray-scaled image, and the grayscale values of four consecutive frames are stacked into a tensor of size (1, 4, 72, 128) as input. The speed and position information are tensors of size (1, 3). The value ranges of data collected for different attributes varies significantly, as such the data has been normalized by the max-min normalization method with an interval [0,1].

D. Network Structure

The network structure for the UAV visual flight control is shown in Figure 4, where, the image information of obstacles, UAV position and speed information are inputted into the

Batch Normalization layer for processing, and this layer is used to make various information have the same distribution. Therefore, adding Batch Normalization layer to the network can speed up the network training and convergence speed.

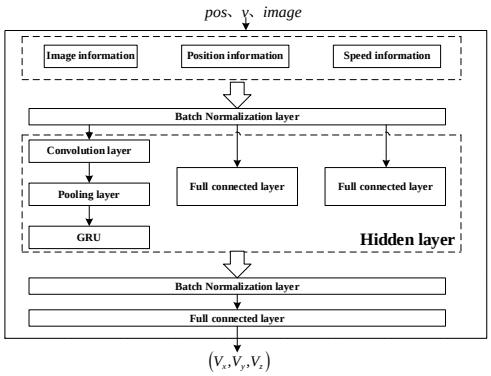

Figure 4. The network structure of UAV visual flight control method

The image information of obstacle is sequentially inputted to the convolutional layer, pooling layer and GRU. The convolutional layer conducts in-depth analysis of each small block in the neural network to obtain more abstract features. The pooling layer is used to reduce the dimensionality of features, quickly reduce the size of the matrix of input, and reduce the number of network parameters, and therefore speed up the calculation process and prevent overfitting. Moreover, the speed and position information of UAV is inputted into full connected layers for processing. Further, the data is integrated together and inputted into a full connected layer to output a tensor of size (1, 3). Finally, the flight control command, the linear velocity of UAV is obtained. The activation function in the network is set as follows:

The Elu function is selected as the activation function for calculating the state value of GRU and the output value of convolutional layer in the hidden layer. And the output of the remaining layers is activated by the tanh function.

E. Reward Function

During the training process, each action taken by the UAV needs to be scored by the reward function. The flight process of UAV is divided into three stages

(1) If UAV flies too far from the existing environment or a collision occurs, the UAV would be considered having crashed. The reward value is then is given as:

$$r = -2 \tag{6}$$

(2) If the UAV reaches near target, the reward value is:

$$r = 2 \tag{7}$$

(3) The reward function during UAV flight is defined as follows:

- A positive reward is given if the UAV is closer to a target than it was at the previous time point, and a negative reward is given if it is not. The distance-based reward is defined as:

$$r_1 = \sqrt{(x_0 - x_{aim})^2 + (y_0 - y_{aim})^2 + (z_0 - z_{aim})^2} - \sqrt{(x - x_{aim})^2 + (y - y_{aim})^2 + (z - z_{aim})^2} \tag{8}$$

where (x_0, y_0, z_0) is the last position of the UAV, (x, y, z) is the current position of the UAV, $(x_{aim}, y_{aim}, z_{aim})$ is the position of the target.

- The UAV should always be flying towards a target. The UAV's flight direction reward is defined as:

$$\mathbf{s} = (x_{aim}, y_{aim}, z_{aim}) - (x, y, z) \tag{9}$$

$$\mathbf{v} = (v_x, v_y, v_z) \tag{10}$$

$$\cos\theta = \frac{\mathbf{s} \bullet \mathbf{v}}{|\mathbf{s}| \bullet |\mathbf{v}|} \tag{11}$$

$$r_2 = 2 * \cos\theta \tag{12}$$

The reward function during UAV flight:

$$r = \alpha r_1 + \beta r_2 \tag{13}$$

where α and β are weight coefficients, and they are adjusted according to the influence of various factors on the control effect in the experiment. And we set $\alpha = 0.6$ and $\beta = 0.4$.

III. EXPERIMENT AND ANALYSIS

The AirSim simulation platform was used for the experiments in this study. AirSim is a high-fidelity simulation platform with realistic visions, and it contains many modules to use to simulate the real environment, such as weather conditions, gravity, etc., A three-dimensional space environment has been considered.

A. Experiment Settings.

The details of the setting to the experiment parameters are as follows:

(1) Algorithm uses Adam for learning the UAV flight control decision network parameters with a learning rate of 10^{-4} and 10^{-3} for the actor and critic respectively.

(2) For *Q*, it includes L_2 weight decay of 10^{-2} and uses a discount factor of $\gamma = 0.99$.

(3) For the soft target updates, this paper uses $\tau = 0.001$. The final output layer of the actor was a tanh layer, to bound the actions. The final layer weights and biases of both the actor and critic are initialized from a uniform distribution $\left[-3\times10^{-3}, 3\times10^{-3}\right]$ and $\left[-3\times10^{-4}, 3\times10^{-4}\right]$ for the low dimensional and pixel cases respectively. This is to ensure the initial outputs for the policy and value estimates are near zero. The other layers are initialized from uniform distributions $\left[-\frac{1}{\sqrt{f}}, \frac{1}{\sqrt{f}}\right]$, where *f* is the fan-in of the layer.

(4) We complete the construction and trains the UAV flight control decision network based on Torch module.

B. Results of network training and test

The changes in the loss of the policy network (UAV flight control decision network) training process are shown in Figure 5. It can be seen that the obstacle avoidance flight result has achieved the expected value after the network trained for 10,000 iterations.

We choose a set of parameters of flight control decision network completing training to test. During the test, the UAV can successfully identify obstacles and avoid them, and

finally reach the designated destination. The flight trajectory of the UAV is shown in Figure 6.

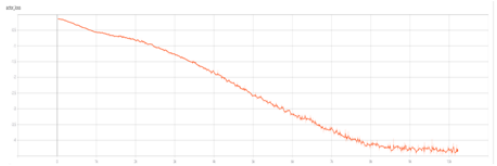

Figure 5. The loss function during training.

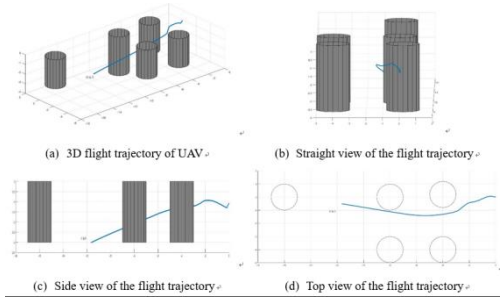

Figure 6. The flight trajectory of UAV

C. *Generalization Ability Test*

The generalization ability of the deep reinforcement learning model used has been tested upon: (1) Verify the validity of image information in the decision-making process. Change the initial flight position and speed of the UAV in the environment, and place a column obstacle ahead the UAV at the same position for testing; And (2) Test the adaptability of the model to different types of image information including the distribution of gray value in image information. Change the shape of the obstacle.

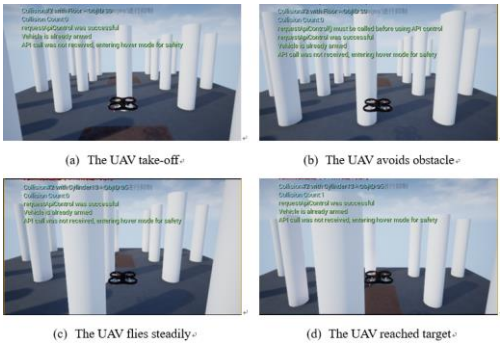

Figure 7. 3D scene of UAV flight after changing the starting point.

As shown in Figure 7, when the obstacle is far away, the UAV flies forward and approaches the specified target. When reaching the obstacle, the drone flies forward to the right to avoid the obstacle. After that, the UAV flies steadily and reaches the target. After changing the starting point to modify the position and speed information in the state information, the UAV can still bypass obstacles and reach the target. This indicates the effectiveness of the image information of obstacles in the decision-making process.

As shown in Figure 8, when the obstacle is far away, the UAV flies forward and approaches the specified target. When reaching the obstacle, the UAV keeps changing its direction, bypassing the obstacle along the surface of the spherical obstacle. After that, the UAV flies steadily and reaches the target. It can be concluded that even though the distribution of gray values in the image data is different, the network can still output the correct action to avoid obstacle.

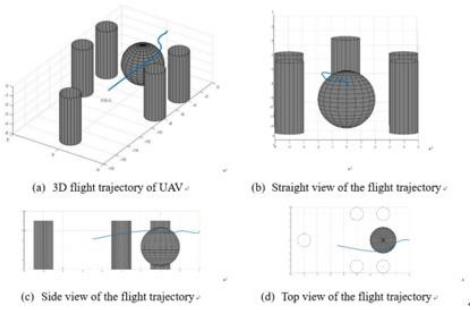

Figure 8. The flight trajectory of UAV after changing obstacles

The model generalization ability test results show that the UAV visual flight control method based on Deep Reinforcement Learning has a strong generalization ability, which can identify unknown obstacles and make decisions according to the distance changes between the UAV and the obstacles, and successfully avoid obstacles.

IV. Conclusion

Aiming at the intelligent perception and obstacle avoidance of UAV for the environment, this paper constructs obstacle-avoidance flight decision network by using the powerful ability of processing in time series data processing of GRU. By training a large amount of image data, the network can directly extract the internal relationship between image information and maneuvering decision variables. The experimental results show that the network is accurate in obstacle-avoidance flight decision.

Finally, the generalization ability test of the model is carried out. The results show that the autonomous obstacle-avoidance method of UAV based on image information has good scalability and improved adaptability to the environment. It provides a solution for autonomous flight of UAV in unfamiliar environment.

ACKNOWLEDGEMENTS

The authors would like to acknowledge National Natural Science Foundation of China (Grant No. 61573285, No.62003267), Open Fund of Key Laboratory of Data Link Technology of China Electronics Technology Group Corporation (Grant No. CLDL-20182101) and Natural Science Foundation of Shaanxi Province (Grant No. 2020JQ220) to provide fund for conducting experiments.

References

[1] Xue X. Indoor UAV Obstacle Avoidance Based on Deep Reinforcement Learning [D]. Harbin Institute of Technology, 2020.

[2] Xu G, Zong X, Yu G Su H. Research on Intelligent Obstacle Avoidance Method for Unmanned Vehicles Based on DDPG[J]. Automotive Engineering, 2019, 41(02): 206-212.

[3] Arnab M, Leonhard H C, Florian H. Time-Varying Parameter Model Reference Adaptive Control and Its Application to Aircraft[J]. European Journal of Control,2019.

[4] Lillicrap, T P, Hunt, J J, Pritzel A, Heess N, Erez T, & Tassa Y, et al. Continuous control with deep reinforcement learning[J]. arXiv preprint arXiv:1509.02971, 2015.

[5] Cho, K. , Merrienboer, B. V. , Gulcehre, C. , Ba Hdanau, D. , Bougares, F. , & Schwenk, H. , et al. Learning phrase representations using RNN

encoder-decoder for statistical machine translation[J]. arXiv preprint arXiv:1406.1078, 2014.

[6] Xingjian S H I, Chen Z, Wang H, Yeung D Y, Wong W K, & Woo W C. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In Advances in neural information processing systems , 2015(pp. 802-810).

[7] Volodymyr M, Koray K, David S, Andrei A R, Joel V, Marc G B, et al. Human-level control through deep reinforcement learning[J]. Nature, 2019, 518(7540):529-533.