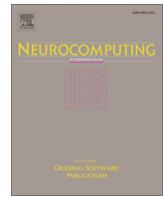




ELSEVIER

Contents lists available at ScienceDirect

# Neurocomputing

journal homepage: [www.elsevier.com/locate/neucom](http://www.elsevier.com/locate/neucom)

## Autonomous quadrotor obstacle avoidance based on dueling double deep recurrent Q-learning with monocular vision

Jiajun Ou<sup>a</sup>, Xiao Guo<sup>b,\*</sup>, Ming Zhu<sup>c</sup>, Wenjie Lou<sup>c</sup><sup>a</sup>School of Aeronautic Science and Engineering, Beihang University, Beijing 100191, China<sup>b</sup>Research Institute for Frontier Science, Beihang University, Beijing 100191, China<sup>c</sup>Institute of Unmanned System, Beihang University, Beijing 100191, China

### ARTICLE INFO

#### Article history:

Received 3 March 2020

Revised 24 January 2021

Accepted 10 February 2021

Available online 2 March 2021

Communicated by Zidong Wang

#### Keywords:

Unmanned aerial vehicle

Obstacle avoidance

Deep reinforcement learning

Depth estimation

### ABSTRACT

This paper proposes a novel learning-based framework to realize quadrotor autonomous obstacle avoidance with monocular vision. The framework adopts a two-stage architecture, consisting of a sensing module and a decision module. The sensing module trained in an unsupervised manner can extract depth information from the on-board camera image. Moreover, the decision module uses dueling double deep recurrent Q-learning to eliminate the adverse effects of the on-board monocular camera's limited observation capacity while choosing practical obstacle avoidance action. The framework has two advantages: (1) it enables the quadrotor to realize autonomous obstacle avoidance without any prior environment information or labeled datasets for training, and (2) its model can be easily updated while facing new application scenarios. The experiments in several different simulation scenes show that the trained framework outperforms a high passing rate in crowded environments and a good generalization ability for transformed scenarios.

© 2021 Published by Elsevier B.V.

## 1. Introduction

Unmanned aerial vehicles (UAVs) are widely used in both military and civil fields nowadays. UAVs can liberate people from monotonous or dangerous work such as searching and rescuing [1], package delivery [2] etc. However, typical UAV operation depends on the human remote control or follows a fixed flight route, which may be labor-intensive and inefficient. With the increase of mission complexity and scale, UAV needs to improve its autonomous flight capability. So UAV is required to perceive the environment, while dealing with the environment information and avoiding obstacles in its expected flight path to achieve autonomous flight. Typical on-board sensors for UAVs are monocular camera, stereo camera, LIDAR, Kinect [3] etc. While some sensors can directly output the depth information, such as LIDAR and Kinect, others get the depth information with additional calculations like a stereo camera.

The quadrotor is one of the most widely used UAVs because of its low cost, flexibility and simple structure. However, the weight and energy consumption of equipped sensors are limited with the restricted payload on quadrotors. In many cases, quadrotors can only afford a fixed monocular camera, providing limited envi-

ronment observation. Therefore, autonomous obstacle avoidance of quadrotor remains a challenging task.

Classical autonomous obstacle avoidance approaches are based on Simultaneous Localization and Mapping (SLAM) [4–6] or Structure from Motion (SfM) [7]. Those approaches solve this problem with two separate processes, mapping and planning. Firstly they build a local map of surroundings based on sensor information, then plan a path and repetitively update the local map [8–10]. SLAM and SfM based methods [11–13] estimate the camera motion and depth by triangulation at each time step. The critical step is the high-frequency feature extraction and matching in reconstructing the three dimensional local map from the sensor data. Though the SLAM and SfM based approaches have been proven to be effective in autonomous obstacle avoidance, their disadvantages are apparent. The feature extraction of SLAM and SfM based approaches may fail when facing an untextured obstacle, and the real-time process requests unbearable computation for the on-board unit [14].

Deep reinforcement learning provides an alternative to autonomous obstacle avoidance [15–17]. These methods do not need feature extraction and matching at the pixel level, so they may be executed more efficiently. However, the typical training process of learning-based methods requires sufficient groundtruth annotation data [18,19], which is expensive to obtain. Moreover,

\* Corresponding author.

E-mail address: [xiaoguo@buaa.edu.cn](mailto:xiaoguo@buaa.edu.cn) (X. Guo).

preparing the training data with groundtruth is even more effortful for quadrotors because the application scenarios are uncertain.

This paper proposes a novel framework based on a deep reinforcement learning algorithm, which can use the image data collected by on-board camera to make real-time decisions for effective obstacle avoidance. A sensing module and a decision module are connected in series in the framework. The sensing module applies unsupervised learning based depth estimation method to perceive surrounding obstacles. Since the image acquisition with a fixed on-board monocular camera can only provide a limited field of view, it leads the quadrotor autonomous obstacles avoidance to become a partially observable Markov decision process. According to the previous states, the decision module is designed to make obstacle avoiding decisions rather than only the current one. So it can solve the problem of partial observability.

Compared with SLAM or SFM based methods, our approach has better execution efficiency for no longer conducting feature matching between adjacent image frames. The characteristics of unsupervised learning make it relatively easier to prepare training dataset, so our approach is more suitable for practical applications. Our framework is trained in the ROS gazebo environment for conducting quadrotor obstacle avoidance in crowded scenarios. The trained framework validates the effectiveness in evaluation, and its performance remains in the transformed scenarios. Our framework's decision module maintains a high success rate in evaluation when dealing with new scenarios, even though the appearance, size, and location arrangement of obstacles are different from the training scenario.

The main contributions are as follows:

- The sensing module of our framework employs the unsupervised learning approach to perform depth estimation, which takes view synthesis as the supervisory signal. Thus, training the sensing module is free from the tedious preparation of data with groundtruth.
- We propose the dueling double deep recurrent Q network to learn the obstacle avoidance policy. Our proposed model shows better learning and executive efficiency under partial observable conditions than several other deep reinforcement models.
- We present a feasible solution for obstacle avoidance in scenario transformation. The decision module of our framework shows a good generalization in new scenarios.

## 2. Related work

Learning-based avoidance methods can be divided into end-to-end architecture and hierarchical architecture. The end-to-end architecture goes directly from sensor data to obstacle avoidance actions. Loquercio et al. [20] design a fast 8-layers residual network to output the steering angle and a collision probability for each input image. The network is trained by the dataset manually annotated by the authors. Kouris et al. [21] train convolutional neural networks (CNN) to predict distance-to-collision from the on-board monocular camera image. The proposed CNN is trained on the datasets annotated with real-distance labels obtained by the Ultrasonic and Infra-Red distance sensors. In the paper [22], the authors combine visual and distance sensor information to make autonomous obstacle avoidance decisions through the deep reinforcement learning algorithm. Han et al. [23] introduce a structure composed of double joint neural network estimators as the decision-maker, realizing obstacle avoidance by taking omnidirectional sonar readings as inputs. Park et al. [24] propose a framework for vision-based obstacle avoidance, where the decision-making policies are trained upon the supervision of actual human flight data. Moreover, Gandhi et al. [19] build a drone to sample

data in the crash, and their model learns a navigation policy from the sampled dataset. To improve data efficiency, Zhu et al. [25] propose a novel simulation environment to train the model, which provides high-quality 3D scenes and a physics engine. Although the end-to-end models can effectively avoid obstacles, the training of these models needs a large number of data labeled with obstacle distance or collision probability, which requires much manual or special device annotation. Researchers make efforts to prepare these training data, which costs a lot of time and workforce.

On the other side, many researchers adopt the hierarchical architecture to solve the monocular obstacle avoidance problem. The typical hierarchical architecture contains two separate parts, environment sensing and decision making. The monocular camera can only provide two-dimensional information directly, and it is necessary to perceive three-dimensional information of the environment. Depth estimation is one of the most commonly used methods in perception. Supervised learning-based depth estimation achieved considerable results [26–30]. For solving the problem that the labeled datasets are difficult to obtain, researchers have proposed depth estimation methods based on semi-supervised learning [31] and unsupervised/self-supervised learning [32–35].

Based on various monocular depth estimation methods, researchers have made progress in autonomous obstacle avoidance. Tai et al. [36] build a highly compact network structure which comprises a CNN front-end network for perception and a fully connected network for decision making. The authors record the synchronized depth maps by Kinect and the control commands by the human operator, then train the network with supervised learning. Sadeghi et al. [37] use the depth channel of Kinect to automatically annotate the RGB images with free-space/non-free-space labels, proposing a learning method to train a fully convolutional neural network, which can be used to perform collision-free indoor flight in the real world. In the paper [38], a fully convolutional neural network is constructed to predict depth from a raw RGB image, followed by a dueling architecture based deep double Q network for obstacle avoidance. Singla et al. [18] use recurrent neural networks with temporal attention to realize UAV obstacle avoidance and autonomous exploration. The authors train a conditional generative adversarial network to generate depth maps from RGB images. Zhang et al. [39] use CNN to estimate depth from RGB image and feed the depth image into the obstacle avoidance system, in which the proposed control algorithm steers the quadrotor to avoid obstacles.

Besides the obstacle avoidance methods based on depth estimation, researchers have proposed some obstacle avoidance methods based on other perception approaches. Back et al. [40] train a CNN based model to estimate the optical flow for obstacle avoidance. Dai et al. [41] design a two-stage end-to-end obstacle avoidance architecture, where a forward-facing monocular camera is used only. They train a CNN based model as the prediction mechanism to predict the steering angle and the collision probability simultaneously. Shin et al. [42] propose a new framework where a supervised segmentation network is trained with labels. Moreover, based on the segmentation, the framework applies policy gradient algorithms to control the drone to avoid obstacles. In the paper [43], the authors put forward a comprehensive solution which consists of deep-learning based object detection, image processing, RGB-D information fusion and Task Control System. For all these researches mentioned above, their model training processes require data with labels or groundtruth. In order to meet the needs of the training process for labeled data, Yang et al. [44] employ an online adaptive CNN for progressively improving depth estimation aided by monocular SLAM, which increases the complexity of the system and the requirements of computation. However, because the application scenario of quadrotors is hard to restrict and predict in practice, it is difficult to obtain sufficient training data for

supervised learning. Therefore, considering the problem of data acquisition, the model training proposed in previous works is neither convenient nor economical, limiting the practical application of these methods.

In order to reduce the difficulty of training and improve the feasibility of applications, we present a novel framework to achieve autonomous obstacle avoidance in this paper. The framework consists of two modules, and its training requires no annotated datasets. The first module is used to sense the environment, which adopts an unsupervised learning based depth estimation to generate a depth map. The second module responds to make obstacle avoidance decisions, whose policy is acquired through deep reinforcement learning. The former can be trained by raw RGB monocular image sequences, and the latter can be trained in the simulation environment. In this way, an autonomous obstacle avoidance method is proposed, which is efficient and relatively easy to train. In the face of new scenes, our framework only requires raw RGB image data to retrain the first module, and then it can adapt to the new working scenario.

### 3. Proposed method

In this paper, a two-stage framework is proposed to sense the environment with an on-board monocular camera and make decisions to avoid obstacles in flight. This framework utilizes an unsupervised deep learning method to estimate depth from the raw RGB monocular image. Furthermore, the framework can further choose proper action to conduct safe flight without collision according to the generated depth information. The selected action acts on the outer loop control of the quadrotor to realize the obstacle avoidance flight. Our framework provides a feasible solution for obstacle avoidance with no prior environmental information required.

#### 3.1. Problem definition

The problem of autonomous obstacle avoidance for quadrotors can be reduced to Markov Decision Processes (MDPs), which can be defined as tuple  $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}(s_{t+1}|s_t, a_t), \mathcal{R}(s_t, a_t) \rangle$ . Here  $\mathcal{S}$  is the set of states of the environment, while  $\mathcal{A}$  is the set of feasible actions.  $\mathcal{T}$  is the transition probability function and  $\mathcal{R}$  is the reward function. At each time step  $t$ , the vehicle receives the state  $s_t \in \mathcal{S}$  and propose action  $a_t \in \mathcal{A}$ . And the received reward  $r_t$  is given by the reward function  $\mathcal{R}(s_t, a_t)$ . In accordance with the transition model  $\mathcal{T}(s_{t+1}|s_t, a_t)$ , the vehicle moves into a new state  $s_{t+1}$ . The action  $a_t$  is sampled from the policy  $\pi = P(a_t|s_t)$ . The expectation of accumulative reward can be approximated by action-state-value function  $Q(s_t, a_t)$ , which is constructed by a deep neural network.

The key to this problem is to find the optimal policy  $\pi$  to maximize the accumulative future reward  $\mathbb{E}[\sum_t^\infty \gamma^t R(s_t, a_t)]$ , where  $\gamma$  is the discount factor. By choosing the optimal action which maximizes the Q-value each time, the optimal Q-value function can be computed using the Bellman equation

$$Q^*(s_t, a_t) = \mathbb{E}_{s_{t+1}} \left[ r_t + \gamma \max_{a_{t+1}} Q^*(s_{t+1}, a_{t+1}) | s_t, a_t \right] \quad (1)$$

The optimal policy is capable of leading the quadrotor to make corrective action decisions to avoid obstacles during flight.

#### 3.2. Sensing with unsupervised depth estimation

The sensing module of our framework constructs a fully convolutional neural network, which is capable of mapping directly from the input RGB images to the estimate of the underlying scene structure. It employs the DispNet [45] to generate a front view

depth map. The DispNet architecture is mainly based on an encoder-decoder design, as shown in Fig. 1. The kernel size of the first four convolutional layers is 7, 7, 5, 5, respectively, and that of the other layers is three in the architecture.

Inspired by the work in the paper [33], the sensing module is trained by the supervision signal that the task of view synthesis generates. The training process only requires the raw RGB image sequences obtained by the on-board monocular camera during the flight. For realizing depth estimation without external supervision, the pose network is introduced into the learning process. The pose network consists of seven convolutional layers followed by a  $1 \times 1$  convolutional layer with six output channels corresponding to three Euler angles and 3-dimensional translation, as shown in Fig. 2.

The image sequences captured by the on-board camera during the flight are stored in the replay buffer. In each training step, three images are sampled from the replay buffer randomly, including one target image  $I_t$  and two nearby source images  $I_s$  ( $I_{t-1}$  and  $I_{t+1}$ ) in the same sequence. These images are input to the depth estimation network and pose network at the same time. The depth estimation network generates the depth map  $\hat{D}_t$  from the target

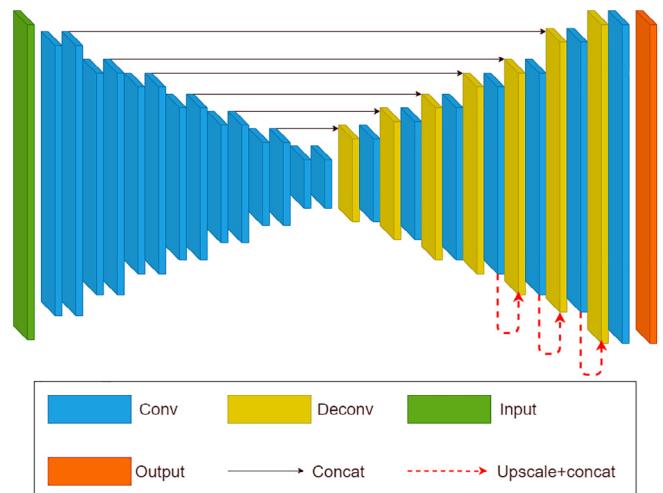


Fig. 1. Network architecture for the DispNet.

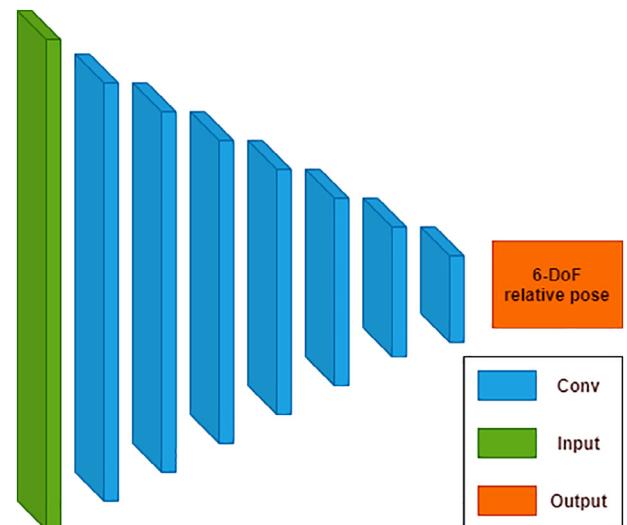


Fig. 2. Network architecture for the pose network.

image  $I_t$  only. The pose network takes both the target image  $I_t$  and the source images  $I_s$  as input, and outputs the relative camera poses  $\hat{T}_{t \rightarrow s}$  ( $\hat{T}_{t \rightarrow t-1}$  and  $\hat{T}_{t \rightarrow t+1}$ ).

Let  $\hat{I}_s$  donates the reconstructed target image from  $I_s$ . To reconstruct  $\hat{I}_s$ , pixels are sampled from  $I_s$  based on the depth map  $\hat{D}_t$  and the relative pose  $\hat{T}_{t \rightarrow s}$ .  $p_t$  represents the coordinates of a pixel in the target view, the corresponding coordinates  $p_s$  in the source view can be obtained as follows

$$p_s \sim K\hat{T}_{t \rightarrow s}\hat{D}_t(p_t)K^{-1}p_t, \quad (2)$$

where  $K$  represents the camera intrinsics matrix. The differentiable bilinear sampling mechanism [46] is used to obtain  $I_s(p_s)$  for populating the value of  $\hat{I}_s(p_t)$ .

The photometric reconstruction loss between the raw target image  $I_t$  and the reconstructed target image  $\hat{I}_s$  is used for training the networks, which can be defined as follows

$$\mathcal{L}_{ps} = \sum_s \sum_p |I_t(p) - \hat{I}_s(p)|, \quad (3)$$

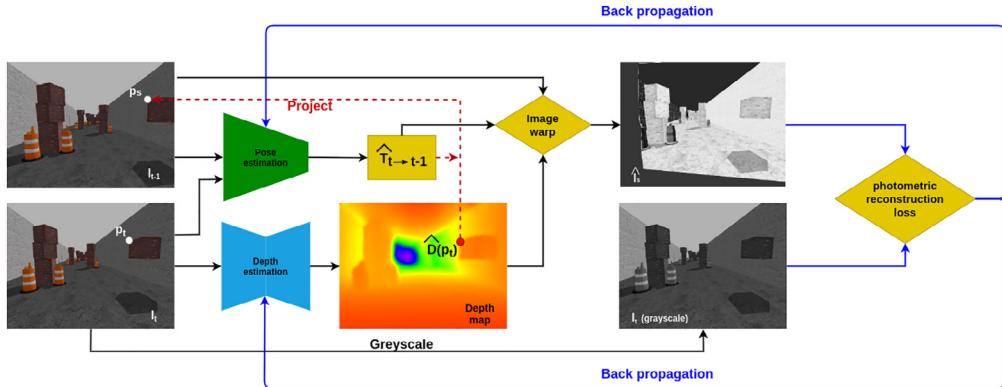
where  $p$  represents the index over pixel coordinates. By utilizing view synthesis as supervision, the depth estimation network is trained in an unsupervised manner from captured image sequences.

The training pipeline is shown in Fig. 3. The pose network and depth estimation network are trained together during the training process. The photometric reconstruction loss is calculated based on the results of depth estimation and pose estimation. By calculating the backpropagation of the photometric reconstruction loss, the parameters of the depth estimation network and the pose estimation network are updated at the same time. Note that the part of  $\hat{I}_s$  beyond the horizon of  $I_t$  will not be counted in the calculation.

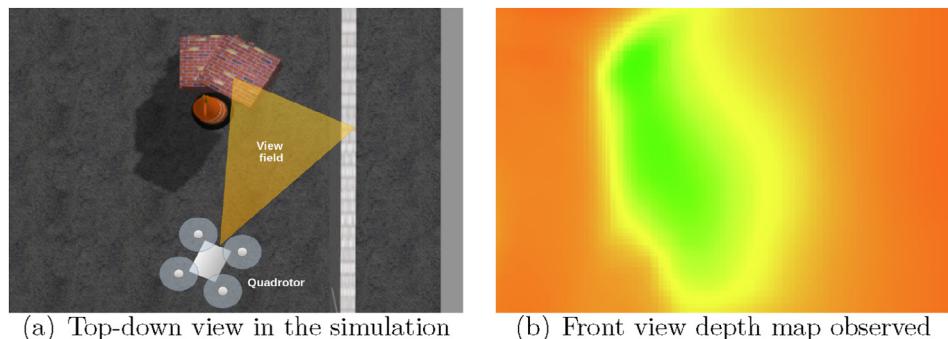
### 3.3. Dueling double deep recurrent Q network

The on-board monocular camera can only provide a limited field of view of the surrounding environment. The partial observability makes it hard to gain the optimal policy in some particular scenes [18]. As shown in Fig. 4, the quadrotor might fly straight forward and crash on the obstacle based on the current partial observation, while the proper action is turning left. Besides, the training data of the depth estimation network is captured by the on-board camera of the quadrotor during the flight. The unsupervised depth estimation method has been proven feasible, but using on-board camera data to train the model may raise a new problem. The limitation of the quadrotor's flight ability and the avoidance of crashes lead the data to be short of comprehensiveness, which may cause poor depth estimation performance in some scenes, shown in Fig. 5.

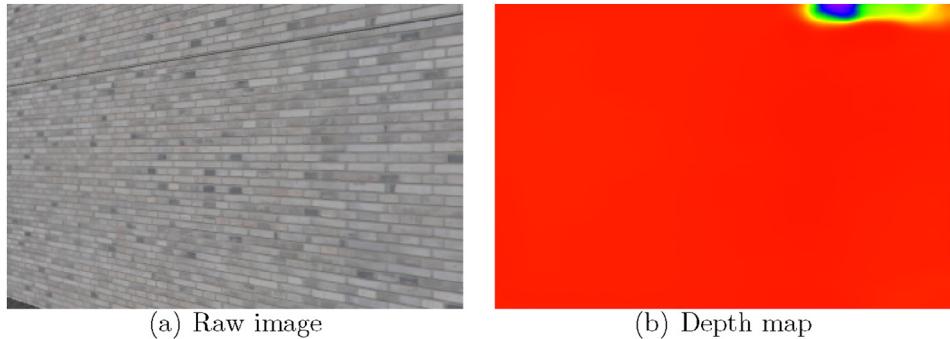
Considering the above situations, we treat the quadrotor obstacle avoidance as Partially Observable Markov Decision Processes (POMDPs) in this paper. The POMDP problem can be defined as tuple  $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \Omega, \mathcal{O} \rangle$ .  $\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}$  are respectively the set of states, actions, transitions, and rewards as before. Here  $\Omega$  is the set of observations, while  $\mathcal{O}$  is the set of the probability distributions. The on-board monocular camera gets observations  $o \in \Omega$  generated from the underlying system state according to the probability distribution  $o \sim \mathcal{O}(s)$ . At current time  $t$ , the observation  $o_t$  can only represent part of the current surrounding environment state  $s_t$ . Since estimating Q-value  $Q(o_t, a_t | \pi) \neq Q(s_t, a_t | \pi)$ , obstacle avoidance action  $a_t$  relying entirely on current observation may be fragile. Therefore, in this paper, a method that can use previous observation experience is proposed for improving the performance of obstacle avoidance. The model can extra useful environment information from sequential observation before the current time, making  $Q(o_t, a_t | \pi)$  closer to  $Q(s_t, a_t | \pi)$ . So it can eliminate the inter-



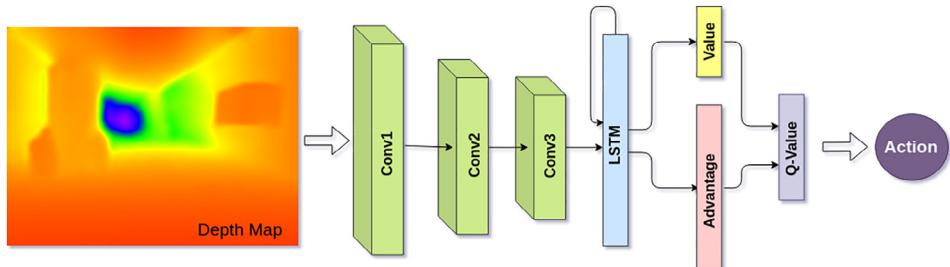
**Fig. 3.** Unsupervised learning based on view synthesis.



**Fig. 4.** A demonstration of collision cased by partial observation.



**Fig. 5.** Poor performance of depth estimation in some scenes.



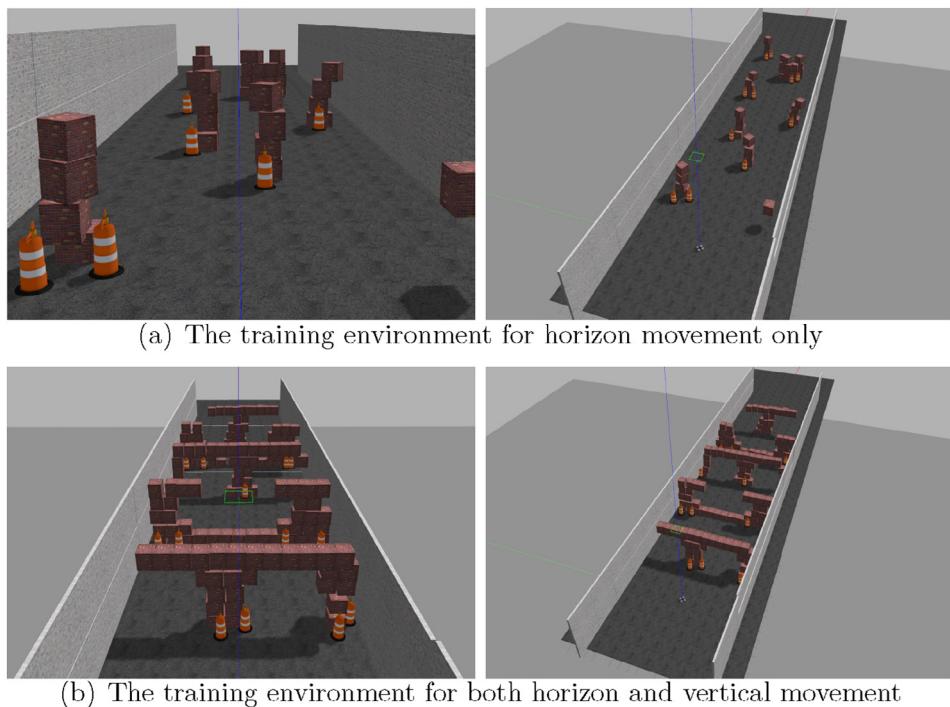
**Fig. 6.** The structure of dueling deep recurrent Q network.

**Table 1**  
Parameters of the dueling deep recurrent Q network.

Item	Size (height,width,channel)	Number	Stride
Depth map	(128,416,1)	–	–
Conv 1	(8,8,4)	–	4
Conv 2	(4,4,8)	–	2
Conv 3	(3,3,8)	–	2
LSTM	–	1152	–
FC for advantage	–	5 or 15	–
FC for value	–	1	–

ference of low-quality observation results and avoid quadrotors getting trapped in cases like the example in Fig. 4.

The decision module in our framework is based on the deep recurrent Q network [47](DRQN) with the dueling and double technology [48,49]. In the traditional dueling network, two streams are used to compute the value and advantage functions. The dueling network can improve performance and training speed. On the other hand, the double technology solving the problem of overoptimistic value estimation. Based on these previous research results, we combine the DRQN and dueling network by replacing



**Fig. 7.** The basic training environments in Gazebo.

one fully connected layer of the dueling network with an LSTM layer. This dueling deep recurrent Q network structure is shown in Fig. 6, and corresponding parameters are shown in Table 1.

#### 4. Training and testing

The proposed framework is implemented using the Pytorch framework. The sensing and decision modules are trained in the Gazebo simulation environment with a step-by-step training strategy. The depth estimation network is firstly trained. The trained depth estimation network is then used to generate depth maps, and the depth maps are used to train the dueling double deep recurrent Q network(D3RQN). Several models are trained and evaluated in multiple different simulation environments. Two basic training environments are used for training. One of them is for training the quadrotor to conduct horizon movement only. The other is for both horizon and vertical movement. Fig. 7 shows the basic training environments in the simulation.

##### 4.1. Depth estimation

The image data for training the depth estimation network is collected in the simulation environments by the on-board monocular camera of the manually controlled quadrotor. These images are used to train the depth estimation network. The training hyperparameters are shown in Table 2.

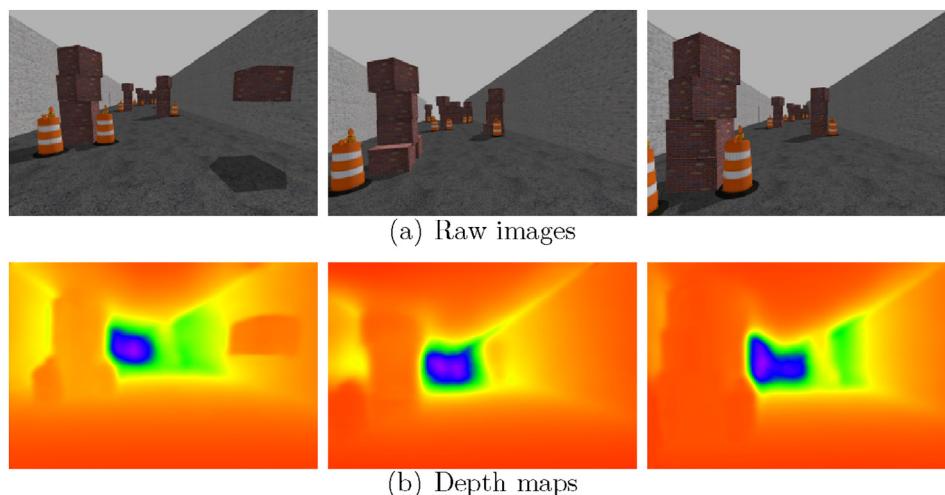
The depth estimation network is evaluated after training 30000 iterations, which is much less than that in the original paper [33]. The examples of depth estimation are shown in Fig. 8. And we test the depth estimation network on an NVIDIA GeForce RTX 2070 GPU with 16 GB RAM and Intel Core i7 processor machine, and the depth map generation rate reaches more than 30 Hz.

**Table 2**  
Parameter settings for training the depth estimation network.

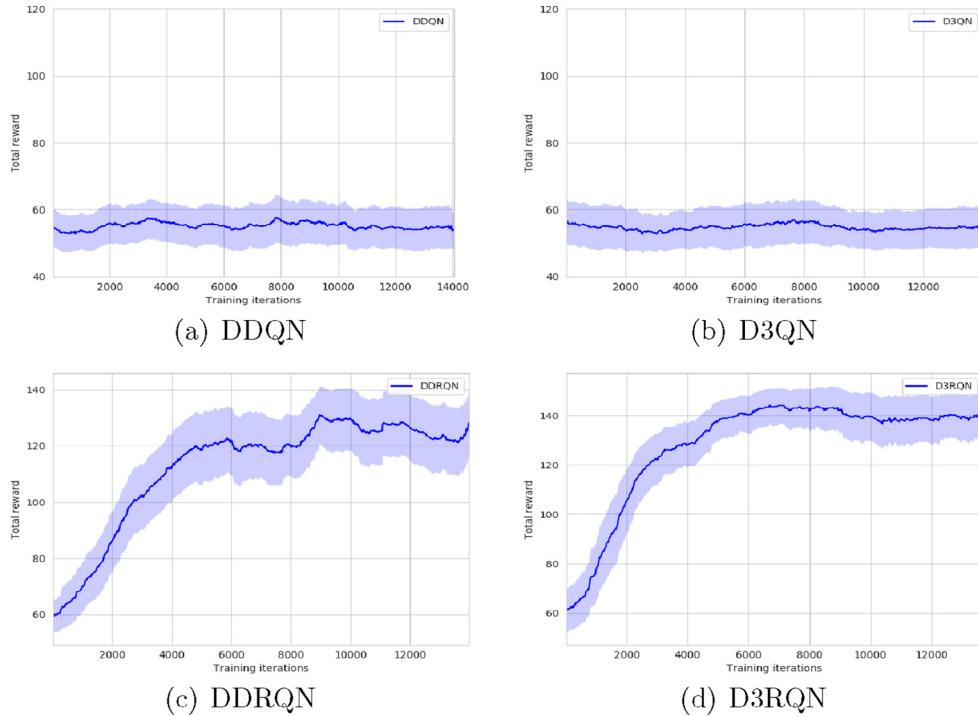
Parameters	Value
Image number	5000
Batch size	4
Learning rate	0.00005
Image acquisition interval	0.4 s
Camera linear velocity	2 m/s
Training iteration	30000
Optimizer	Adam

**Table 4**  
Parameter settings for training the dueling double deep recurrent Q network.

Parameters	Value
Batch size	32
Discount factor	0.99
Learning rate	0.0003
Input sequence length	5
Action time interval	0.4 s
Target network update frequency	300
Optimizer	Adam



**Fig. 8.** Performance of depth estimation (Red: near, Blue: far).



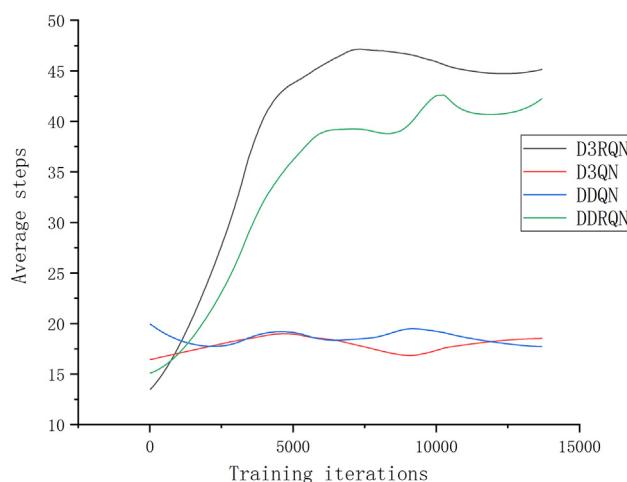
**Fig. 9.** Learning curves of the DDQN, D3QN, DDRQN and our D3RQN.

With the execution of each action, the quadrotor obtains a reward, which defined as

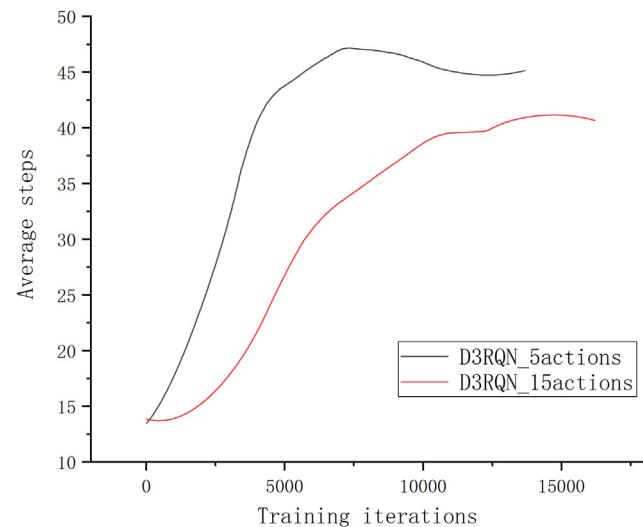
$$R = \begin{cases} d_{\text{nearest}} & \text{if safe} \\ -1 & \text{otherwise} \end{cases} \quad (4)$$

where the  $d_{\text{nearest}}$  is the distance to the nearest obstacle, and the safe distance is 0.5. When the  $d_{\text{nearest}}$  is smaller than the safe distance, or the flight altitude is out of range (from 0.5 m to 4 m), The flight is considered unsafe, and the training episode ends with a negative reward.

Besides our dueling double deep recurrent Q network, three other RL based models are trained in the simulation environment with similar parameters. They are double deep Q network (DDQN), dueling double deep Q network (D3QN), and double deep recurrent Q network (DDRQN). The learning curves of the four models are shown in Fig. 9. Fig. 10 presents the comparison of different models. And Fig. 11 compares our dueling double deep recurrent Q network models with different action settings.



**Fig. 10.** The performance comparison of the four methods after smoothing.

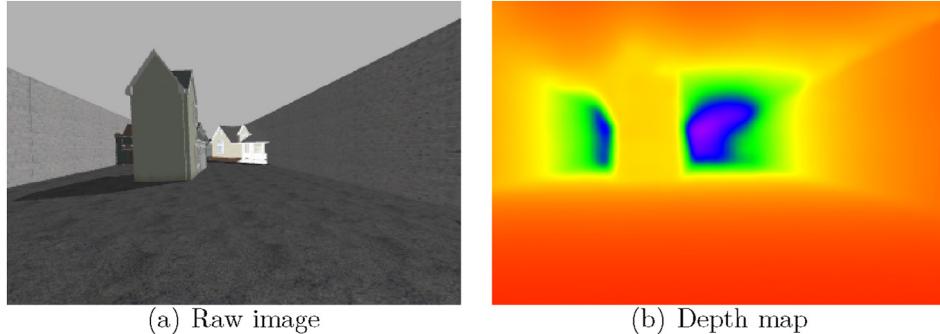


**Fig. 11.** The performance comparison of different action space after smoothing.

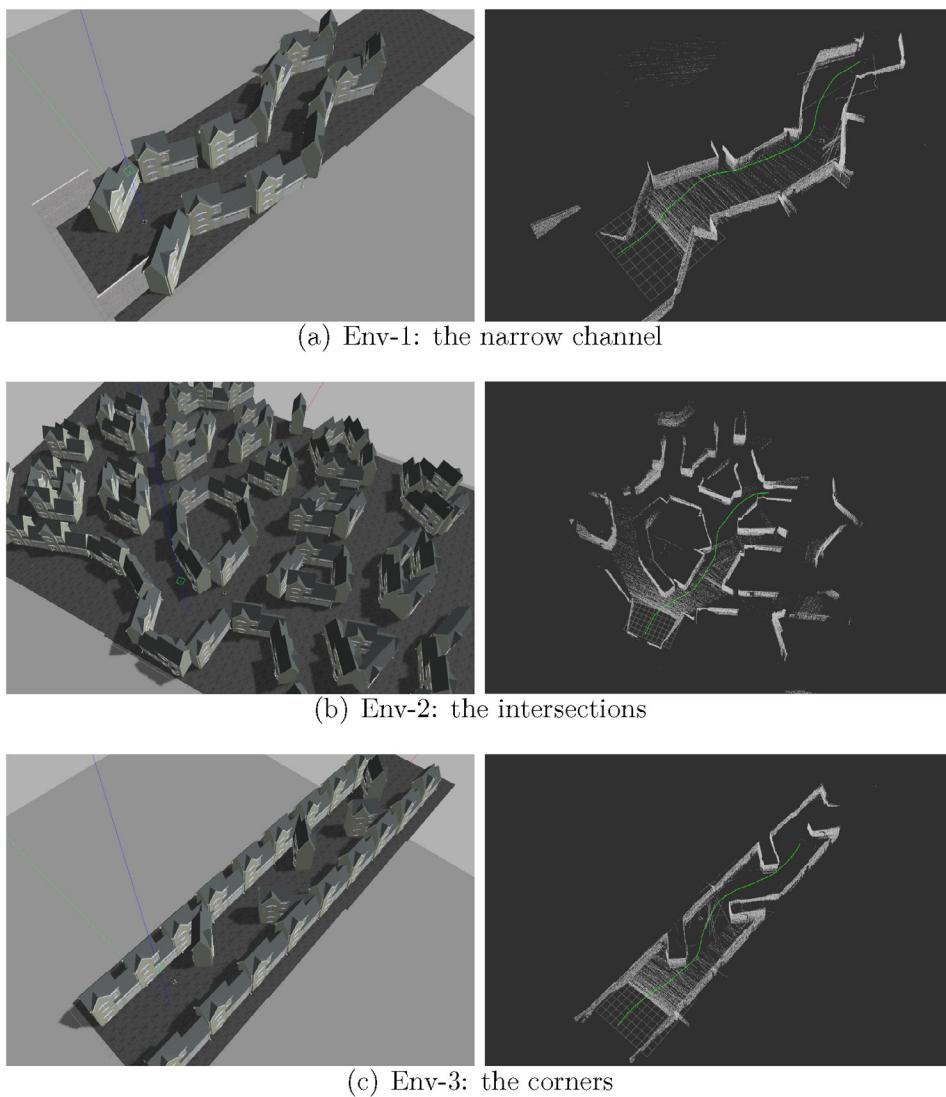
**Table 5**  
Test results of 5 different models.

Model	Success rate
Straight	0
Random	0.002
DDQN	0.137
D3QN	0.152
DDRQN	0.673
Our approach(5 actions)	0.994
Our approach(15 actions)	0.967

Five RL based models are tested in the Gazebo environment. In the test, once the model controls the quadrotor to fly more than 50 steps safely, it is considered a success. These models are evaluated by calculating the success rate of each model in 2000 times test



**Fig. 12.** Depth prediction performance after scenario transformation.



**Fig. 13.** The test environments and success trajectories.

flight. The results are shown in Table 5. Our whole framework can run on the machine mentioned in Section 4.1 at more than 15 Hz in the test.

#### 4.3. Performance after scenario transformation

Since the scenario uncertainty in UAV applications is significant, the quadrotor obstacle avoidance ability should be effective in

different scenarios. Previous researches have focused on building complex models to adapt to different scenarios as much as possible. However, it is hard for training datasets to cover all possible scenario types. And a complex model is not suitable for the on-board processing unit of a quadrotor. Rather than solving all problems in one model, our approach is dedicated to realizing more convenient training when facing new application scenarios.

**Table 6**

Test results of obstacle avoidance after scenario transformation.

Model	Success rate (5 actions)			Success rate (15 actions)		
	Env-1	Env-2	Env-3	Env-1	Env-2	Env-3
Straight	0	0	0	0	0	0
Random	0.003	0.001	0	0	0	0
Our approach	0.923	0.968	0.938	0.908	0.942	0.911

In this section, new simulation environments are used to test our framework. The appearance, size, shape and location of obstacles in the new simulation environments are different from those in the basic environment in Fig. 7. Before conducting the test, the only thing required is retraining the depth estimation network with image sequences obtained in the new environments. The depth estimation performance after scenario transformation is shown in Fig. 12.

With the retrained depth estimation network, the whole framework is tested in new simulation environments. All the operating parameters such as action time interval, action values and nearest distance remain the same with those in the basic training environments. The test environments and the success trajectories<sup>1</sup> are shown in Fig. 13. It is worth emphasizing that we reuse the dueling double deep recurrent Q network trained in the basic environments without any fine-tune operation. These simulation scenarios respectively represent narrow channels, intersections and corners. And Table 6 presents the performance of our method after the scenario transformation.

## 5. Discussion

In this paper, a learning-based framework is proposed for quadrotor autonomous obstacle avoidance. This framework has some characteristics as follows:

- An unsupervised learning-based method for depth estimation is used for the environment perception module in our framework. It is novel to apply this method to the quadrotor autonomous obstacle. In this paper, the module is trained and tested with raw image data obtained by the on-board monocular camera in the simulation. The training and testing results show that the trained module can effectively estimate the distance of obstacles on the quadrotor route. However, due to the limitation of the quadrotor's flight ability, the on-board camera is difficult to obtain enough data under certain circumstances, which results in the decline of depth estimation ability.
- The quadrotor mentioned in this paper only relies on a monocular camera to obtain environment information, limiting its observation ability and making it difficult to make effective obstacle avoidance decisions. To solve this problem, we propose the D3RQN to learn the policy efficiently with limited observations. It can learn the obstacle avoidance policy from previous observations rather than only from the current one. Compared with some other typical reinforcement learning based methods, our framework has better learning efficiency and test performance.
- Since scenario transformation is pervasive in UAV applications, it is essential to keep obstacle avoidance ability after scenario transformation. The test results show that our framework can effectively make proper obstacle avoidance decisions in the new scenarios after retraining the depth estimation network

only, even though the obstacles in new scenarios are different in appearance, shape, size and location arrangement. Besides, retraining the depth estimation network in our framework requires raw image sequences without labels or groundtruth only, which is convenient to prepare.

## 6. Conclusion

In this paper, the framework based on the D3RQN is presented. It can guide the quadrotor to achieve autonomous obstacle avoidance on top of the image captured by an on-board monocular camera only. The training and testing results demonstrate that the D3RQN has a better learning efficiency and testing performance than some other approaches such as double DQN, D3QN and double DRQN. The test in different scenarios shows that our framework has a good scenario generalization ability.

In our future work, the proposed framework will have a more complex network structure to control the quadrotor to perform more complex actions to avoid moving obstacles. Another interested area for our future work would be implementing and testing the proposed framework on a real quadrotor. The improvement of efficiency is also in consideration to fit the limited on-board computing resource better.

## CRediT authorship contribution statement

**Jiajun Ou:** Methodology, Software, Writing - original draft, Writing - review & editing, Validation. **Xiao Guo:** Conceptualization, Software, Writing - original draft, Writing - review & editing. **Ming Zhu:** Supervision, Writing - review & editing. **Wenjie Lou:** Data curation, Validation.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgement

aaa

## References

- [1] S. Waharte, N. Trigoni, Supporting search and rescue operations with uavs, in: 2010 International Conference on Emerging Security Technologies , IEEE, 2010, pp. 142–147.
- [2] B.D. Song, K. Park, J. Kim, Persistent uav delivery logistics: Milp formulation and efficient heuristic, Comput. Ind. Eng. 120 (2018) 418–428.
- [3] Z. Zhang, Microsoft kinect sensor and its effect, IEEE Multimedia 19 (2) (2012) 4–10.
- [4] R. Mur-Artal, J.M.M. Montiel, J.D. Tardós, ORB-SLAM: a versatile and accurate monocular SLAM system, IEEE Trans. Rob. 31 (5) (2015) 1147–1163.
- [5] J. Engel, T. Schöps, D. Cremers, Lsd-slam Large-scale direct monocular slam, in: European conference on computer vision, Springer, 2014, pp. 834–849.
- [6] M. Montemerlo, S. Thrun, Simultaneous localization and mapping with unknown data association using fastslam, 2003 IEEE International Conference on Robotics and Automation (Cat No. 03CH37422), vol. 2, IEEE, 2003, pp. 1985–1991.

<sup>1</sup> The video recordings of the success trajectories in simulation tests are available in [https://github.com/Mayooo0/Aoa\\_D3RQN\\_results](https://github.com/Mayooo0/Aoa_D3RQN_results)

- [7] C. Wu, Towards linear-time incremental structure from motion, in: International Conference on 3D Vision-3DV 2013, IEEE, 2013, pp. 127–134.
- [8] T. Zeng, B. Si, Mobile robot exploration based on rapidly-exploring random trees and dynamic window approach, in: 2019 5th International Conference on Control, Automation and Robotics (ICCAR), 2019. .
- [9] El Harik, Chouaib Houssein, Audun Korsaeth, Combining hector slam and artificial potential field for autonomous navigation inside a greenhouse, *Robotics* (2018).
- [10] C.Y. Wu, H.Y. Lin, Autonomous mobile robot exploration in unknown indoor environments based on rapidly-exploring random tree, in: International Conference on Industrial Technology, 2019. .
- [11] V.S. Kalogeiton, K. Ioannidis, G.C. Sirakoulis, E.B. Kosmatopoulos, Real-time active slam and obstacle avoidance for an autonomous robot based on stereo vision, *Cybern. Syst.* 50 (3) (2019) 239–260.
- [12] K.-T. Song, Y.-H. Chiu, L.-R. Kang, S.-H. Song, C.-A. Yang, P.-C. Lu, S.-Q. Ou, Navigation control design of mobile robot by integrating obstacle avoidance and lidar slam, in: 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE, 2018, pp. 1833–1838. .
- [13] S. Wen, Y. Zhao, X. Yuan, Z. Wang, D. Zhang, L. Manfredi, Path planning for active slam based on deep reinforcement learning under unknown environments, *Intel. Serv. Robot.* (2020) 1–10.
- [14] J. Li, Y. Bi, M. Lan, H. Qin, M. Shan, F. Lin, B.M. Chen, Real-time simultaneous localization and mapping for uav: a survey, in: Proc. of International micro air vehicle competition and conference, 2016, pp. 237–242.
- [15] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, et al., Mastering the game of go without human knowledge, *Nature* 550 (7676) (2017) 354–359.
- [16] S. Levine, C. Finn, T. Darrell, P. Abbeel, End-to-end training of deep visuomotor policies, *J. Mach. Learn. Res.* 17 (1) (2016) 1334–1373.
- [17] V. Mnih, K. Kavukcuoglu, D. Silver, A.A. Rusu, J. Veness, M.G. Bellemare, A. Graves, M. Riedmiller, A.K. Fidjeland, G. Ostrovski, et al., Human-level control through deep reinforcement learning, *Nature* 518 (7540) (2015) 529–533.
- [18] A. Singla, S. Padakandla, S. Bhatnagar, Memory-based deep reinforcement learning for obstacle avoidance in uav with limited environment knowledge, *IEEE Trans. Intell. Transp. Syst.* (2019).
- [19] D. Gandhi, L. Pinto, A. Gupta, Learning to fly by crashing, in: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2017, pp. 3948–3955.
- [20] A. Loquercio, A.I. Maqueda, C.R. Del-Blanco, D. Scaramuzza, Dronet: Learning to fly by driving, *IEEE Robot. Autom. Lett.* 3 (2) (2018) 1088–1095.
- [21] A. Kouris, C.-S. Bouganis, Learning to fly by myself: a self-supervised cnn-based approach for autonomous navigation, in: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) , IEEE, 2018, pp. 1–9.
- [22] S. Yang, Z. Meng, X. Chen, R. Xie, Real-time obstacle avoidance with deep reinforcement learning three-dimensional autonomous obstacle avoidance for uav, in: Proceedings of the 2019 International Conference on Robotics, Intelligent Control and Artificial Intelligence, 2019, pp. 324–329.
- [23] X. Han, J. Wang, J. Xue, Q. Zhang, Intelligent decision-making for 3-dimensional dynamic obstacle avoidance of uav based on deep reinforcement learning, in: 11th International Conference on Wireless Communications and Signal Processing (WCSP) , IEEE, 2019, pp. 1–6.
- [24] B. Park, H. Oh, Vision-based obstacle avoidance for uav via imitation learning with sequential neural networks, *Int. J. Aeronaut. Space Sci.* (2020) 1–12.
- [25] Y. Zhu, R. Mottaghi, E. Kolve, J.J. Lim, A. Gupta, L. Fei-Fei, A. Farhadi, Target-driven visual navigation in indoor scenes using deep reinforcement learning, in: 2017 IEEE international conference on robotics and automation (ICRA), IEEE, 2017, pp. 3357–3364. .
- [26] D. Eigen, C. Puhrsch, R. Fergus, Depth map prediction from a single image using a multi-scale deep network, in: Advances in neural information processing systems, 2014, pp. 2366–2374. .
- [27] D. Eigen, R. Fergus, Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture, in, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 2650–2658.
- [28] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3431–3440.
- [29] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, N. Navab, Deeper depth prediction with fully convolutional residual networks Fourth international conference on 3D vision (3DV), IEEE 2016 (2016) 239–248.
- [30] Y. Hua, H. Tian, Depth estimation with convolutional conditional random field network, *Neurocomputing* 214 (2016) 546–554.
- [31] Y. Kuznetsov, J. Stuckler, B. Leibe, Semi-supervised deep learning for monocular depth map prediction, in, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6647–6655.
- [32] C. Godard, O. Mac Aodha, G.J. Brostow, Unsupervised monocular depth estimation with left-right consistency, in, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 270–279.
- [33] T. Zhou, M. Brown, N. Snavely, D.G. Lowe, Unsupervised learning of depth and ego-motion from video, in, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1851–1858.
- [34] Z. Yin, J. Shi, Geonet: unsupervised learning of dense depth, optical flow and camera pose, in, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 1983–1992.
- [35] L. Chen, W. Tang, T.R. Wan, N.W. John, Self-supervised monocular image depth learning and confidence estimation, *Neurocomputing* (2019).
- [36] L. Tai, S. Li, M. Liu, A deep-network solution towards model-less obstacle avoidance, in: 2016 IEEE/RSJ international conference on intelligent robots and systems (IROS), IEEE, 2016, pp. 2759–2764. .
- [37] F. Sadeghi, S. Levine, Cad2rl: Real single-image flight without a single real image, arXiv preprint arXiv:1611.04201 (2016).
- [38] L. Xie, S. Wang, A. Markham, N. Trigoni, Towards monocular vision based obstacle avoidance through deep reinforcement learning, arXiv preprint arXiv:1706.09829 (2017). .
- [39] Z. Zhang, M. Xiong, H. Xiong, Monocular depth estimation for uav obstacle avoidance, in, in: 2019 4th International Conference on Cloud Computing and Internet of Things (CCCIOT) , IEEE, 2019, pp. 43–47.
- [40] S. Back, G. Cho, J. Oh, X.-T. Tran, H. Oh, Autonomous uav trail navigation with obstacle avoidance using deep neural networks, *J. Intell. Robot. Syst.* (2020) 1–17.
- [41] X. Dai, Y. Mao, T. Huang, N. Qin, D. Huang, Y. Li, Automatic obstacle avoidance of quadrotor uav via cnn-based learning, *Neurocomputing* (2020).
- [42] S.-Y. Shin, Y.-W. Kang, Y.-G. Kim, Reward-driven u-net training for obstacle avoidance drone, *Expert Syst. Appl.* 143 (2020) 113064.
- [43] D. Wang, W. Li, X. Liu, N. Li, C. Zhang, Uav environmental perception and autonomous obstacle avoidance: a deep learning and depth camera combined solution, *Comput. Electron. Agric.* 175 (2020) 105523.
- [44] X. Yang, H. Luo, Y. Wu, Y. Gao, C. Liao, K.-T. Cheng, Reactive obstacle avoidance of monocular quadrotors with online adapted depth prediction network, *Neurocomputing* 325 (2019) 142–158.
- [45] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, T. Brox, A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4040–4048.
- [46] M. Jaderberg, K. Simonyan, A. Zisserman, K. Kavukcuoglu, Spatial transformer networks, *Advances in Neural Information Processing Systems 28 (NIPS 2015)* (06 2015). .
- [47] M. Hausknecht, P. Stone, Deep recurrent q-learning for partially observable mdps, in: 2015 AAAI Fall Symposium Series, 2015. .
- [48] Z. Wang, T. Schaul, M. Hessel, H. Van Hasselt, M. Lanctot, N. De Freitas, Dueling network architectures for deep reinforcement learning, arXiv preprint arXiv:1511.06581 (2015). .
- [49] H. Van Hasselt, A. Guez, D. Silver, Deep reinforcement learning with double q-learning, in: Thirtieth AAAI conference on artificial intelligence, 2016.



**Jiajun Ou** received the B.S. degree in aircraft design from Beihang University, Beijing, China, in 2012. He is currently pursuing a Ph.D. degree with the School of Aeronautic Science and Engineering, Beihang University. His research interests include intelligent control, deep learning and reinforcement learning.



**Xiao Guo** received the B.S. and Ph.D. degrees in aircraft design from Beihang University, Beijing, China, in 2009 and 2013, respectively. He was a Postdoctoral Fellow with Beihang University, from 2013 to 2018, where he is currently an Assistant Professor with the Frontier Institute of Science and Technology Innovation. His current research interests include reinforcement learning, machine learning, and flight control.



**Ming Zhu** received the M.Sc. and Ph.D. degrees in aircraft design from Beihang University, Beijing, China, in 1998 and 2006, respectively. He was a Postdoctoral Fellow with Beihang University, from 2006 to 2007, where he is currently an Associate Professor with the Institute of Unmanned System. His research interests include solar unmanned aerial vehicles and unmanned aerial vehicle structure optimization.



**Wenjie Lou** received the Ph.D. degree in aircraft design from Beihang University, Beijing, China, in 2019. From 2018 to 2020, he was a Post-doctoral Fellow in the School of Electronic and Information Engineering, Beihang University. He is currently an Assistant Professor with the Institute of Unmanned System. His research interests include nonlinear control systems, flight control, and reinforcement learning.