Memory-based Deep Reinforcement Learning Method for Obstacle Avoidance in UAV

Abstract

Abhik Singla
Indian Institute of Science
abhiksingla10@gmail.com

Sindhu Padakandla
Indian Institute of Science
sindhupr@iisc.ac.in

Shalabh Bhatnagar
Indian Institute of Science
shalabh@iisc.ac.in

1 INTRODUCTION

Unmanned aerial vehicles (UAVs) are cyber-physical systems that can be operated autonomously using onboard computers. Owing to their small size and light weight, UAVs can penetrate into constricted spaces or effortlessly glide over pre-specified geographical areas, the majority of which may possibly be beyond the reach of humans. However, UAVs still lack the ability to avoid obstacles, which is a non-trivial task because the obstacles might be so positioned that avoiding them requires delicate and dexterous movements. To be able to avoid obstacles, the UAV must be able to perceive the distance between itself and the obstacles along with other visual cues such as the shape of the obstacle and it's height. This crucial visual information enables a UAV to infer traversable spaces and obstacles.

Kinect, LIDAR, SONAR, optical flow, and stereo camera sensors are widely used for depth estimation (see [9]) and hence these can be potentially used for obstacle avoidance (OA) as well. However, these sophisticated sensors are expensive and add unnecessary burden to the UAV in terms of weight as well as consumption of power. Other sensors, like for example, the monocular camera, is essential for every UAV application, because, it gives visual information. The monocular camera is a low-cost sensor which provides RGB images of the UAV's ambient environment. In comparison to the heavy-weight sensors mentioned earlier, a monocular camera is light-weight. The question then is whether we can use a monocular camera for depth estimation as well and plausibly for obstacle avoidance.

Taking cue from how humans learn to avoid obstacles with *limited* access to the environment, we propose a deep reinforcement learning (DRL) method which enables the UAV controller to collect relevant information from monocular RGB images observed over time and utilize this information to avoid obstacles dexterously. Our work based on recurrent neural network (RNN) architecture with an additional function called *Temporal Attention* adds a new dimension to the existing work on UAV obstacle avoidance (see [1, 8]). Using this architecture the UAV controller learns a control policy to avoid obstacles.

2 MODEL FORMULATION AND METHOD

The objective of our work is to find a suitable policy for UAV navigation that avoids obstacles (both stationary and mobile). We propose a POMDP model $\langle S, A, P, R, \Omega, O, \gamma \rangle$ for the OA problem. Here S is the set of states of the environment, while A is the set of *feasible* actions. P is the transition probability function that models the evolution of states based on actions chosen and is defined as $P : S \times A \times S \rightarrow [0, 1]$. R is the *reinforcement* or the *reward* function defined as $R : S \times A \rightarrow \mathbb{R}$. The reward function serves as a *feedback* signal to the UAV for the action chosen. Ω is the set of observations and an observation $o \in \Omega$ is an estimate of the true state s. $O : S \times A \times \Omega \rightarrow [0, 1]$ is a conditional probability distribution over Ω, while $\gamma \in (0, 1)$ is the discount factor. At each time t, the environment state is $s_t \in S$. The UAV takes an action $a_t \in A$ which causes the environment to transition to state s_{t+1} with probability $P(s_{t+1}|s_t, a_t)$. Based on this transition, the UAV receives an observation $o_t \in \Omega$ which depends on s_{t+1} with probability $O(o_t|s_{t+1}, a_t)$. The aim is to solve the obstacle avoidance problem, which translates to the task of finding an optimal *policy* $\pi^* : \Omega \rightarrow A$. By determining an optimal policy, the UAV controller is able to select an action at each time step t that maximizes the expected sum of discounted rewards over all starting states s, which is denoted as $\mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)|s_0 = s\right]$.

The input to our model is the monocular RGB image, without any depth information. Our model extracts the depth map from the RGB image which is the observation o for the UAV controller. The depth map predicted from the RGB image indicates the distance between the objects and the UAV. Given an observation, the feasible actions (A) available for the UAV are "go straight", "turn right" and "turn left". The reward function R is designed using the depth information as $R_t = \min\left(1, \frac{d_t - r_d}{\sigma - r_d}\right)$, where d_t is the distance to the nearest obstacle at time t and depends on the action taken, r_d is the radius of the drone and σ is the threshold distance. In order to determine the functions P and O, we must be aware of the structure of the environment and the motion dynamics of the UAV. In practice, these are impossible to know, since the UAV navigates in unknown, unstructured environments in the presence of other factors like wind, turbulence etc. So our method, denoted as DRQN+A [9], learns optimal policy for UAV navigation using deep Q-Networks [6], which utilizes a neural network parametrized by weights (w) to approximate the Q-value (denoted as $Q(s, a|w)$) for a given state input. Experience replay improves the stability of the algorithm in which experience tuples (s, a, r, s') are stored in a replay memory (D). However, in the UAV obstacle avoidance problem, the method has access to only the depth map (which is an observation). In order to better estimate the underlying states and their evolution we augment DQN with a recurrent architecture, namely LSTM [4]. The recurrent layer integrates the observations over time to better estimate the underlying states. In addition to LSTM, our architecture utilizes an additional layer known as temporal attention [7]. The temporal attention layer makes the recurrency more effective

by identifying the weights of the past observations in accordance with their importance on decision-making.

3 EXPERIMENTAL SETUP AND RESULTS

Our proposed method takes as input a RGB image (denoted x) of the surrounding environment, extracts a depth map (image) (denoted y) from the RGB image and provides an optimal direction to steer the UAV away from obstacles. In order to obtain a depth image from a RGB image, we utilize conditional generative adversarial network (cGAN) [5] for the image-to-image translation.

cGAN Training to obtain Depth Images : The proposed conditional GAN is initially trained on a total of 90, 000 RGB-D image pairs collected from the *Gazebo* simulated environments each having different characteristics. We simulated a number of indoor environments which consist of broad and narrow hallways, small and large enclosed areas with floorings ranging from asphalt to artificial turf. The simulated environments also contain structured and unstructured obstacles like humans, traffic cones, tables etc., placed at random positions and with random orientation. The walls and obstacles with diverse shapes, textures and colours provide abundant visual information for effective learning. The RGB-D image pairs are collected using a Kinect sensor mounted on the flying drone in simulation, covering all possible viewpoints. Further, the dataset is augmented off-line by random flipping, adding random jitter and random alteration to the brightness, saturation, contrast and sharpness. The network is trained on the entire collected dataset for 20 epochs in batches of size 4. For the learned OA policy to be effectively transferable to the real physical systems, we degrade the kinect sensor RGB images with Gaussian blurring, random jitter and superpixel replace to make the visual information more close to reality.

DQN Training with LSTM and Temporal Attention : For RL algorithms to learn an effective collision avoidance policy, the UAV learning agent must have enough experience of undesirable events like collision. Training a learning algorithm on a fragile drone in a physical environment is expensive and hence the performance of DRL algorithms is usually demonstrated on simulated environments. In this work, we build and test our UAV collision avoidance algorithms on the simulated environments used to obtain RGB images for depth network evaluation. Our method initially trains the UAV by starting off with simple hallway environments free of obstacles. Gradually, the environment complexity is increased by narrowing down the pathways, enclosing the free space and increasing the density of obstacles. The proposed control network is trained to learn the observation-action value over the last L observations (depth images received from the cGAN architecture) corresponding to the three actions *"go straight"*, *"turn left"* and *"turn right"*, respectively.

Experimental Results : We evaluate the performance of our proposed method which utilizes Deep Recurrent Q-network with Temporal Attention (DRQN+A) architecture. The performance results are compared with the baseline DQN [8], D3QN [2] and DRQN [3]. We also implement two other policies - random and straight. The random policy picks an action with equal probability for each observation, while the straight policy always picks the "go straight" action. The metric used for performance evaluation is the average number of steps taken until collision with an obstacle. All methods are trained in 12 different simulated indoor environments comprising of hallways and rooms with obstacles of varying structures and sizes. The trained models are tested on six simulated environments which are not used for training, out of which results for three environments are shown in Table 1. These environments comprise of enclosed areas with randomly scattered static obstacles of varying sizes and structures. The first ia a maze-like environment, the second is small enclosed area having poles in between. The third environment simulates a cafe-like environment and has 7 human actors randomly walking inside the cafe.

	Env-1	Env-2	Env-3
Straight	61±16	58±14	76±23
Random	125±84	176±121	113±83
DQN	207±103	229±95	286±142
D3QN	248±109	271±104	297±133
DRQN+A	323±134	342±131	326±156

Table 1: Results indicating the average number of steps taken by UAV (along with standard deviation) until collision.

We analyze performance of DRQN+A for 200 episodes in each environment. Table 1 indicates the average number of steps the UAV takes until collision after training. From Table 1, it can be seen that using our approach, the UAV flies for the maximum number of time instants until collision. We also observed through experiments that our designed cGAN depth network had an inference rate of 1.4Hz on an NVIDIA GeForce GTX 1050 mobile GPU with 8GB RAM and Intel i7 processor machine, which is quite advantageous for robotic applications. Additionally, we observed that the average energy consumption of DRQN+A is 0.0571 Wh/m and 0.0743 Wh/m for D3QN. The reasoning for this is that in testing we observed that with our proposed method, the UAV's "wobbling" motion is highly reduced when compared to D3QN.

REFERENCES

[1] P. Chakravarty et al. 2017. CNN-based single image obstacle avoidance on a quadrotor. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*. 6369–6374.

[2] Van Hasselt et al. 2016. Deep Reinforcement Learning with Double Q-Learning.. In *AAAI*, Vol. 2. Phoenix, AZ, 5.

[3] Matthew Hausknecht and Peter Stone. 2015. Deep Recurrent Q-Learning for Partially Observable MDPs. In *2015 AAAI Fall Symposium Series*.

[4] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[5] P. Isola et al. 2017. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 5967–5976. https://doi.org/10.1109/CVPR.2017.632

[6] Volodymyr Mnih et al. 2015. Human-level control through deep reinforcement learning. *Nature* 518, 7540 (2015), 529.

[7] W. Pei et al. 2017. Temporal attention-gated model for robust sequence classification. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 820–829. https://doi.org/10.1109/CVPR.2017.94

[8] Fereshteh Sadeghi and Sergey Levine. 2017. CAD2RL: Real single-image flight without a single real image. In *RSS XIII, Cambridge, Massachusetts, USA, July 12-16.*

[9] A. Singla, S. Padakandla, and S. Bhatnagar. 2019. Memory-Based Deep Reinforcement Learning for Obstacle Avoidance in UAV With Limited Environment Knowledge. *IEEE Transactions on Intelligent Transportation Systems* (2019), 1–12. https://doi.org/10.1109/TITS.2019.2954952