Project: Deep RL Arm Manipulation

Jd Fiscus

1 Abstract

In this project for Deep Reinforcement Arm Manipulation project there were two main objectives in order to complete the goal. These objectives were to create a Deep Q Learning (DQN)[1] agent, which utilizes reward functions(win, loss) in order to teach the robotic arm.

The objectives to achieve the end goal are:

- Have the robotic arm touch an object(green cylinder) with an accuracy of 90% in less than 100 runs
- Have the gripper on the robotic arm touch an object(green cylinder) with an accuracy of 80% in less than 100 runs

2 Reward Functions

Located in the ArmPlugin.app file are the reward functions. For both objectives the same arm joint position control was used which is to increase or decrease the joint position. This robot had two joint actions. Also for both objectives the rewards were the same; REWARD_WIN = 0.1 and REWARD_LOSS = 0.1.

In both objectives the REWARD_LOSS would be given if any of the following factors occurred:

- Robot arm touched the ground
- Robot arm did complete within a time to live(TTL)

For objective 1 if the robotic arm successfully hit the green cylinder it would receive a reward of REWARD_WIN * 5.

However, in objective 2 where the gripper had to hit the green cylinder there needs to be an additional check before giving a reward. After much trial and error there had to be both a REWARD_WIN and REWARD_LOSS on that conditional. Ultimately to achieve this objective REWARD_WIN * 20 a successful gripper touch and REWARD_LOSS * 5 for not.

[1] "Q-learning - Wikipedia." https://en.wikipedia.org/wiki/Q-learning. Accessed 29 May. 2018.

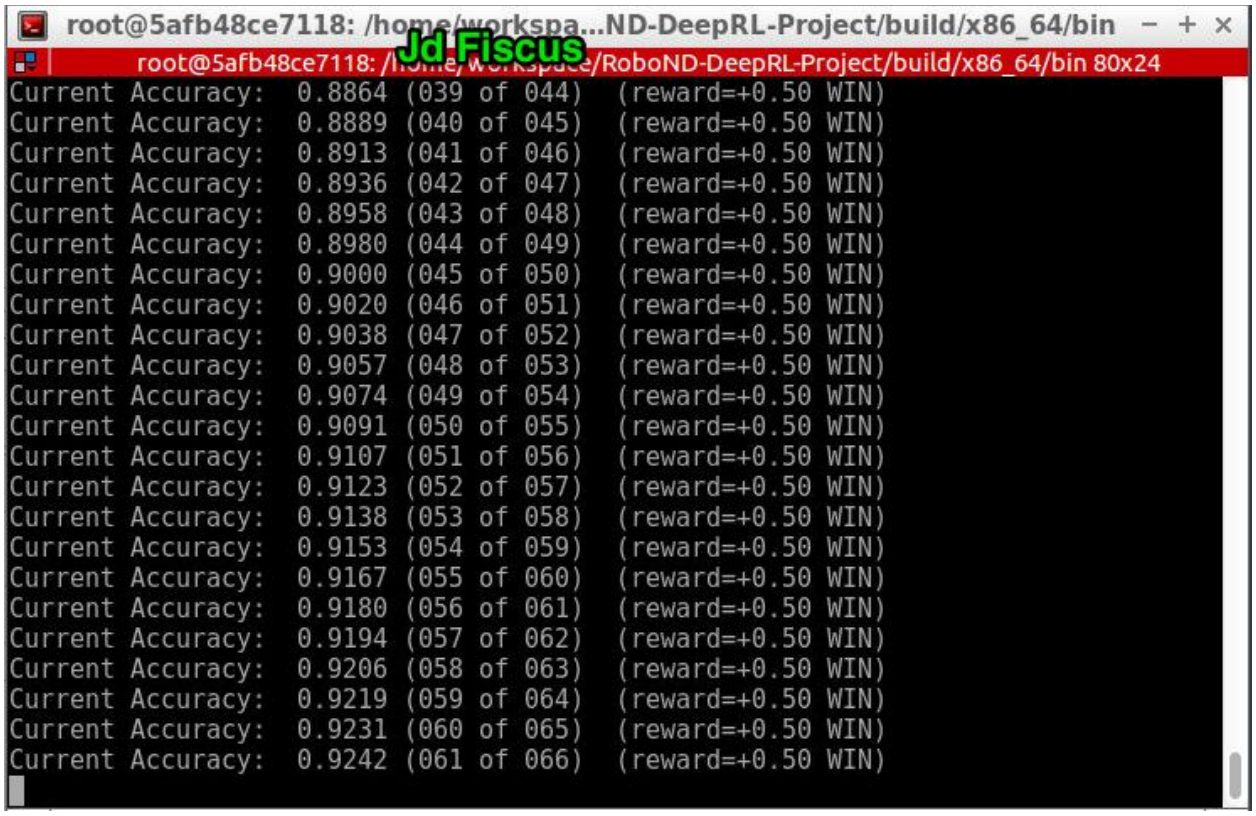
3 Hyper Parameters

It seemed that for both objectives the same hyper parameters were sufficient. Input width and height were set to 64x64. The learning rate was 0.01, a higher replay memory of 20000 was set, and the batch size increased to 512, all helped with learning due to the more complex task.

4 Results

Objective 1

In this objective the goal is to have the robotic arm touch an object(green cylinder) with an accuracy of 90% in less than 100 runs. Below are the successful results:

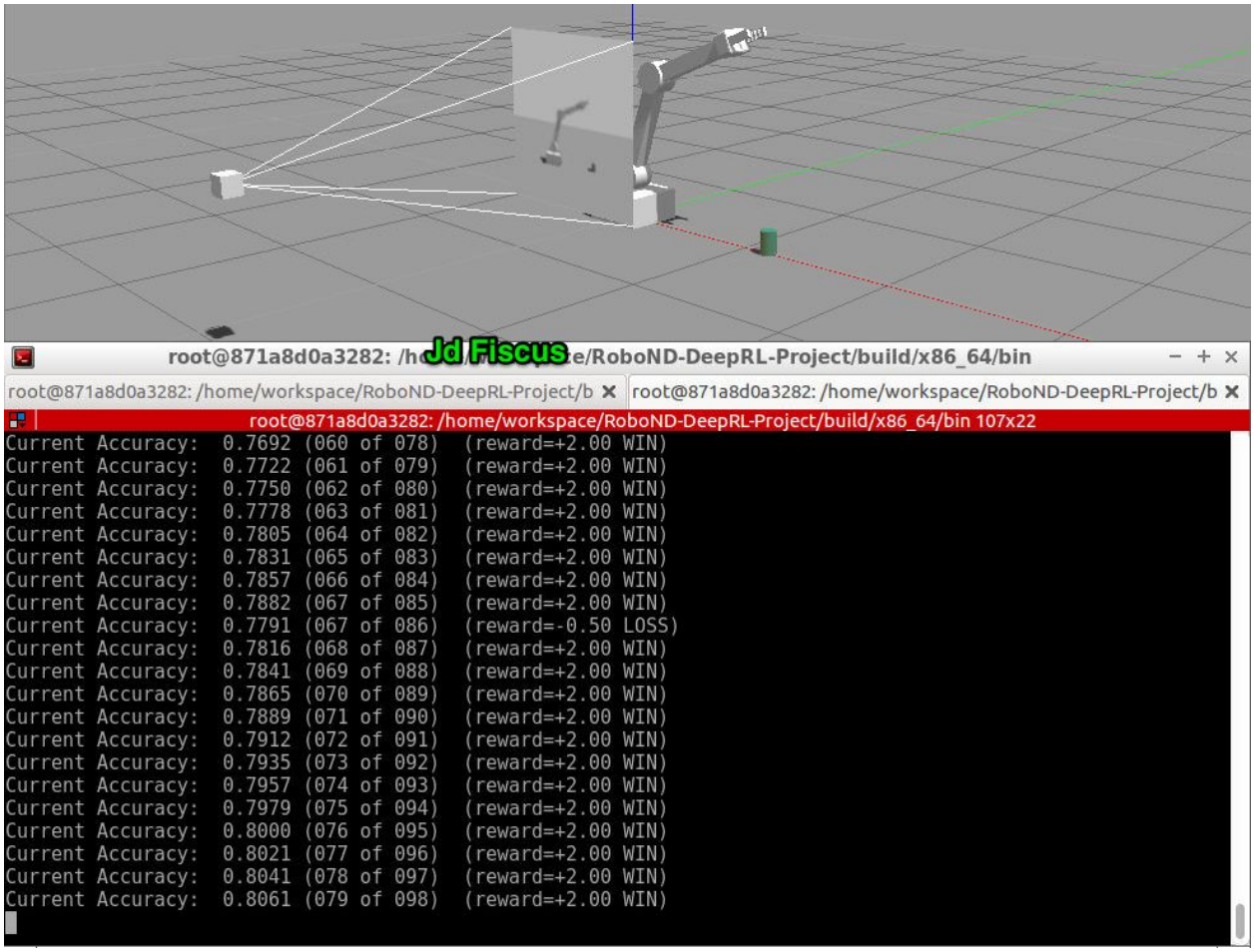


Video

For this objective the robotic arm seemed to learn quick and there were very little loss rewards. However on the times it did fail it was either hitting the ground, or struggling in the TTL. Also there was a behavior were it would get a successful result but launch the cylinder off screen.

Objective 2

In the second objective the goal is to have the gripper on the robotic arm touch an object(green cylinder) with an accuracy of 80% in less than 100 runs.



This objective was more difficult to train, which seemed to take a lot of trial and error. The first issue that happened was the robotic arm seemed to consistently fail on TTL. Then it started to get some successful rewards. However, still it would run into problems hitting the cylinder with its arm by over extending or under extending and hitting the floor. Finally, it seemed to get to the objectives goal.

5 Future Work

There seems to be some fine tuning that could be done in order to increase the learning rate. This could possibly be done with better reward functions. Understanding the amount of pressure the joint is moving once the arm touching the cylinder could also help. Finally, the most repeatable issue was the fail with TTL. To fix TTL it would be beneficial to increase the live time.