Deep RL Arm Manipulation

Mohamed Sayed Antar

Abstract—create a DQN agent and define reward functions to teach a robotic arm to carry out two primary objectives

Index Terms—Neural Networks, DQN, Reinforcement Learning, DeepRL, Deep Neural Networks.

1 Introduction

By creating a DQN agent and define reward functions to teach a robotic arm to carry out two primary objectives:

- Have any part of the robot arm touch the object of interest, with at least a 90% accuracy.
- Have only the gripper base of the robot arm touch the object, with at least a 80% accuracy.

2 Reward Functions

for sure the most important part to meet the objectives for this project is the rewards part For both objectives,almost the same reward function was used in the two objectives except the REWARD_WIN part in the first objective we checked for the state of the collision in case it happened between the object and any part of the arm, but in the second objective we give the reward in case only the gripper collide with the object.

another win reward for the network which depending on the distance between the gripper and the object it is the same in both cases.

for the REWARD_LOSS part the network punished with loss when the arm collide with the ground not the object, the same for the second objective except that in the second case we give the network a penalty in case any other part of the arm collide with the object instead of the gripper itself.

in both objectives the REWARD_WIN was set to 2500, and the REWARD_LOSS was set to -250, in the first objective the winning reward was the double of the REWARD_WIN which equal 5000 and in the second case if the required collision is happened only the network will get the REWARD_WIN multiplied by 10 which is 25000 but if any other part of the arm not the gripper collide with the object the network take a negative number as a penalty for this.

2.1 hyperparameters

2.2 INPUT width and height

for the INPUT_WIDTH and INPUT_HEIGHT we took it as 64 by 64 to use less memory and lower computational power but in case of more complicated system then we will need more resolution for that.

2.3 ACTIONS

ACTIONS or number of actions which is the double of the number of the joints a they are 3 then 6 actions created to control them.

2.4 OPTIMIZER

it set to RMSprop then Adam and the best results achieved with Adam not RMSprop.

2.5 LEARNING_RATE

by trying several values the learning rate at 0.05 in the first objective give a very good results in case of high learning rate the arm will give good results in only the first few episodes, in case of the second objective the 0.01 and 0.02 give a good results in this case.

2.6 REPLAY_MEMORY

this value left as it is from the initial setting.

2.7 BATCH_SIZE

by trail and error this value gave the best results when 128 not 16 or 32.

2.8 USE_LSTM

by setting the LSTM to true as previous lessons explain

2.9 LSTM_SIZE

by trail and error this value gave the best results when 256 not 32 or 64.

2.10 ALPHA

by setting alpha to several values like 0.1, 0.3, 0.7 and even 0.9 the best results when it was 0.3.

2.11 velocity

by reducing the velocity limits by half to help in collide with the right part.

3 Results

3.1 first objective

for the first objective after trying several times as the arm was going crazy some times eventually it became stable with the previous settings and reached to a good accuracy as the figure shows.

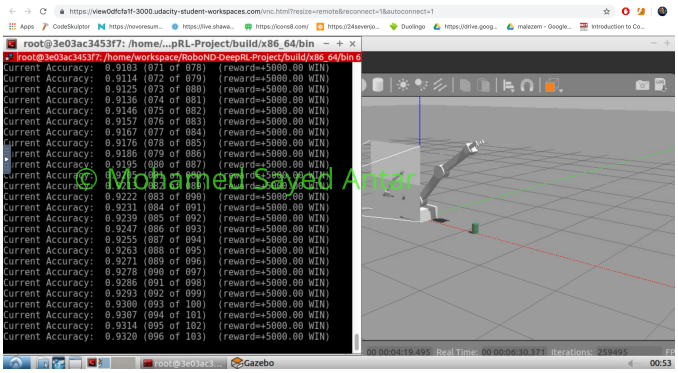

Fig. 1. objective 1 ”the Arm”

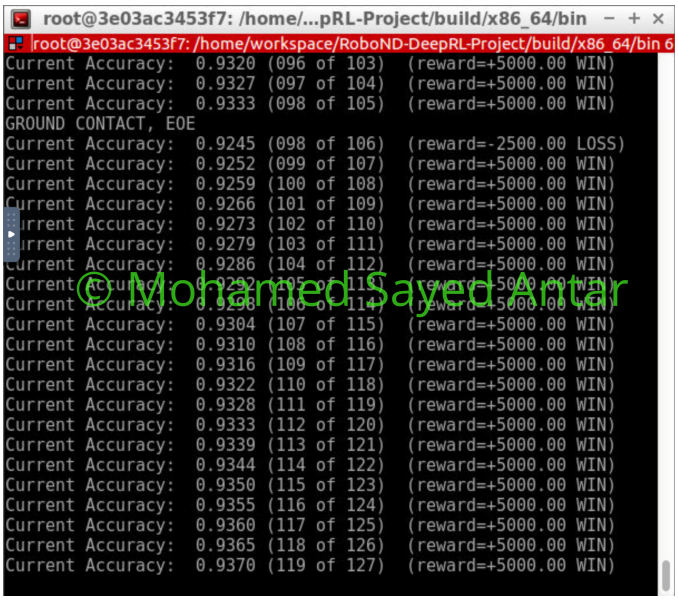

Fig. 2. objective 1 ”the Arm”

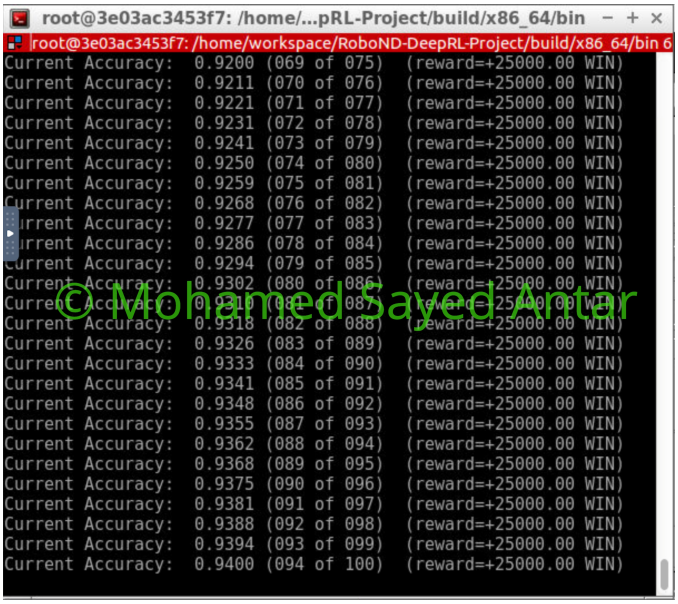

Fig. 3. objective 2”the Gripper”

3.2 second objective

like the first objective after trying several times with the rewards and hyperparameters eventually the required accuracy was reached as the figure shows.

4 FUTURE WORK

spending more time setting the hyperparameters will lead to better accuracy as happened, also the rewards make huge changes to the model so improving it will lead to better results.

some times at the start the robot don’t touch the object it just go closer to it and eventually it collide with the ground or even the frames end the episode, by spending more time on this problem will lead to understand the project well and avoid such a problem in the future.