Istanbul Technical University
Faculty of Computer and Informatics
Computer Engineering Department

BLG 435E
Artificial Intelligence
Homework III

Uğur Uysal - 150140012

Jan 2nd, 2019

Contents

1 Literature Review

Reinforcement Learning (RL) is a sub field of Machine Learning where an agent learns by **interacting** with its environment, observing the results of these interactions and receiving a reward (positive or negative) accordingly. This way of learning mimics the fundamental way in which we humans (and animals alike) learn [1]. In robotics, interacting with environment can not be avoided and reinforcement learning is useful other methods when there are interactions unlike supervised algorithms for standard classification tasks. Kober stated that "Reinforcement learning enables a robot to autonomously discover an optimal behavior through trial-and-error interactions with its environment."[2]. The key part of the reinforcement learning is trial-error method, it is human-like learning. Furthermore recent developments at deep learning makes reinforcement learning more powerful, complex value functions and policy gradients with neural networks outperformed human-level at some tasks such as chess[4], DOTA2(video game categorized as Multiplayer Online Battle Arena - MOBA) [5].

1.1 Problem - FetchPush

From Open AI Gym, FetchPush problem is chosen. The problem is pushing an object(a box) to desired location with an robotic arm [3]. I picked two articles that works on this problem from different perspectives. One of them [5] puts random obstacles to environment and compare their methods and other methods. It uses active vision where the viewpoint of camera changes for better views of environment. There is nice analogy, the humans moves their head get better view and focus then why robots should not do that? Moreover when there are distractors in environment, observing environment dynamically is highly beneficial. But the proposed methods are not successful enough under occlusions compared to environments without distractors.

Second method that I inspected did not focus FetchPush problem directly. They used "Deep Deterministic Policy Gradients" and "Hindsight Experience Replay (RL method specifically dealing with sparse rewards)" for several tasks one of them is FetchPush problem. But the article focuses on improving the agent make learning more robust. The RL algorithms have episodes which are trials that we learn from. They [7] proposed a new technique called "Experience Ranking" which to decide for episode whether it should be stored and replayed by using CNNs.

For same task, these two methods addresses different problems, one of them focuses on distractors in environment and other one focuses on making learning efficient and robust. The first article employed the active vision for solving the problem but they are not success enough when compared to agent that learns distractor-free environment. Second article focuses on improving RL on this task and they are successful about decide an episode whether it is worth being stored for future learning, they made this judgments with pre-trained neural nets.

2 Model

In general, RL algorithms have states, actions and rewards. For this problem, state holds the information of robotic grasp(location, velocity -needed for push), location of box, velocity of box, location of target point, and the actions are change location of grasp and push(move forward in some direction) and the reward is -1 for time slices and +1000 for when object is at the right place.

2.1 Input - Output - Algorithm

The input of method is raw pixels we since observe the world through a camera in abstract output is an action. We can utilize deep neural networks for detecting objects in image. For policies, we can use Deep Q Learning too. "Hindsight Experience Replay" HER should be used to avoid learning mistakes by spare rewarding.

3 Advantages and Disadvantages

The advantages of my model is that it is simple(Ocham's razor). A disadvantage is sparse rewarding(we only get positive reward at the end of the task, otherwise agent get always negative signal) that can be problem but using HER can solve this problem. Also this model and algorithm for only learning one simple task, if we want to imitate human-learning we need more general algorithms. Training neural networks for days and getting only one robot on simulator to push boxes is not really considerable success for humanity, we need to transfer the knowledge to different domains.

References

[1] RL

[2] Reinforcement learning in robotics: A survey

[3] Open AI Gym for Robotics

[4] Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm

[5] OpenAI 5V5 Dota2 Bots

[6] Reinforcement Learning of Active Vision for Manipulating Objects under Occlusions

[7] Deep Learning with Experience Ranking Convolutional Neural Network for Robot Manipulator