Final Project Proposal

Alex Day

March 23, 2021

1 Introduction

The issue of efficient and safe movement is both a well-studied and yet unsolved problem in robotics. This issue can be generally modeled with a set of agents where each agent goes through alternate phases of observing and selecting new velocities. Ideally these new velocities should bring the agents closer to the goal while avoiding collisions with static obstacles and other agents.

There are numerous ways that agents can sense their environment, however, by far the most common include LIDAR scans, where beams of light are projected around the agent and the distance to the closest object is returned based on the reflection, and RGB-D images where depth information is concatenated with regular color images in order to give a 3D representation of the environment around the agent. RGB-D images can also be separated into the depth map and color components of the image.

This project will apply reinforcement learning to multiagent collision avoidance by training an agent to reach a goal given only a depth map and a goal displacement through an environment with other agents and static obstacles.

2 Related Work

There has been some focus on learning collision free movement in static environments using depth images and reinforcement learning in the past. Wu *et al.* [1] used a two-stage noisy DDQN with the previous four depth maps as imput and eight discretized actions (three linear and five angular) as the output, Cimurs *et al.* [2] used a Convolutional Deep Deterministic Policy Gradient with both the last ten depth images as well as the relative goal position as input and a continuous linear and angular action as output, and Wu *et al.* [3] used a modified DDQN with four depth images as input and outputs Q-values over two separate sets of actions letting the agent select a linear and angular velocity simultaneously. It should be noted that the three previous works are all end-to-end methods.

Most deep reinforcement learning in multiagent scenarios is using other sensors, namely LIDAR. For example, Long *et al.* [4] used Proximal Policy Optimization with history of lidar scans combined with a relative goal position to output a preferred velocity that was then provided to an existing collision avoidance method as the goal velocity.

3 Proposal

I propose a combination of previous work to train an agent to **traverse to a goal in a multiagent scenario using depth images and relative goal position** hoping that the results from [2, 4] hold and that the agents will be able to both generalize to unseen scenarios and learn a more complex understanding of the obstacles based on the 2D images.

3.1 Environment

The environment I am using is an extension of the MiniWorld[1] environment. The environment is comprised of a large open room with agents which use the ORCA [5] collision avoidance algorithm and one RL agent. These ORCA agents will have perfect knowledge of both the position and velocity of the RL agent whereas the RL agent will only be able to observe a depth map image from a front-facing sensor and a relative goal position. All proportions of the agents, including radius, height, and camera position, are based on a turtlebot[2] equiped with a Microsoft Kinect V2[3].

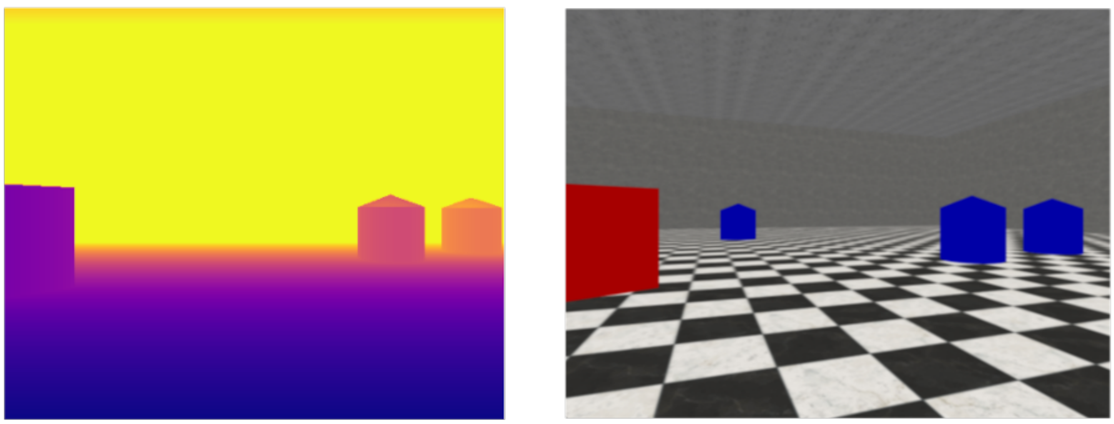

Figure 1: A scene from the environment captured as both a depth map and in color

3.2 Deep Reinforcement Learning Techniques

My work will closely mimic the work done by [4] in that I will be using a shared experience buffer between multiple robots and train them using an Actor-Critic/PPO approach. I will also be following their task simplification [6] method by beginning training with no objects in the environments and randomizing both the starting and ending positions for a small number of agents. Once the network has learned this simple task, obstacles will be introduced to make the environment more complex.

References

[1] K. Wu, M. A. Esfahani, S. Yuan, and H. Wang, "Depth-based obstacle avoidance through deep reinforcement learning," in *Proceedings of the 5th International Conference on Mechatronics and*

[1] `https://github.com/maximecb/gym-miniworld`
[2] `https://clearpathrobotics.com/turtlebot-2-open-source-robot/`
[3] `https://docs.depthkit.tv/docs/kinect-for-windows-v2`

Robotics Engineering, 2019, pp. 102–106.

[2] R. Cimurs, J. H. Lee, and I. H. Suh, “Goal-oriented obstacle avoidance with deep reinforcement learning in continuous action space,” *Electronics*, vol. 9, no. 3, p. 411, 2020.

[3] K. Wu, M. Abolfazli Esfahani, S. Yuan, and H. Wang, “Learn to steer through deep reinforcement learning,” *Sensors*, vol. 18, no. 11, p. 3650, 2018.

[4] P. Long, T. Fanl, X. Liao, W. Liu, H. Zhang, and J. Pan, “Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 6252–6259.

[5] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, “Optimal reciprocal collision avoidance for multi-agent navigation,” in *Proc. of the IEEE International Conference on Robotics and Automation, Anchorage (AK), USA*, 2010.

[6] M. Kerzel, H. B. Mohammadi, M. A. Zamani, and S. Wermter, “Accelerating deep continuous reinforcement learning through task simplification,” in *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2018, pp. 1–6.