

Practical Course Robotics

Kanishk Navale
3437531

Olga Klimashevskaya
3525388

30th November 2021

1 Objective

The project demonstrates how to teach a robot to perform teleoperation tasks using model-free trajectory planning, i.e., without explicit inverse kinematics.

One of the teleoperation tasks can be smart sorting, i.e., objects of predefined shape and/or colour have to be placed on correct tiles (i.e., coloured squared space on the scene floor), whereas objects do not necessarily have to be of the same size. More specifically, a yellow brick has to be placed on a red tile, and a green cylinder has to be placed on a blue tile.

During teaching session, a certain number of example objects are situated in the visible for the robot area where the marked goal-tiles are identifiable as well. These objects represent a so-called learning pile. The robot attempts to approach an object, grasp it, move to a tile and place it there. After each step in this action sequence, the robot receives an automatic signal that confirms or rejects their choice which enables the robot to learn from their experience.

Object by object, the robot learns the correct mapping of the object type to the tile type. After the learning pile is sorted, the robot is supposed to be ready to perform the sorting task on the testing pile. Hence, we can see again a new set of example objects together with the corresponding empty tiles for the testing stage. During this stage, the robot performs the final task of sorting after which the final accuracy score is displayed to the user.

2 Work Plan

The project will be implemented in the RAI environment using algorithms of Reinforcement Learning (RL).

2.1 RL Methodology

In this setup, the states s_t are represented by end effector's locations, i.e.,

$$s_t = (\textit{Current Goal}, \textit{Achieved Goal}, \textit{Desired Goal}),$$

and the actions a_t include

$$a_t = (\textit{Control Signal}, \textit{Grasp}),$$

where *Control Signal* can include *Velocity*, *Acceleration* etc.

Below we provide a preliminary RL pipeline with the approximate time-for-deployment distribution:

1. (5%) Define the RL environment.
2. (45%) Move the robot's end effector from point A and to point B via linear path planning using Dense/ Sparse rewards.
3. (40%) Extend the RL model for the task of grasping/releasing the object.
4. (10%) Use the RL model to combine steps (2) and (3) to move the robot with the grasped object to the tile position and drop the object.

2.2 Project Scope

While keeping the team work split as even as possible, i.e., Kanishk ($\approx 50\%$) and Olga ($\approx 50\%$), we will approach this project in the following sequence of steps with the approximate time-for-deployment distribution:

1. (35%, in December) Build the RAI/ OpenAI/ ROS environment to give rewards on *Desired Goal* state. Here we might use some perception for object identification.
2. (15%, in December) Program an RL algorithm to collect the data from the environment.
3. (50%, in January) Prove the algorithm for the robot going from point A to point B, in order to make it model-free.

We shall keep our code in the github repository where one can also track our workflow.