

UR5 Obstacle-Avoidance Pick-and-Place with Arm Assistance
Weekly Progress Report 3
Yucheng Kang, Zhiyi Ren
April 24, 2018

- This Week's Goals
 - Implement customized planner with intermediate pose input and core functionalities from OMPL.
 - Integrate planner with moveit and octomap.
- This Week's Progress
 - Moveit planning and control
 - UR5 now can follow arm motion, with a frequency of about 1 Hz. Since we add gripper, camera mount, and table in the planning scene, moveit constantly plans small motion for UR5.

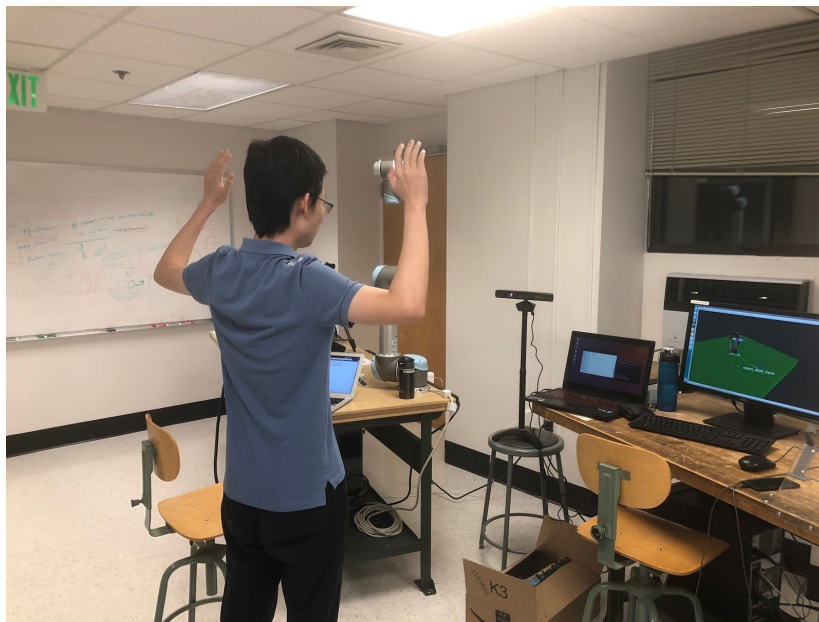


Fig. 1 Kinect detecting arm



Fig. 2 UR5 following human arm motion

- Integrate octomap into Moveit
 - We followed the tutorial to configure realsense and octomap in moveit. Planning now takes the obstacles into account.

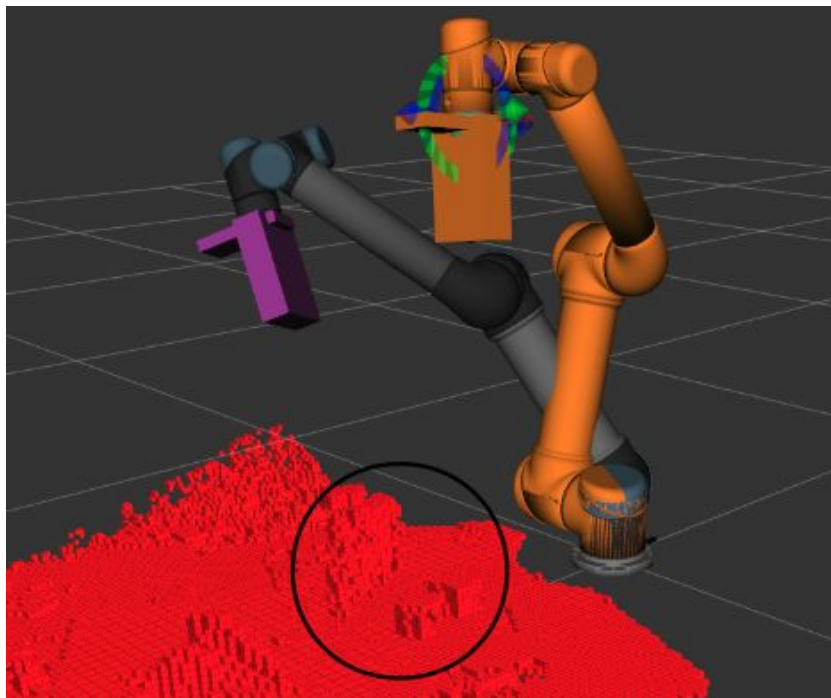


Fig. 2 Octomap with Moveit in RViz

- Hand-eye calibration
 - We tested the hand-eye calibration and used the transform from ee_link to camera_link for further integration.
- Changes in Project Scope/Goals
 - We did not finish implementing intermediate pose input for moveit planning this week. We plan to finish this next week. We will be likely to use cartesian path in Moveit.
 - We want to implement detecting hand orientation and gesture using kinect. Hand orientation will indicate the end-effector orientation, so the camera can see obstacles from multiple views. Gesture is used to indicate intermediate pose.
- Lessons learned
 - Simulating motion of UR5 in Rviz usually takes longer time than what actual UR5 takes.
 - The aruco-hand-eye package from JHU LCSR uses YAW-PITCH-ROLL convention instead of ROLL-PITCH-YAW.
 - Real-time updating octomap in Rviz is very CPU/RAM-consuming.
- Next week's goals
 - Implement intermediate pose input for moveit planning
 - Implement hand orientation and gesture detection.
 - Integrate arm tracking, and motion planning