UR5 Obstacle-Avoidance Pick-and-Place with Arm Assistance
Weekly Progress Report 2
Yucheng Kang, Zhiyi Ren
April 17, 2018

- This Week's Goals
  - Implement AR tag tracking and calibration with R200.
  - Implement visual servoing with R200.
- This Week's Progress
  - Hand-eye calibration
    - This is based on *visp_hand2eye_calibration*. It only solves hand-to-eye transformation with given data. The whole process is done by *aruco_hand_eye*.
    - The code is finished but we have not tested yet.
  - Converting point cloud to octomap
    - This is actually in 3rd week's plan.
    - This package is based on *realsense_camera* and *octomap_mapping*.
    - As the package *realsense_camera* is generating point cloud, we need to write a new package *realsense_mapping* to convert the point cloud (see Fig. 1) to octomap (see Fig. 2), which will be fed to moveit planning scene in the future.

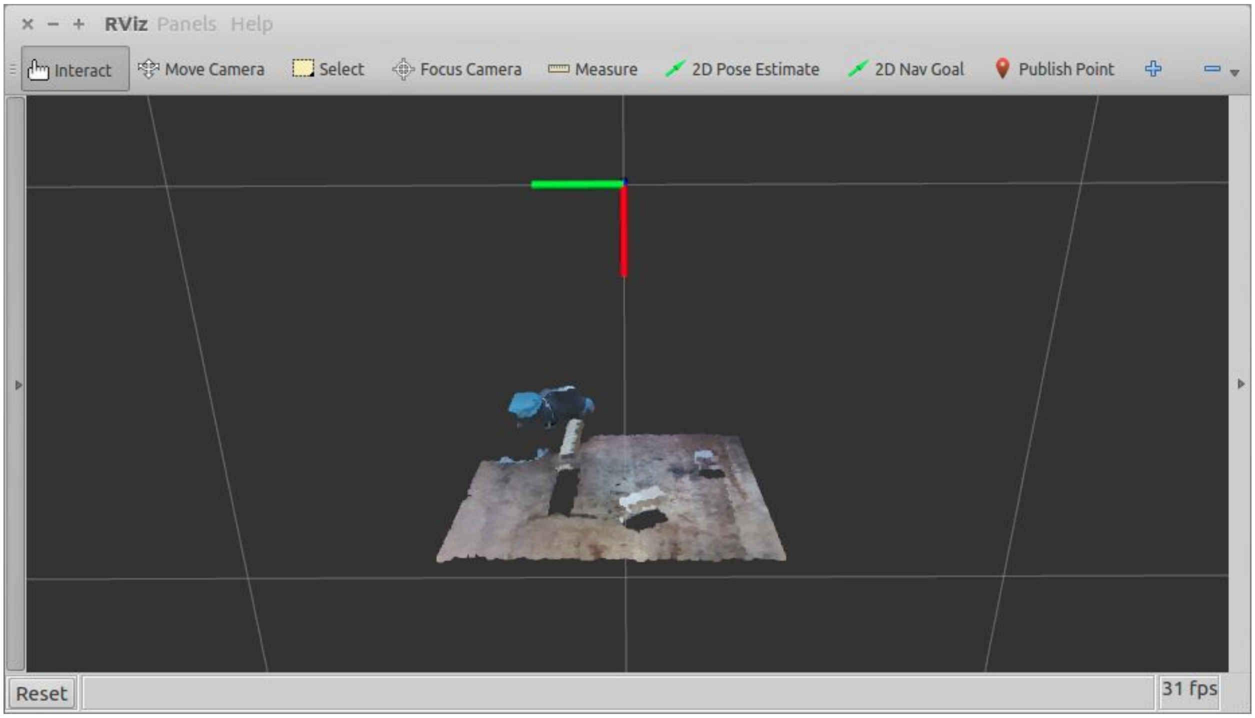

Fig. 1 Point cloud data in RViz

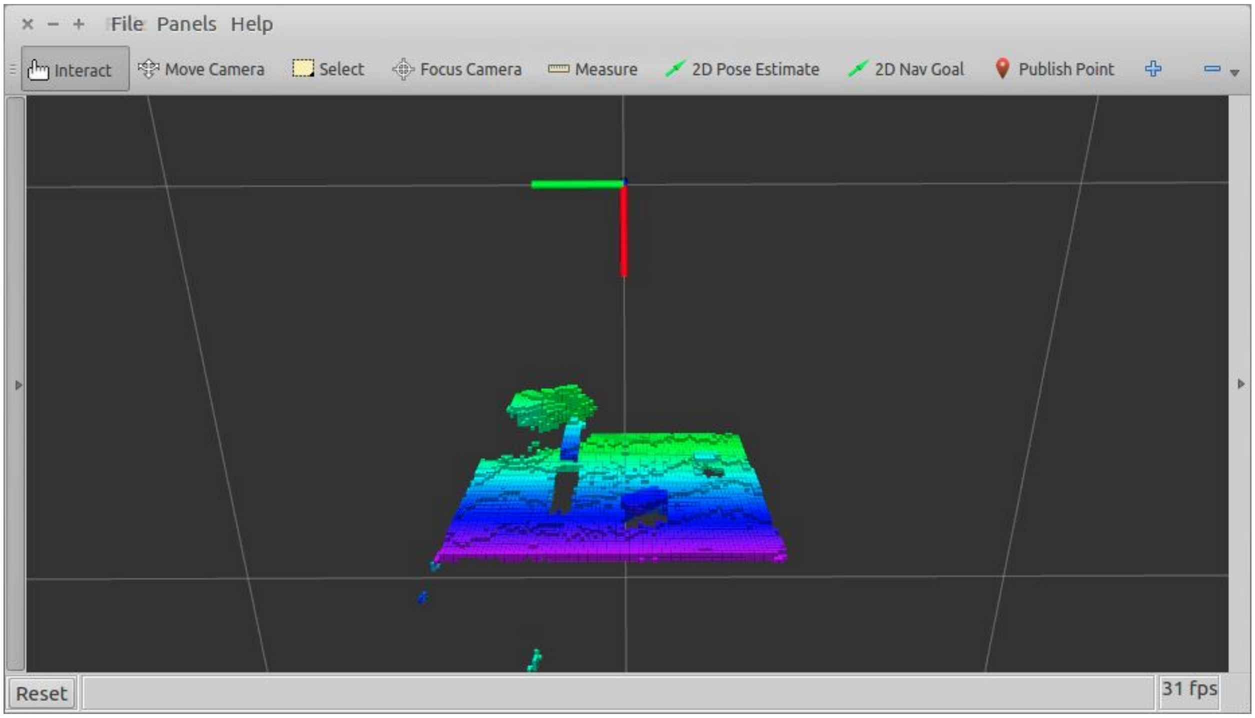

Fig. 2 Octomap data in RViz

- Visual servoing
  - We have gone through the tutorial of the ViSP (Visual Servoing Platform) stack, but we notice that it requires clicks on image for initialization, which can be very tedious in our scenario.
  - Hence, we postpone this part and plan to use AR tag for recognizing objects for now.
- Moveit planning and control
  - We spent some time understanding the whole framework of moveit, especially the motion planning part. We were able to add the camera mount, gripper, and the table into the planning scene via API, and performed planning. We limited the joint angle of base to be small, and limited the joint angle of elbow link to be negative, so the robot keeps an elbow-down pose.
- Central node
  - We implemented a node that serves as the central “controller” of the system. It needs to further implemented.

- Changes in Project Scope/Goals
  - No changes.
- Lessons learned
  - Use rqt_launchtree to analyze large launch files. It helped us better understand the structure of moveit framework.
  - Use rqt more for debugging

    - R200 has a significant minimal range of ~0.5m. When it is near the object, there will be no valid depth data.
    - Use apt-get to save time. We had dependencies problems in building ViSP from source (different parts depend on different versions of OpenCV, missing some files...) and we spent a lot of time on this. Thanks to sudo privilege.
- Next week's goals
    - Implement customized planner with intermediate pose input and core functionalities from OMPL.
    - Integrate planner with moveit and octomap.