

INDIVIDUAL PROGRESS REPORT

Lekha Walajapet Mohan

Team D: HARP

Teammates: Alex Brinkman, Rick Shanor, Abhishek Bhatia,
Feroze Naina

ILR03

Oct. 30, 2015

I. Individual Progress:

In this progress report, I have discussed my progress with respect to my project for the past week. I started with familiarizing myself with Point Cloud Library, which is an open project for 2D/3D image and point cloud data processing (source: www.pointclouds.org). PCL is most preferred library for developers working on cloud data, given an extensive documentation and application packages for beginners. But, example code or algorithm for different modular libraries is very limited which limits our scope for comparison between various other algorithms, especially when users are beginners in C++. I got myself familiarized with PCL, as most of our base code will be based on point cloud data. With help of documentation and modular examples given, I was able to implement object recognition and pose estimation on a generic data set given in the library (Fig 1).

One of the biggest problems I faced was that I was not able to implement my algorithm in the APC Data Set. I have given an analysis of the reason in II.Challenges in the following. In order to experiment different image file types such as .obj, .ply etc., which have to be represented in mesh model and then converted to .pcd(point cloud) files, I familiarized myself with a software called the MeshLab. MeshLab is an open source software that processes 3D triangular meshes. But, again it was not able to export image file in the form of .pcd. I was working on converting an .obj file to .xyz and then converting it to .pcd file. The biggest issue was that there aren't code examples given for the above conversion and hence, it took a lot of time to debug the code snippets given online and implement them for our use case.

Algorithm:

1. Start
2. Load point cloud files
3. Setup resolution invariance
 - 3.1 input sampling size
 - 3.2 model resolution
 - 3.3 input descriptor radius
4. Compute Normals
5. Downsample clouds to extract Keypoints
6. Compute descriptor for keypoints
7. Find model tree correspondences with KdTree
8. Find neighbor descriptor and add it to correspondences vector
9. Clustering using geometric consistency

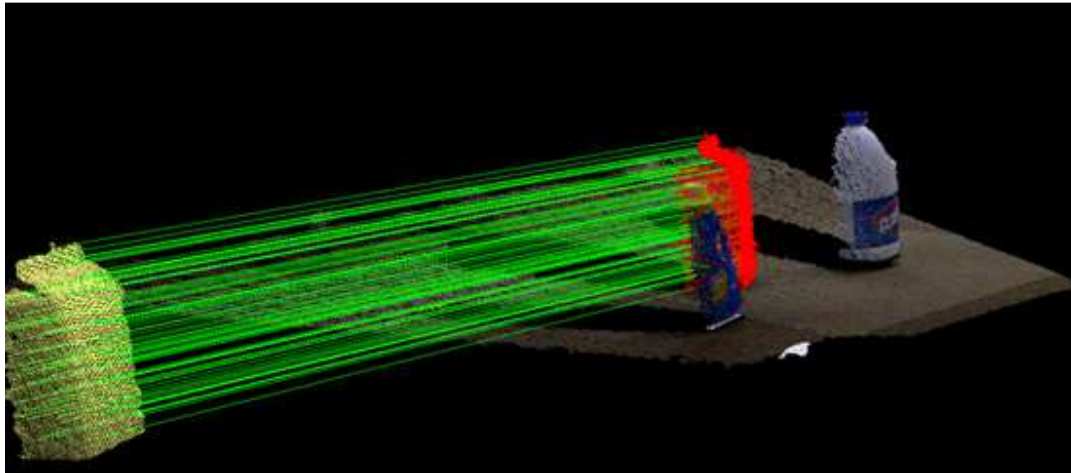


Fig 1. Object estimation and correspondence matching

II. Challenges

Our first challenge was that our official sponsor Mr. Jim Osborn is no more interested in sponsoring our project as he was more keen in the business case of the pantry bot. On learning that our business teams were different with unique use cases, he told us he wouldn't be interested in sponsoring anymore with us. Though that did not affect the performance of our project, it did make us think through of effects if any. Professor Maxim Likachev will now be our sponsor. Even if we fail to prove our eligibility to represent CMU in Amazon Picking Challenge, Maxim will still lend his PR2 for MRSD project demonstration.

I faced many challenges the past week working on perception subsystem. With respect to Point Cloud Library, I was able to find only one example. Owing to my knowledge constraint in C++, I was not able to write one myself. This disabled me in testing various other algorithms. I am working on my C++ coding and I should be able to overcome this challenge for the spring.

Alex found it difficult to implement the SMACH controller, owing to limited knowledge about it in our team. Rick found it difficult to implement color segmentation algorithm owing to limited examples available. He implemented his own algorithm, but we are aware that it may not be the best one. Feroze and Alex had problems setting up the system to get the Kinect-2 working. We haven't succeeded in getting kinect-2 data yet, but we will be working on getting it resolved. Abhishek Bhatia was under the guidance of our technical advisor for tele-operation of PR2 and hence, he didn't face challenges as such. But, it did take some time to familiarize himself with using PR2 in PS3 controller.

III Team Work

Alex was working on SMACH controller setting up the state machines. SMACH is a state machine controller built in using Python libraries. Alex also fabricated the gripper for PR2 arm with which we did our suction demo. Rick was working on designing the vacuum suction gripper, experimenting the pressure conditions and grasp capabilities for objects of varying sizes and weights. Rick was also working on image segmentation using color and depth data, where combining the image segments would predict item based on neighbor approximation(Fig 2). Abhishek was working on familiarizing himself with turtle sim and tele-operation of PR2. We all did a suction demo gripper to analyze how it worked. We attached a gripper to the PR2 and a suction cup to the vacuum cleaner, tested it with objects of various sizes and weights. We took interesting notes on shear strength, the success at grasping objects lying at different orientations. Feroze was working on PR2 arm planner which he simulated in Gazebo and showed it running in Arvis, applying inverse kinematics at XYZ positions.

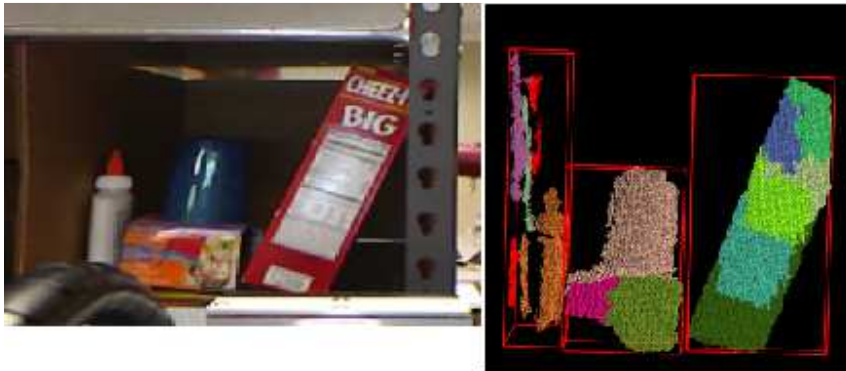


Fig 2. Color segmentation using RGB-D data
Picture Courtesy: Rick Shanor

III Future Plans

Alex will be working on prototyping suction gripper focus on it's build quality. Rick will be working on impeller system and imparting shelf content knowledge into color based clustering. Feroze and Abhishek will be working on electronics design for the gripper. Feroze will also be working on commanding the arm inside the state controller and moving PR2 applying inverse kinematics. Abhishek will be programmatically commanding the base of PR2 and I will be working on designing algorithm to filter out shelf from depth data.

