Progress Review 8

Rick Shanor
Team D: Human Assistive Robotic Picker
Teammates: Alex Brinkman, Feroze Naina, Abhishek Bhatia, Lekha Mohan
ILR07
February 25, 2016

Individual Progress

Over the last two weeks, I have worked on the perception subsystem as well as the setup and installation of the UR5. For the perception subsystem, I first baselined the PERCH algorithm using our models. Next, I worked with Abhishek to write an RGB item identification algorithm that runs in parallel to the geometry based ICP methods. To setup the UR5, I had to modify the base to better support torsional loads of the arm during rotation. I also fabricated a new, lower profile end effector and Kinect camera mount. Finally, I worked with Alex and Abhishek to integrate the UR5 control software within our framework.

PERCH, Perception via Search for Multi-Object Recognition and Localization, is an algorithm developed in Carnegie Mellon's Search Based Planning Lab. This algorithm uses object models to render Kinect point clouds. These rendered images are used in a search tree to optimize a cost function to explain a scene and determine object pose, even under heavy occlusions. In order to test this algorithm in our system, I wrote a wrapper function which captures a Kinect frame and isolates the item point cloud. Figure 1 shows one example output of the PERCH pipeline. Perch is very accurate and also globally optimizes so no pre-segmentation is needed. However, the graph search currently takes a long time, even for a scene with three items. Also, the algorithm is only configured to handle 2D rotations. I plan to continue working with SBPL to see if we can optimize this algorithm for the picking task, in parallel with other own vision algorithm.

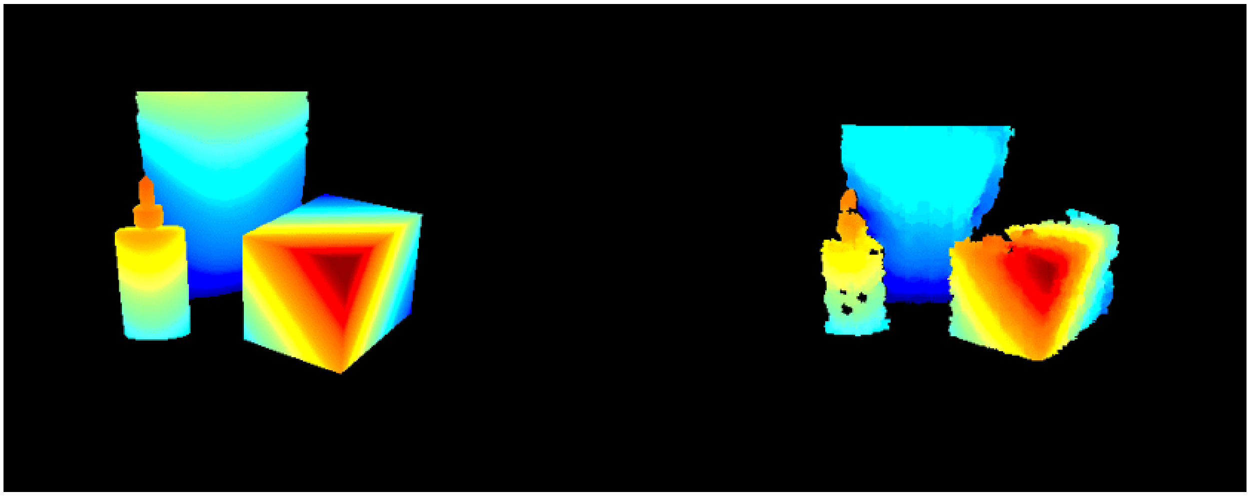

Figure 1: Initial PERCH results: rendered scene (right) and Kinect point cloud (left)

Abhiskek and I wrote a color-based identification algorithm. The algorithm works by comparing the Kinect color histogram with a database of collected histograms. First, the scene is converted to Lab color space, which is more robust to changes in lighting. Then, a histogram is generated for each of the L, a, and b color channels. The bin width for each color channel can be tuned to optimize performance. After this, identification is performed based on a k-nearest neighbors search. The algorithm outputs a matching percentage between each possible item and the scene. This output score is then combined with the geometry based ICP result. These two scores are combined, weighed differently for each item, in to identify the item on the shelf. Figure 2 shows two different histograms; one for the Cheeze-Its box and one for the Oreo box. As you can see, the distributions are similar for different images of each item but different from item to item.

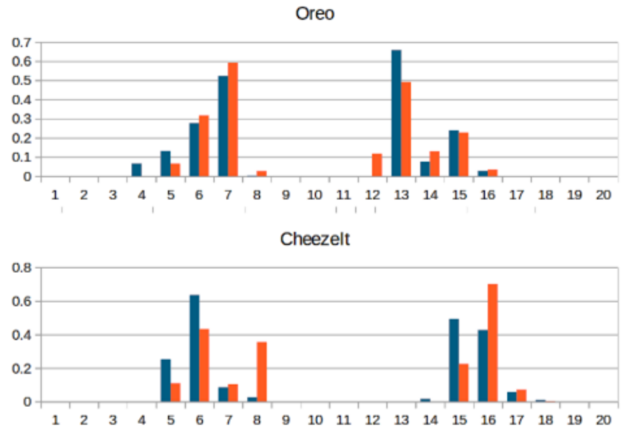

Figure 2: Color Histograms for 3 Items

Our team received the UR5 on Tuesday morning. Since then, we have been working to set up the system with our machine. When the UR5 first arrived, the base was not stiff enough in the torsional direction. In order to increase stiffness, I added plates connecting the four pillars, increasing the moment of inertia. As a temporary fix, I laser-cut these plates out of acrylic for fast fabrication and assembly. This fix worked pretty well; however, I have ordered metal to replace these plastic plates and increase stiffness even more.

Last PR, after noticing that our end-effector design inhibited the robot's collision free configuration space, I designed a more compact tool. This week, I fabricated this new piece, shown in figure X. The design was verified on the robot. So far, the robot has been able to reach all 9 bins of the shelf and there have been no self-collisions. This new tool can be seen mounted to the robot in figure 3 and 4.

Figure 3: New Kinect end-effector mount

After improving the base, I worked with Alex together to get our system pipeline working on the UR5. Along the way, we ran into many hurdles. First, our localization algorithm had to be tuned. Since the robot only has a reach of about .8m, it must be configured pretty close to the shelf. This means that we cannot see the entire shelf in one frame. By tuning the max search radius of the localization algorithm, we were able to localize while only viewing one row of the shelf. Eventually, the goal is to stitch together multiple images and then use the entire shelf for localization. Next, the motion planning algorithm was not producing good arm trajectories. Even moving to a point 10cm away, the arm flailed up and around.

By switching the MoveIt! Planner from an RRT algorithm to an RRT* algorithm, which optimizes based on path length, we were able to greatly improve arm trajectories. This is especially important such that we don't tangle our cords and vacuum hose during testing. Last night, we were able to semi-autonomously pick up one item off the shelf. We used the Kinect point cloud to set arm goal states, and the arm would go to the desired position. These movements were accurate enough to pick up items. The UR5, set up in from of the shelf, is shown in figure 4.

Figure 4: UR5 Integration

Challenges

As a team, the biggest challenge we faced was having to wait until Tuesday for the UR5. However, we were well prepared by the time it arrived and the initial setup was surprisingly painless. Personally, I have been struggling to generate ideas on how to improve the perception system. The RGB identification is a good start, but I plan to investigate additional techniques as well. These ideas are discussed in detail in the plans section.

Teamwork

As a team, we interviewed with the APC committee last Friday. In addition, over the last two weeks, Lekha and Feroze finished up the off-line grasping tool for labeling grasping surfaces on items. They were able to generate several preconditioned point clouds. In addition, Feroze got SimTrack working on his machine. He was able to create geometric models of a few of the 2015 items. This will be crucial if we qualify for the 2016 competition, as we will have to model all 40 new items. Abhishek worked closely with me in generating the RBG identification algorithms. I focused on the high level tasks of merging the geometry algorithm and managing the histogram data while Abhishek focused on the histogram generation and comparisons. Finally, Alex and Abhishek set up the UR5 controller to communicate with our machine. Alex worked with me to establish baseline localization. In addition, he worked on calibrating the Kinect extrinsic parameters.

Plans

As a team, we find out if qualify for the 2016 APC. This will dictate the direction we proceed over the next months. No matter what, I plan to look into machine learning algorithms that could better merge the histogram comparisons and the geometry based identification methods. In addition, I will work on shelf filtering and vision procession with the Kinect mounted to the robot such that we can autonomously grab one item for next PR.