Lekha Walajapet Mohan

Team D: HARP

Teammates: Alex Brinkman, Rick Shanor,Abhishek Bhatia,

Feroze Naina

ILR08

Feb. 24, 2016

I. Individual Progress:

For Progress Review 9, I was working on devising the grasp models for the APC list of item dictionary(2015). The grasping approach that we currently deploy has two aspects. One is to generate an online grasping model and the second method is to generate an offline model. For offline grasp generation, we manually load the ground truth point cloud data of every item, click a point cloud manually(Fig 1a) and store the clicked points in a separate cloud. Our algorithm works in such a way that, once a point is clicked, a sphere of a given radius is constructed and the points that fall within the sphere enclosure are pushed into a new cloud. We then merge the grasp model cloud and the ground truth data to visualize the estimated grasping surface(Fig 1b). We click the points which we feel are good grasping estimates for the end-effector to reach out to the item.

Algorithm for online grasp generation is:

- Point cloud information is captured by the kinect V2 camera and the shelf content is filtered out.
- Euclidean clustering and segmentation algorithm is implemented to detect the individual items
- The ground truth model of the target item is loaded and the detected point cloud cluster of the captured data is registered using Iterative Closest Point algorithm.
- Normals for the given point cloud is estimated.
- The manually generated offline point cloud data for the target item is loaded for estimating the grasping surface for the target item.
- For every point in the point cloud, the offline grasp model is matched to evaluate the possible grasping surface area available for the end-effector to grasp the item from plausible surface

Once the point cloud is read, the points are down sampled to estimate the normals from the down sampled point cloud. By modeling the planar co-efficients, the planar inliers can be obtained from the point cloud. I modeled the grasping surface area for items of various shapes, like the glue bottle(cylindrical), stir straw box(rectangle), tennis ball(flat surface with a sphere in the middle). The primary concern was to avoid the corners as it would be risky for the vacuum suction gripper to grasp from the edges. The normals estimated were very hard to visualize while generating the point cloud as the data was in a 3D form. 3D visualization might be ambiguous for the human eye to figure out which point are we exactly clicking at.

My next step was to design an automated process for the manual grasp estimation. On discussing this with our sponsor Professor.Maxim Likachev and our technical advisor Mr.Venkat, they strongly appreciated the idea of generating manual grasp models carefully. Although our first try at generating these models seemed to be a good start, we fully realize that we need to devise an automated and more accurate algorithm to generate the point cloud models.

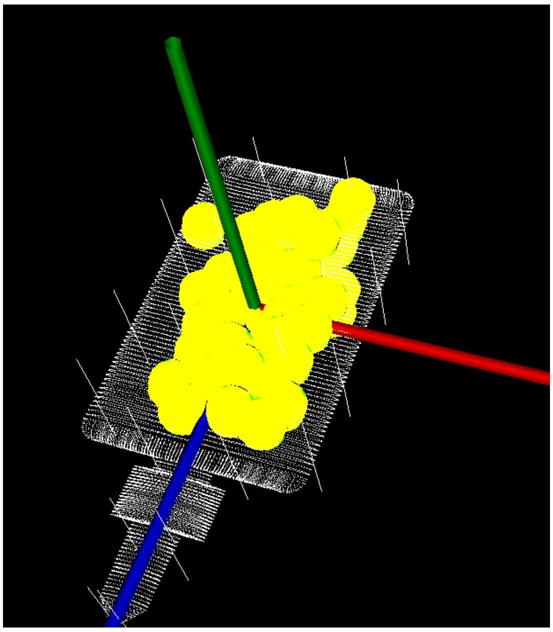

Fig 1a
Manually choosing the cloud points

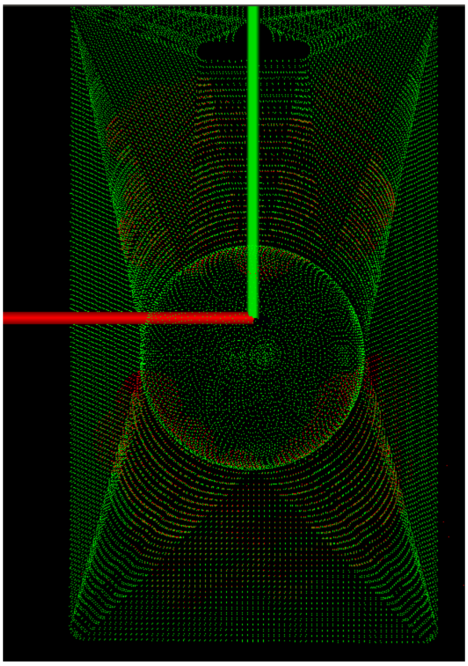

Fig 1b
Grasp model cloud(red points)

II **Challenges**

Biggest challenge that I faced for this week was to manually estimate the grasping surface. I used a keyboard call back function, which upon pressing a combination of key board buttons, would construct a sphere and push the points into a new cloud. The problem was that I was not able to judge the position of the points. Only when we visualize the cloud in multiple views, did I realize that my manual clicks had been inaccurate.

As it is, the generation of grasp models is a tedious and time consuming task and we can not afford to have wrong estimates. Hence, we realized the need to have an automated algorithm, where in, as I move the keys in the keyboard, a cube navigates through a cloud and upon user's approval, those points in that cube get pushed into a new cloud, defining the grasping area. Although simulation estimates a great grasping surface position, the real time robot might have some latency. The dataset is yet to be released by Amazon Picking Challenge officials, which is another big challenge. If the shapes or sizes are too complicated, we would have to design an algorithm in lesser duration of available time.

III **Team Work**

As we have received our new UR5 arm, we have refined our goal plans to speed up our integration of individual subsystems. Alex Brinkman was working on extrinsic calibration for the UR5 arm. He was also involved in building a stable platform for the UR5 arm. Rick and

Abhishek were working on detection of objects using RGB data. They have been working on generating color histogram of objects obtained under various orientations using generated test images. Using the generated histograms, they were able to detect the objects using neighborhood clustering of similar histogram. Feroze was working on getting the SIMTRACK simulation for APC dictionary items. SIMTRACK uses test images of an object, stitches them together to form a 3D model. I, as explained above, was working on generating the grasp surfaces.

IV Future Plans

I will be working on generating offline database for the grasping surface by designing an automated algorithm . I will also be involved in generating the ground truth model once the items are released. Alex will be involved in refining the online grasping surface generation. He will also be working on generating the entire pipeline for autonomous single item picking task. Rick will be involved in improving the PERCH algorithm. Rick, along with Abhisek will be involved in designing the perception algorithm that makes use of both geometric information from point clouds and the RGB data. Feroze, along with me will be involved in generating the grasping pipeline. He will also be involved in creating 3D models using SIMTRACK