4/13/2016

Progress Review 12

Individual Lab Report #11

Abhishek Bhatia

Team D: Team HARP (Human Assistive Robotic Picker)
Teammates: Alex Brinkman, Feroze Naina, Lekha Mohan, Rick Shanor

I. Individual Progress

For this week's progress review, our major milestones were to get autonomous multi bin-test up and running with great accuracy, integrate Perch based vision pipeline and generate some results, and develop a backup Object Identification approach for the picking task. The first thing that I started working on in this respect was to develop a pipeline that utilizes the CNN based vision pipeline (already developed) for object identification. The idea behind this was to utilize another (secondary) Kinect-V2 for object identification, incase the confidence in the identification using the original pipeline was below par. In this approach, we go ahead and grasp the item, even if we are not exactly sure about the item's identification, then regenerate the identification result before dropping the item back into the bin. This is done using a setup that has the secondary Kinect-V2 mounted at the back of the order bin (tote), Figure 1, and right before dropping the item in the bin, the item is moved on top of the secondary Kinect-V2, which processes the item, does the identification and passes the information to main pipeline using computer networking.

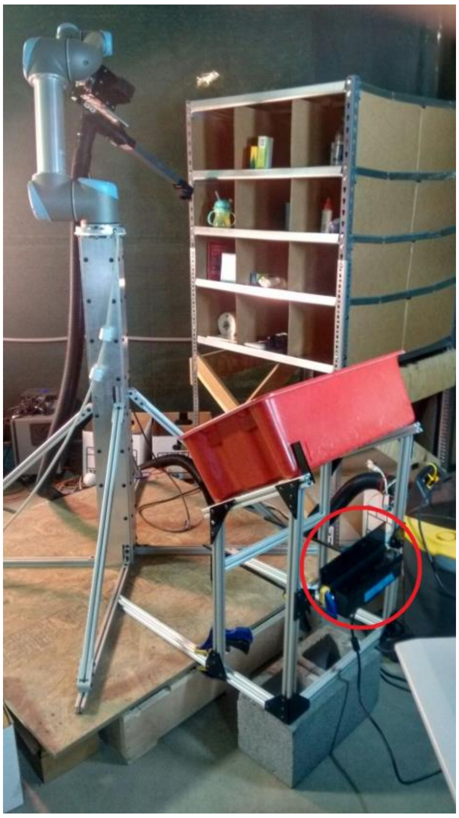

Figure 1: Image showing the setup for backup identification pipeline. The secondary Kinect-V2 (highlighted in red) facing upwards is used for this purpose

To do this, I worked on two things. First was developing the segmentation pipeline for the secondary Kinect-V2. Basically, once the object is moved above the secondary Kinect-V2, we need to segment the object out of the scene captured by the Kinect-V2. This segmented scene which contains just the object is now passed as a test image to the CNN network that makes a prediction for the item. The benefit of having this pipeline is that incase we end up picking an incorrect object, we may lose points if we drop an incorrect object into the order bin, with this pipeline, we can confirm that we have infact grasped the correct object. And incase we grasped the incorrect object, we can move this item back into the bin. So, the segmentation pipeline was developed as a server where the client passes the server the pointcloud and jpg image captured through the secondary Kinect-V2. The arm location behind the bin is picked strategically such that the 3-D scene captured by the Kinect-V2 has the item (grasped by the end-effector) with the minimum depth. Everything else in the scene has depth greater than the object. We then process the pointcloud and manually calculate the depth threshold such that we mark the value of every point that has the depth greater than this threshold in this pointcloud as NULL. Finally, we port this information from the pointcloud back to the jpg image, such that every pixel having the depth greater than the threshold has the rgb value corresponding to color black. This way, we are left with an image that just has the object from the scene in the image, which is used as a test image by the CNN to make the prediction, Figure 2.

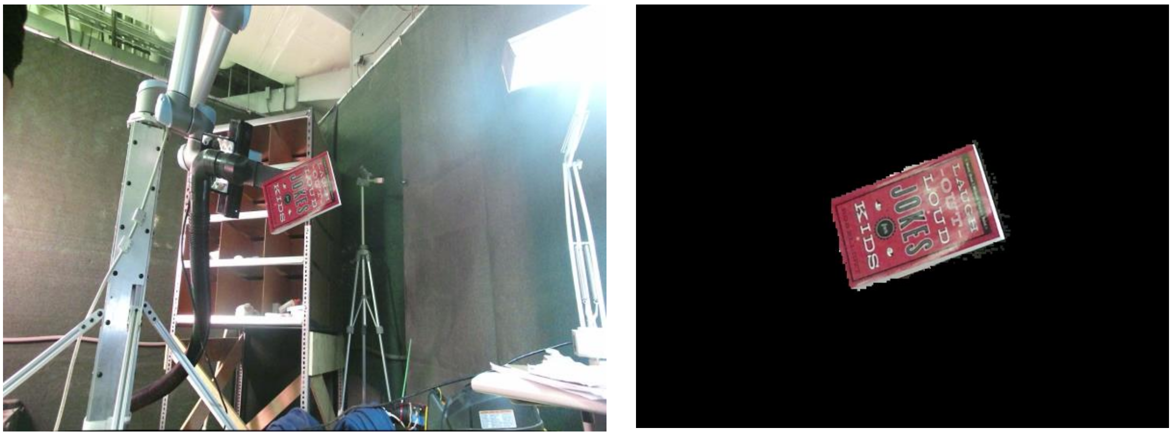

Figure 2: Image shows the end-effector moved behind the tote and infront of the secondary Kinect. This scene captured by the secondary Kinect on the left and the segmented output on the right. While testing, the secondary Kinect was kept on a table, which eventually has found its position below the order bin/tote.

The result of this experiment was great and we achieved almost 100% accuracy for the limited items that we tested upon. The second thing I worked on related to this was modifying the existing executive (state machine) to include this another state where we move the arm above the secondary Kinect-V2 before dropping the object into the order bin. Currently we have not interfaced the networking between the two computers, one where the main pipeline runs, and two where the processing for secondary Kinect-V2 happens. This was done on an experimental basis and we plan to include this as part of our main pipeline in future.

By the time I was done with this, Alex, Feroze and Rick worked together to include CNN based vision pipeline as part of our main pipeline and improved the grasp planner, such that we could carry out the multi-bin testing extensively. So, we all took out turns and spent couple of hours for 4 days before the dress rehearsal to test the system extensively. We tested the system against various corner cases, made notes of the failing cases and identified the faults in the subsystems and fixed them. Besides this, I also spent some time trying to understand the main executive properly, understanding all the states and state transitions to make sure everything was as expected. I found some discrepancies in this process which I discussed with Rick and fixed the executive. I am still in the process of verifying this modified executive.

II. Challenges

The major challenges that I faced this week were, first understanding and fixing a failure originating in the segmentation using secondary Kinect-V2, and second trying to understand the failure scenarios in the multi-item bins. The first one was challenging in the respect that when I had tested the segmentation individually, I had not taken care of a scenario when there is no object grasped by the end-effector. In such a scenario, no points in the pointclouds were detected with depth less than the threshold set. This was causing a seg-fault during conversion to jpg from the point cloud, Figure 3. So, when such a scenario came while testing the whole pipeline, it took me some time to decipher the cause of this failure. But, once I understood the issue, I fixed it in no time. I saw no more surprises in the segmentation pipeline. The second challenge was related to corner cases in our picking pipeline and we fixed them on a case to case basis.

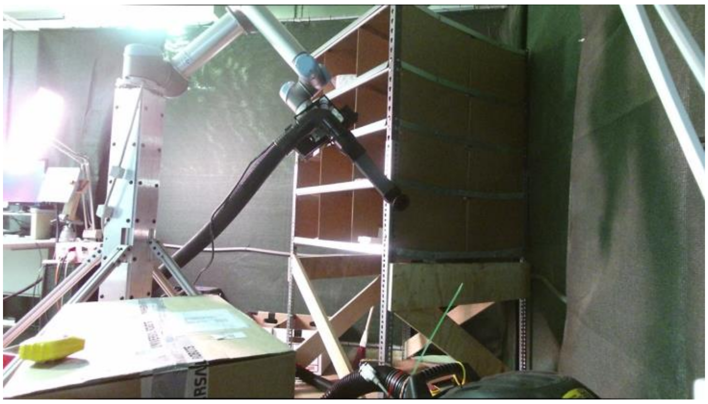

Figure 3: Image shows the end-effector moved behind the tote and infront of the secondary Kinect. This scene is captured by the secondary Kinect. This was a particular failing scenario where my server was giving a seg-fault.

III. Teamwork

For this week's progress review, we worked to improve the accuracy on our multi bin test.

Alex: Alex primarily worked on fixing the state machine to account for multi-bin scenarios. He worked to generate good pre-cached trajectories, which were eventually used during the demonstration.

Feroze: Feroze worked on fixing the online grasp planner to account for multiple items in a bin.

Lekha: Lekha worked on generating the 3D models for items with no definite shapes. The 3D models are required by Perch, which will be used to identify some items, which don't give great accuracy using the CNN.

Rick: Rick primarily worked on integrating the new CNN based vision pipeline as part of the main executive. He also worked on setting and integrating PERCH with the help of Venkat.

Abhishek: I primarily worked on segmentation of items from the scene using a secondary Kinect. I also spent a lot of time, testing various picking scenarios, analysing and fixing failures.

IV. Future Plans

My major targets for the Spring Validation Experiment include writing a high level planner that decides on the order in which we target shelf bins as to visit the bins with lesser number of items first. Also, I'd like to spend some time in parallelizing the vision processing. The idea is to capture the scenes from multiple bins in a single go and let the vision server do the processing in the background, this will eventually speed up our whole operation as carrying out vision processing serially is very time consuming.