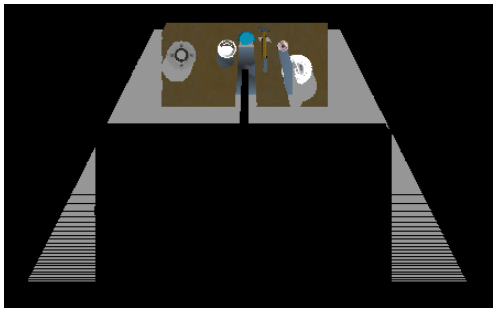
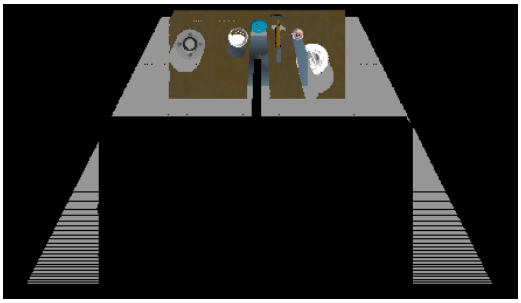
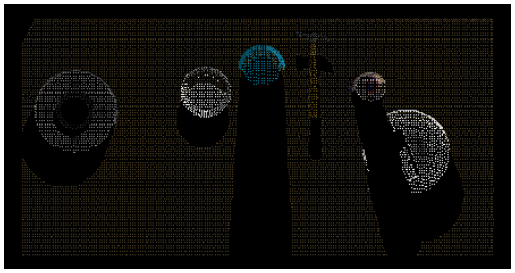
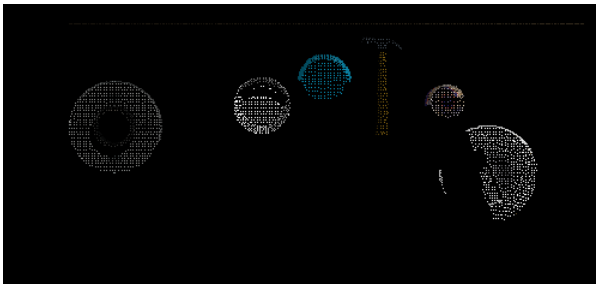
UDACITY PROJECT - PICK AND PLACE PERCEPTION

RUBRIC POINT EVALUATION

1. Complete Exercise 1 steps. Pipeline for filtering and RANSAC plane fitting implemented.

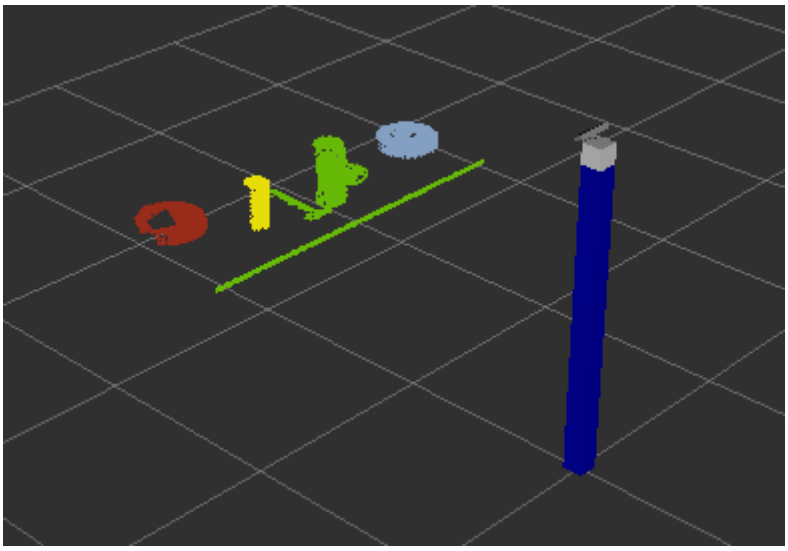
The following manipulations were done to process the image received by the RGBD camera and the following results were achieved.

Raw Image	Voxel Downsampled
	
Pass Through Filtered	**Inliers**
	

2. Complete Exercise 2 steps: Pipeline including clustering for segmentation implemented.

The following steps were implemented to visualise objects as separate clouds by Euclidean Clustering.

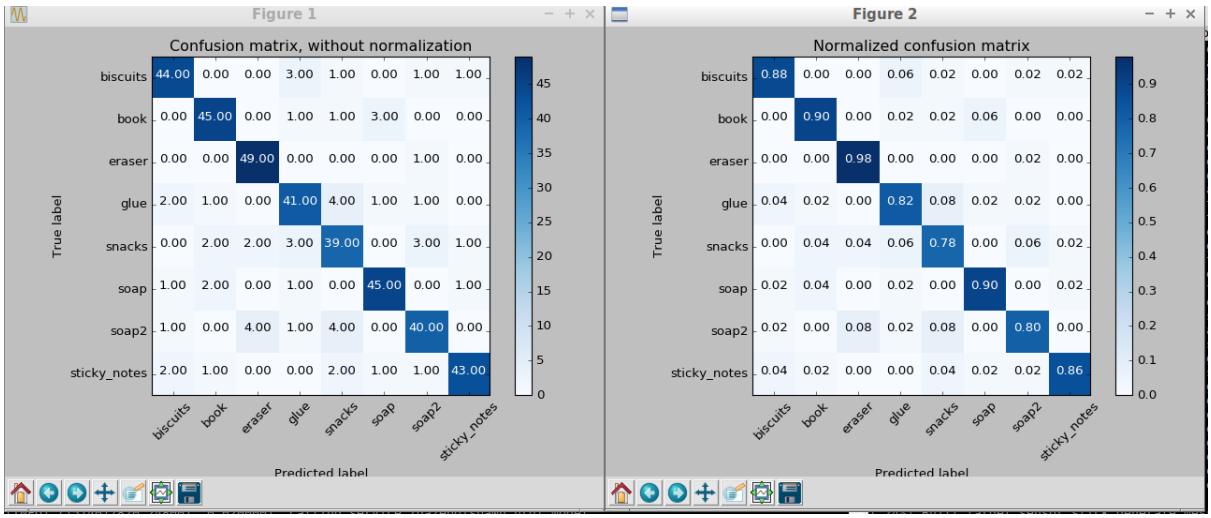
- Cluster the points using Euclidean clustering
- Traverse each cluster cloud and assign them different colors
- Publish the cluster cloud into a separate topic: cluster_cloud



3. Complete Exercise 3 Steps. Features extracted and SVM trained. Object recognition implemented.

For the robot to understand and recognize objects, it first needs to understand what the object looks like, very similar to a child learning to recognize objects by its five senses. Since we have only sight here to consider, the following steps are done to achieve the same.

- Generate features – understanding the outline, color, shape and different visual features of the objects.
- Train your SVM – I consider SVM to be the brain behind the entire operation – teaching it to understand and recognize the object in front of it. This is the output after I trained my SVM



- Label the objects – This is just the output to make sure that the SVM has understood the objects properly.

PICK AND PLACE SETUP

For all three tabletop setups (`test*.world`), perform object recognition, then read in respective pick list (`pick_list_*.yaml`). Next construct the messages that would comprise a valid `PickPlace` request output them to `.yaml` format.

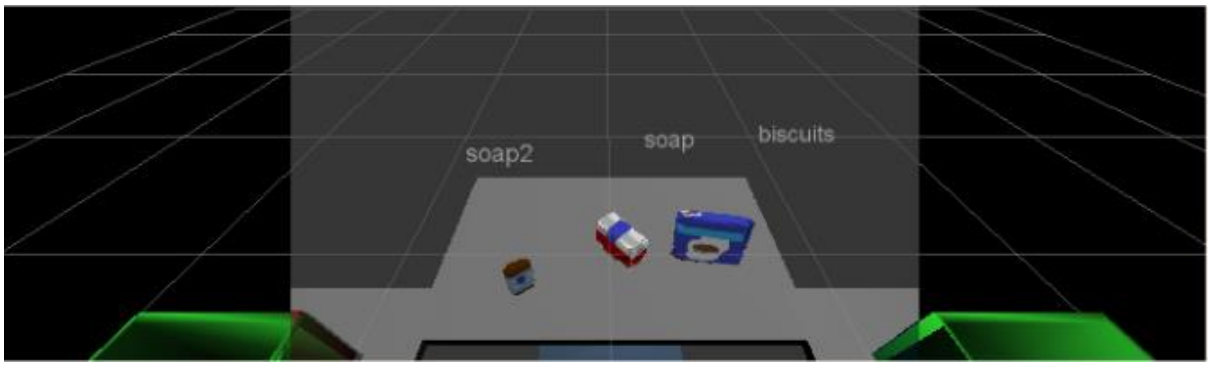
The above steps were implemented in the order specified. Additionally, the part of writing the details into the YAML file was also implemented. Corresponding project file is attached.

There are a couple of things that I have done a bit different:

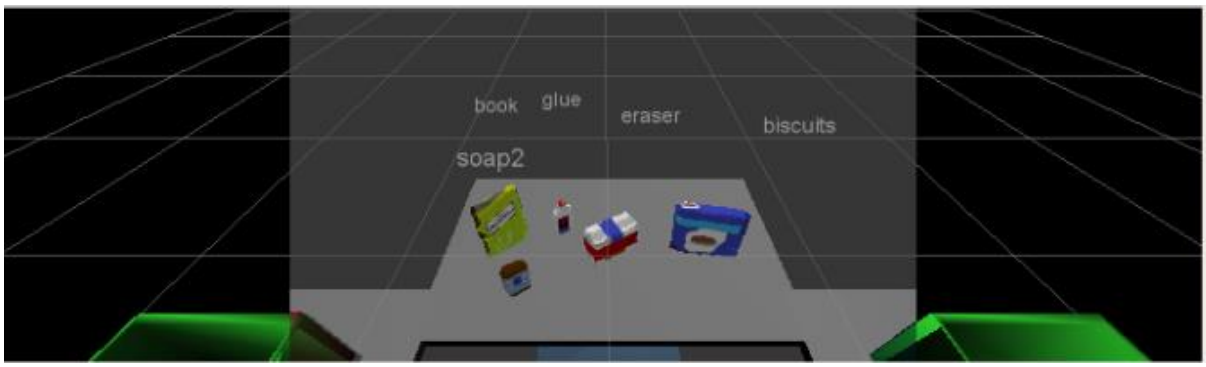
1. Implement additional statistical outlier filter after obtaining inliers. This is to sharpen the output for clustering.
2. Restricting the plane segmentation by y axis as well since the camera view covered a small part of the boxes on either side, causing it to be recognized as another cluster.
3. Filter, mean and tolerance parameters have been adjusted for the camera view of this project.

Output:

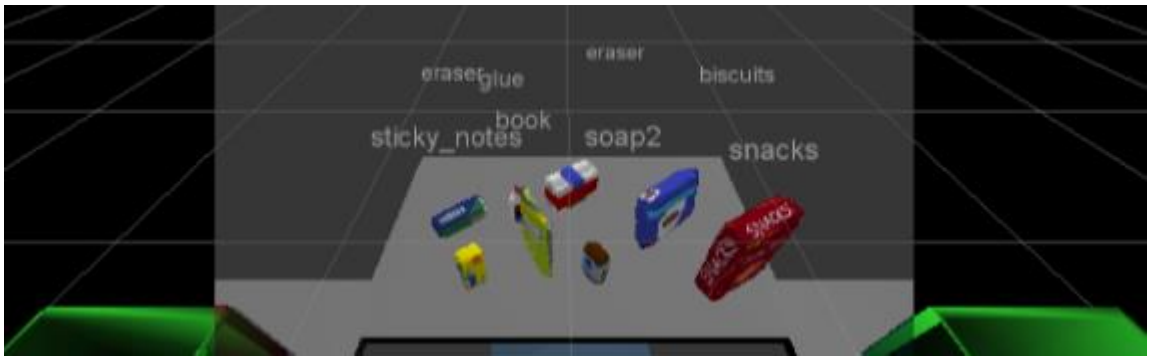
Pick List 1



Pick List 2



Pick List 3


eraser
eraser
glue
biscuits
book
sticky_notes
soap2
snacks