

Project: Perception Pick & Place

Required Steps for a Passing Submission:

1. Extract features and train an SVM model on new objects (see `pick_list_*.yaml` in `/pr2_robot/config/` for the list of models you'll be trying to identify).
 2. Write a ROS node and subscribe to `/pr2/world/points` topic. This topic contains noisy point cloud data that you must work with.
 3. Use filtering and RANSAC plane fitting to isolate the objects of interest from the rest of the scene.
 4. Apply Euclidean clustering to create separate clusters for individual items.
 5. Perform object recognition on these objects and assign them labels (markers in RViz).
 6. Calculate the centroid (average in x, y and z) of the set of points belonging to that each object.
 7. Create ROS messages containing the details of each object (name, pick_pose, etc.) and write these messages out to `.yaml` files, one for each of the 3 scenarios (`test1-3.world` in `/pr2_robot/worlds/`). [See the example `output.yaml` for details on what the output should look like.](#)
 8. Submit a link to your GitHub repo for the project or the Python code for your perception pipeline and your output `.yaml` files (3 `.yaml` files, one for each test world). You must have correctly identified 100% of objects from `pick_list_1.yaml` for `test1.world`, 80% of items from `pick_list_2.yaml` for `test2.world` and 75% of items from `pick_list_3.yaml` in `test3.world`.
 9. Congratulations! Your Done!
-

Writeup / README

You're reading it!

Hi. I am Miguel Colmenares, engineer in electronics and i want to tell you about my experience in this project.

And part of this project i shared it in my instagram: @uchobogens

Kinematic Analysis

Exercise 1, 2 and 3 pipeline implemented

1. Completed Exercise 1 steps:

[Imagen: tabletop]

robond@udacity: ~/RoboND-Perception-Exercises/Exercise-1

```

robond@udacity: ~/RoboND-Perception-Exercises/Exercise-1 80x24
Desktop  Downloads  Music  Public  Templates
Documents  miniconda3  Pictures  RoboND-Perception-Exercises  Videos
robond@udacity:~$ cd Robo
bash: cd: No such file or directory
robond@udacity:~/RoboND-Perception-Exercises/
robond@udacity:~/RoboND-Perception-Exercises$ ls
CODEOWNERS  Exercise-2  Lab  project_template.py  python-pcl
Exercise-1  Exercise-3  LICENSE  Projeto  README.md
robond@udacity:~/RoboND-Perception-Exercises$ cd Exercise-1
robond@udacity:~/RoboND-Perception-Exercises/Exercise-1$ ls
extracted_inliers.pcd  RANSAC.py  tabletop.pcd
extracted_outliers.pcd  README.md  voxel_downsampled.pcd
pass_through_filtered.pcd  tabletopnew.pcd
robond@udacity:~/RoboND-Perception-Exercises/Exercise-1$ pcl_viewer tabletop.pcd

The viewer window provides interactive commands; for help, press 'h' or 'H' from
within the window.
> Loading tabletop.pcd ERROR: In /build/vtk6-dmAaMa/vtk6-6.2.0+dfsg1/Rendering/0
openGL/vtkOpenGLTexture.cxx, line 200
vtkOpenGLTexture (0x1015ce0): No scalar values found for texture input!

[done, 1733 ms : 202627 points]
Available dimensions: x y z rgb
57 # TODO Call the segment function to obtain set of inlier indices
58 # and model coefficients
59 inliers, coefficients = seg.segment()
60
61 # Extract inliers
62 extracted_inliers = cloud_filtered.extract(inliers, negative=False)
63 filename = 'extracted_inliers.pcd'
64 pcl.save(extracted_inliers, filename)
65
66 # Extract outliers
67 extracted_outliers = cloud_filtered.extract(inliers, negative=True)
68 filename = 'extracted_outliers.pcd'
69 pcl.save(extracted_outliers, filename)
70
71 # Save pcd for table
72 # pcl.save(cloud, filename)
73
74 # Save pcd for tabletop objects
75 filename = 'tabletopnew.pcd'
76 pcl.save(cloud, filename)
77

```

PCD viewer

tabletop.pcd
301.9 FPS

Python ▾ Tab Width: 8 ▾ Ln 69, Col 39 ▾ INS In this

image I have the environment, the table and the objects in high resolution.

[Imagen: voxeldown]

robond@udacity: ~/RoboND-Perception-Exercises/Exercise-1

```

robond@udacity: ~/RoboND-Perception-Exercises/Exercise-1 80x24
The viewer window provides interactive commands; for help, press 'h' or 'H' from
within the window.
> Loading tabletop.pcd ERROR: In /build/vtk6-dmAaMa/vtk6-6.2.0+dfsg1/Rendering/OpenGL/vtkOpenGLTexture.cxx, line 200
vtkOpenGLTexture (0x1015ce0): No scalar values found for texture input!

[done, 1733 ms : 202627 points]
Available dimensions: x y z rgb
^C
robond@udacity:~/RoboND-Perception-Exercises/Exercise-1$ ls
extracted_inliers.pcd      RANSAC.py          tabletop.pcd
extracted_outliers.pcd      README.md        voxel_downsampled.pcd
pass_through_filtered.pcd   tabletopnew.pcd
robond@udacity:~/RoboND-Perception-Exercises/Exercise-1$ pcl_viewer voxel_downsampled.pcd
The viewer window provides interactive commands; for help, press 'h' or 'H' from
within the window.
> Loading voxel_downsampled.pcd ERROR: In /build/vtk6-dmAaMa/vtk6-6.2.0+dfsg1/Rendering/OpenGL/vtkOpenGLTexture.cxx, line 200
vtkOpenGLTexture (0x29c9cf0): No scalar values found for texture input!

[done, 461 ms : 55336 points]
Available dimensions: x y z rgb

```

57 # TODO call the segment function to obtain set of intlier indices
58 # and model coefficients
59 inliers, coefficients = seg.segment()
60
61 # Extract inliers
62 extracted_inliers = cloud_filtered.extract(inliers, negative=False)
63 filename = 'extracted_inliers.pcd'
64 pcl.save(extracted_inliers, filename)
65
66 # Extract outliers
67 extracted_outliers = cloud_filtered.extract(inliers, negative=True)
68 filename = 'extracted_outliers.pcd'
69 pcl.save(extracted_outliers, filename)
70
71 # Save pcd for table
72 # pcl.save(cloud, filename)
73
74 # Save pcd for tabletop objects
75 filename = 'tabletopnew.pcd'
76 pcl.save(cloud, filename)
77

voxel_downsampled.pcd
366.8 FPS

PCD viewer

filter was applied to lower the resolution of the captured image.

[Imagen: passthrough]

A voxel

robond@udacity: ~/RoboND-Perception-Exercises/Exercise-1 RANSAC.py (~/RoboN... robond PCD viewer 52 23:37

```
robond@udacity: ~/RoboND-Perception-Exercises/Exercise-1 - + x PCD viewer
robond@udacity: ~/RoboND-Perception-Exercises/Exercise-1 80x24
vtkOpenGLTexture (0x29c9cf0): No scalar values found for texture input!
[done, 461 ms : 55336 points]
Available dimensions: x y z rgb
^C
robond@udacity:~/RoboND-Perception-Exercises/Exercise-1$ ls
extracted_inliers.pcd      RANSAC.py          tabletop.pcd
extracted_outliers.pcd      README.md        voxel_downsampled.pcd
pass_through_filtered.pcd   tabletopnew.pcd
robond@udacity:~/RoboND-Perception-Exercises/Exercise-1$ pcl_viewer pass
The viewer window provides interactive commands; for help, press 'h' or 'H' from
within the window.
No .PCD or .VTK file given. Nothing to visualize.
robond@udacity:~/RoboND-Perception-Exercises/Exercise-1$ pcl_viewer pass_through
_filtered.pcd
The viewer window provides interactive commands; for help, press 'h' or 'H' from
within the window.
> Loading pass_through_filtered.pcd ERROR: In /build/vtk6-dmAaMa/vtk6-6.2.0+dfsg
1/Rendering/OpenGL/vtkOpenGLTexture.cxx, line 200
vtkOpenGLTexture (0x21fbce0): No scalar values found for texture input!

[done, 170 ms : 16302 points]
Available dimensions: x y z rgb
57 # TODO Call the segment function to obtain set of inlier indices
58 # and model coefficients
59 inliers, coefficients = seg.segment()
60
61 # Extract inliers
62 extracted_inliers = cloud_filtered.extract(inliers, negative=False)
63 filename = 'extracted_inliers.pcd'
64 pcl.save(extracted_inliers, filename)
65
66 # Extract outliers
67 extracted_outliers = cloud_filtered.extract(inliers, negative=True)
68 filename = 'extracted_outliers.pcd'
69 pcl.save(extracted_outliers, filename)
70
71 # Save pcd for table
72 # pcl.save(cloud, filename)
73
74 # Save pcd for tabletop objects
75 filename = 'tabletopnew.pcd'
76 pcl.save(cloud, filename)
77
```

pass_through_filtered.pcd
437.1 FPS

Python ▾ Tab Width: 8 ▾ Ln 40, Col 35 ▾ INS

With the

pass_throgh filter i only get the table with the objects.

[Imagen: tableinliers]

robond@udacity: ~/RoboND-Perception-Exercises/Exercise-1

```
The viewer window provides interactive commands; for help, press 'h' or 'H' from
within the window.
> Loading pass_through_filtered.pcd ERROR: In /build/vtk6-dmAaMa/vtk6-6.2.0+dfsg1/Rendering/OpenGL/vtkOpenGLTexture.cxx, line 200
vtkOpenGLTexture (0x21fbce0): No scalar values found for texture input!

[done, 170 ms : 16302 points]
Available dimensions: x y z rgb
^C
robond@udacity:~/RoboND-Perception-Exercises/Exercise-1$ ls
extracted_inliers.pcd      RANSAC.py          tabletop.pcd
extracted_outliers.pcd      README.md        voxel_downsampled.pcd
pass_through_filtered.pcd   tabletopnew.pcd
robond@udacity:~/RoboND-Perception-Exercises/Exercise-1$ pcl_viewer extracted_inliers.pcd
The viewer window provides interactive commands; for help, press 'h' or 'H' from
within the window.
> Loading extracted_inliers.pcd ERROR: In /build/vtk6-dmAaMa/vtk6-6.2.0+dfsg1/Re
ndering/OpenGL/vtkOpenGLTexture.cxx, line 200
vtkOpenGLTexture (0x130a310): No scalar values found for texture input!

[done, 137 ms : 11881 points]
Available dimensions: x y z rgb
```

PCD viewer

extracted_inliers.pcd
426.8 FPS

13 items (44 hidden)

Free space: 3.4 GiB (Total: 15.6 GiB) With the

extracted_inliers filter i only get the table.

[Imagen: objecetliers]

robond@udacity: ~/RoboND-Perception-Exercises/Exercise-1

```

robond@udacity: ~/RoboND-Perception-Exercises/Exercise-1 80x24
extracted_outliers.pcd  README.md      voxel_downsampled.pcd
pass_through_filtered.pcd  tabletopnew.pcd
robond@udacity:~/RoboND-Perception-Exercises/Exercise-1$ pcl_viewer extracted_inliers.pcd
The viewer window provides interactive commands; for help, press 'h' or 'H' from
within the window.
> Loading extracted_inliers.pcd ERROR: In /build/vtk6-dmAaMa/vtk6-6.2.0+dfsg1/Re
ndering/OpenGL/vtkOpenGLTexture.cxx, line 200
vtkOpenGLTexture (0x130a310): No scalar values found for texture input!

[done, 137 ms : 11881 points]
Available dimensions: x y z rgb
^C
robond@udacity:~/RoboND-Perception-Exercises/Exercise-1$ pcl_viewer extracted_outliers.pcd
The viewer window provides interactive commands; for help, press 'h' or 'H' from
within the window.
> Loading extracted_outliers.pcd ERROR: In /build/vtk6-dmAaMa/vtk6-6.2.0+dfsg1/R
endering/OpenGL/vtkOpenGLTexture.cxx, line 200
vtkOpenGLTexture (0x14541a0): No scalar values found for texture input!

[done, 88 ms : 4421 points]
Available dimensions: x y z rgb

```

PCD viewer

extracted_outliers.pcd
1575.8 FPS

14 items (44 hidden)

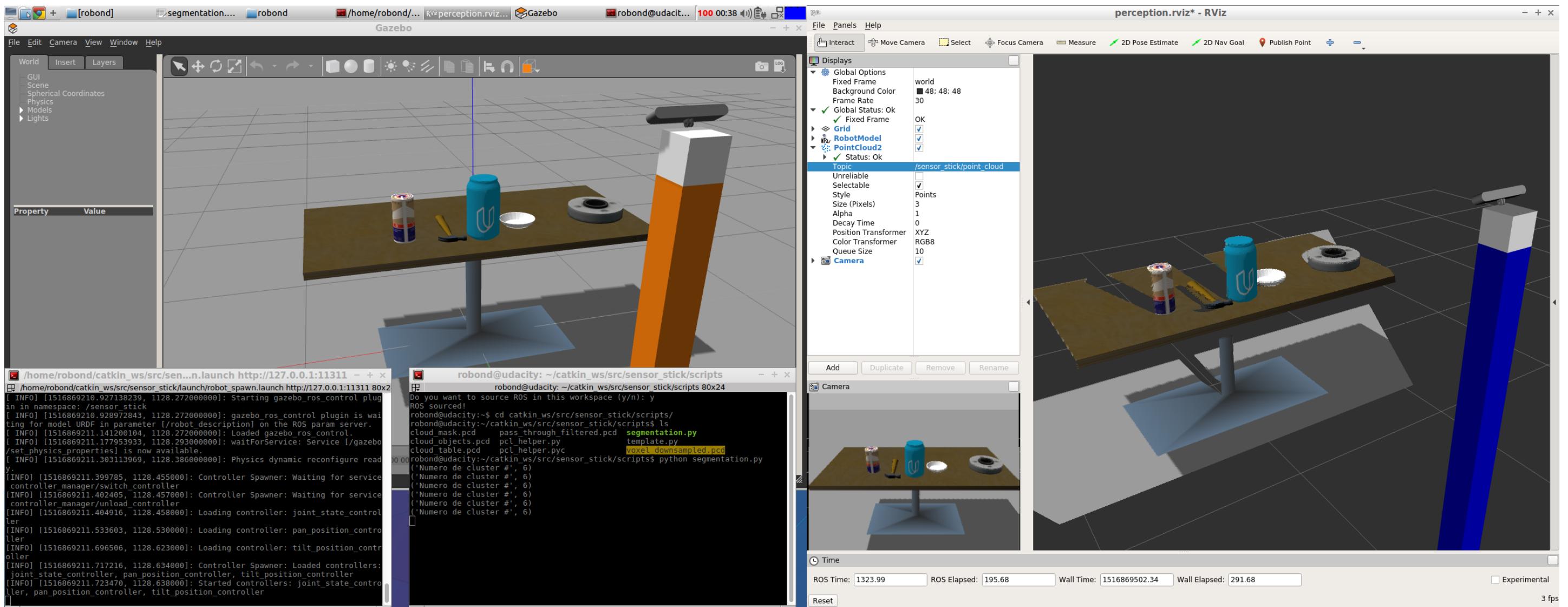
Free space: 3.4 GiB (Total: 15.6 GiB) And with

the extracted_outliers filter i only get the objects.

2. Completed Exercise 2 steps:

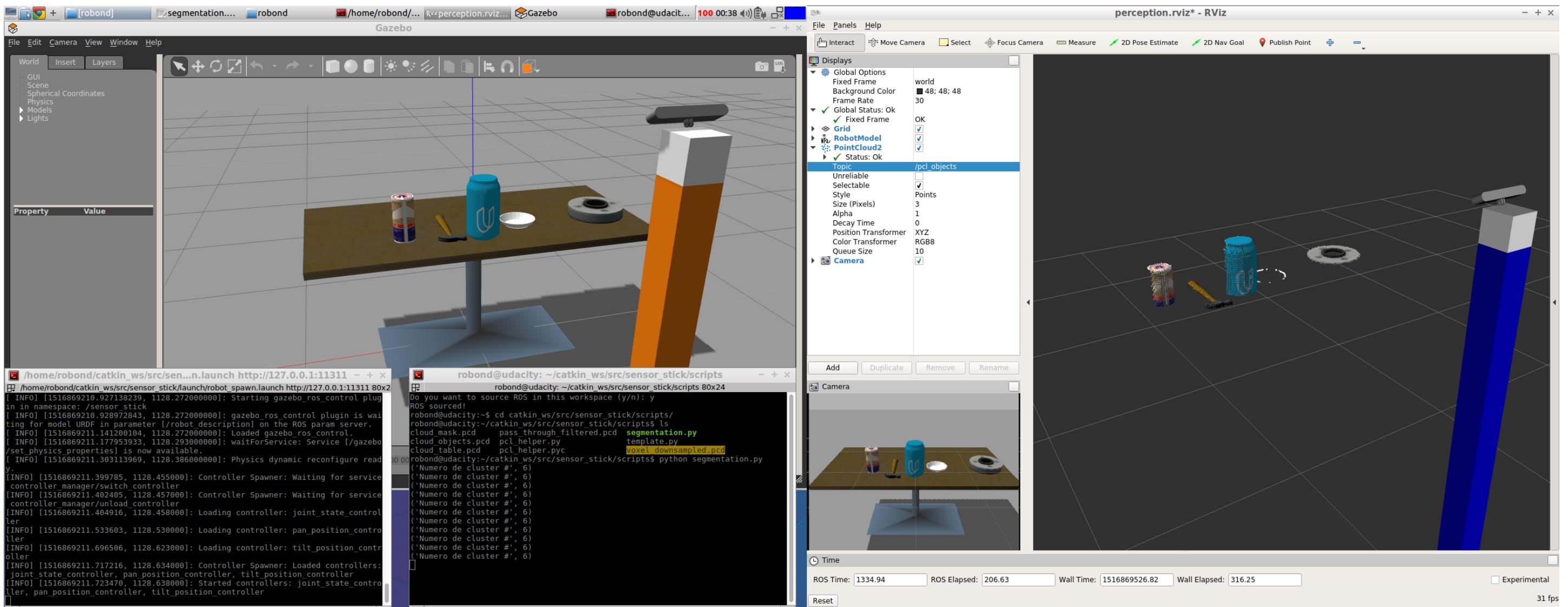
Now that I get only the objects, I can identify each object in a cluster and assign it a color.

[Imagen: pointcloud]



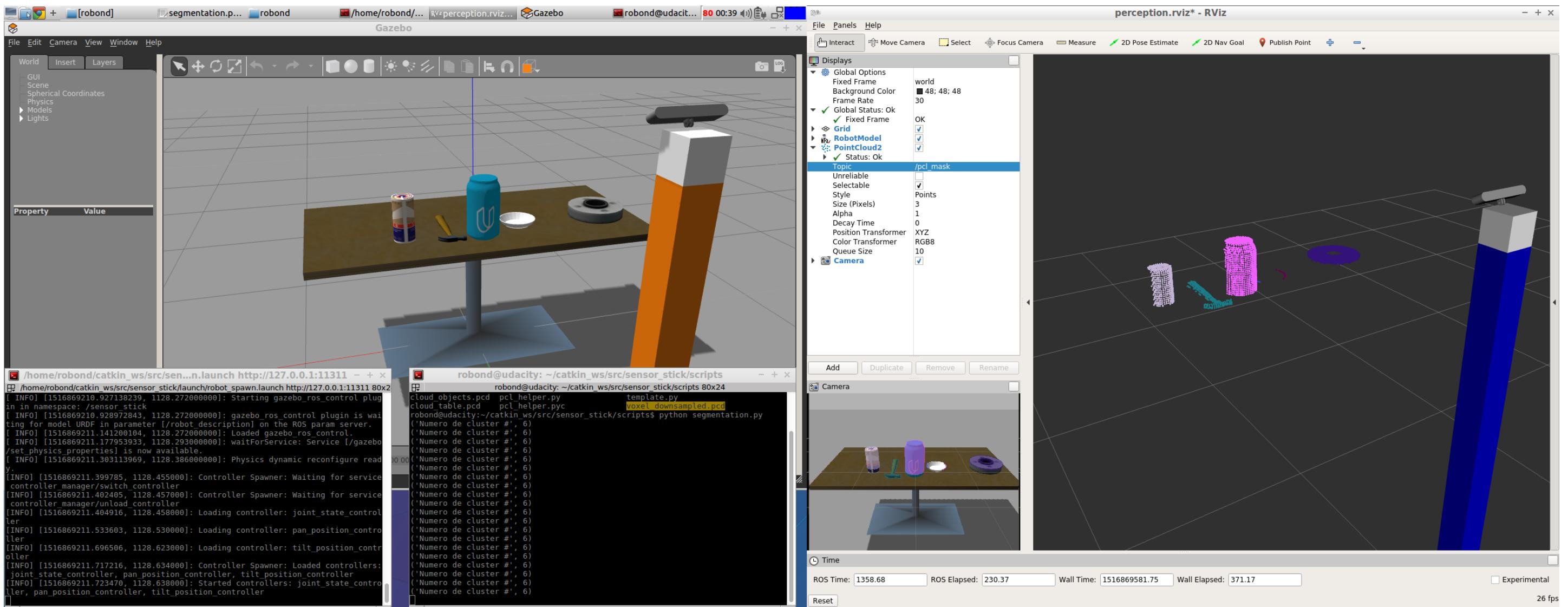
Here you can appreciate the environments in Gazebo and Rviz.

[Imagen: pclobjcts]



With the Topic: "/pcl_objects" you can appreciate only the objects.

[Imagen: pclmask]



With the Topic: "/pcl_mask" you can appreciate only the cloud_cluster mask.

[Imagen: objcYmask]

The screenshot shows a terminal window and two PCD viewer windows. The terminal window displays ROS commands and their outputs:

```

robond@udacity: ~/catkin_ws/src/sensor_stick/scripts
robond@udacity: ~/catkin_ws/src/sensor_stick/scripts 80x22
robond@udacity:~/catkin_ws/src/sensor_stick/scripts$ pcl_viewer cloud ob
The viewer window provides interactive commands; for help, press 'h' or 'H' from
within the window.
No .PCD or .VTK file given. Nothing to visualize.
robond@udacity:~/catkin_ws/src/sensor_stick/scripts$ pcl_viewer cloud_objects.pcd
The viewer window provides interactive commands; for help, press 'h' or 'H' from
within the window.
> Loading cloud_objects.pcd ERROR: In /build/vtk6-dmAaMa/vtk6-6.2.0+dfsg1/Rendering/OpenGL/vtkOpenGLTexture.cxx, line 200
vtkOpenGLTexture (0x2b74210): No scalar values found for texture input!

[done, 73 ms : 2372 points]
Available dimensions: x y z rgb

```

The two PCD viewer windows show 3D point clouds. The top window is titled "cloud_objects.pcd" and shows three objects: a blue cylinder, a red cylinder, and a green ring. The bottom window is titled "cloud_mask.pcd" and shows the same objects with different colors: a cyan cylinder, a magenta cylinder, and a blue ring. Both windows have a FPS counter at the bottom right.

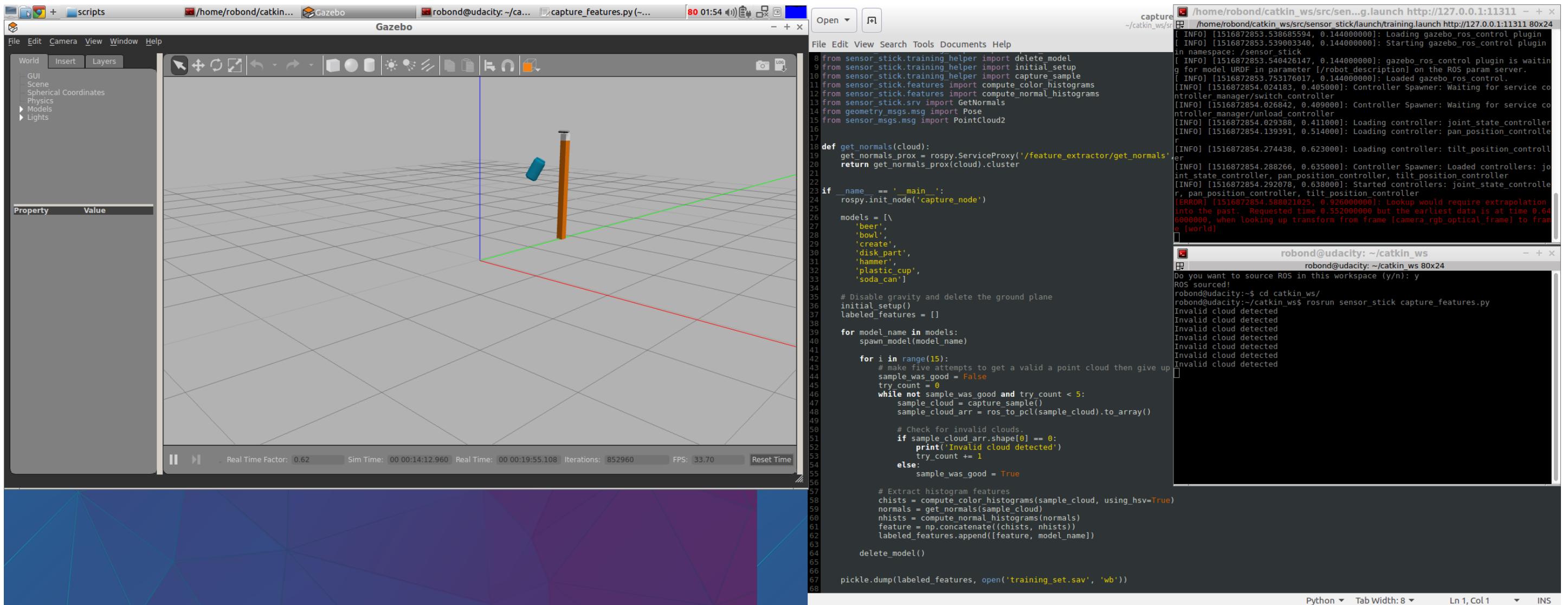
can better appreciate the objects and the masks.

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

Features extracted:

[Imagen: capturefeature]

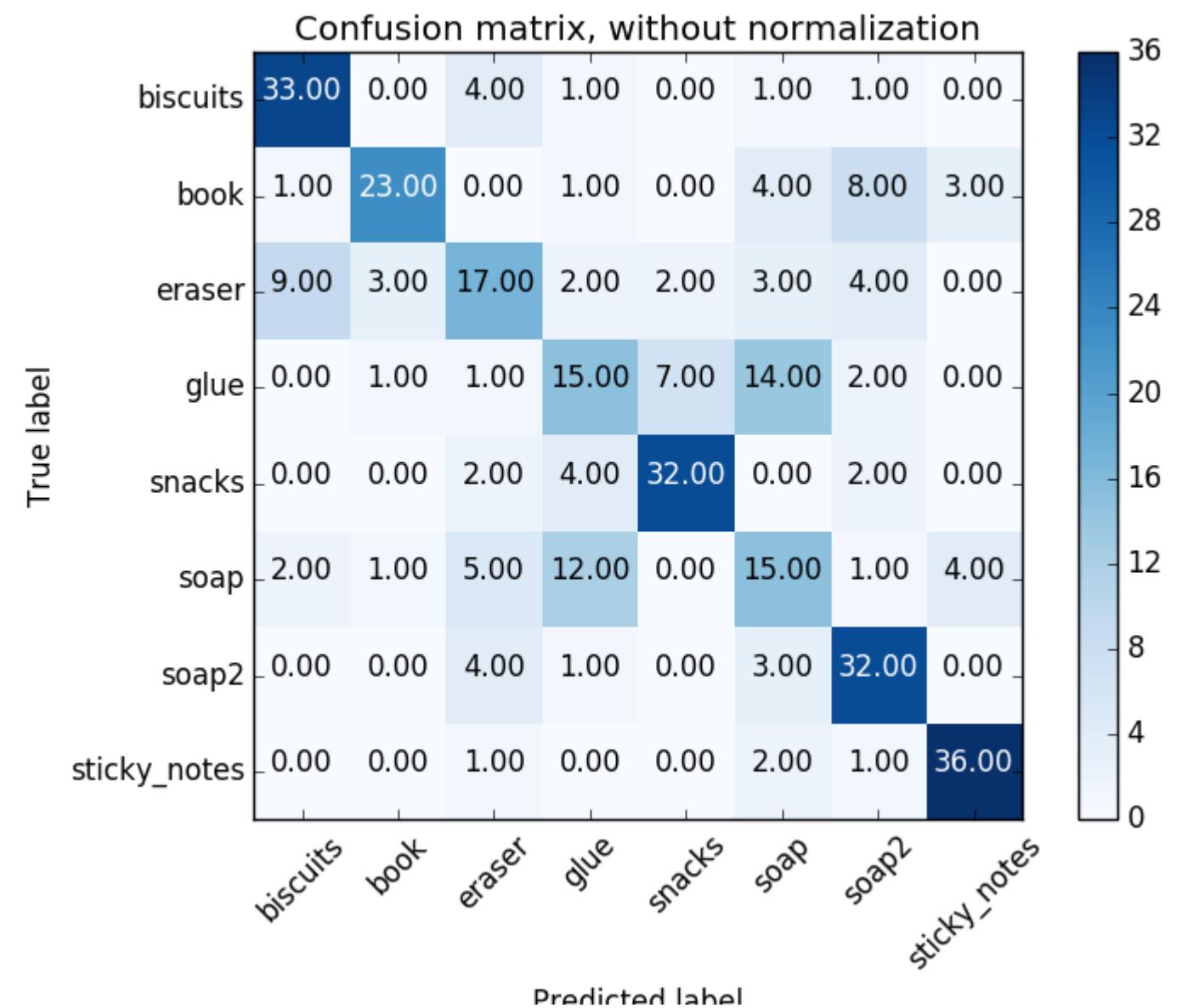
Here you



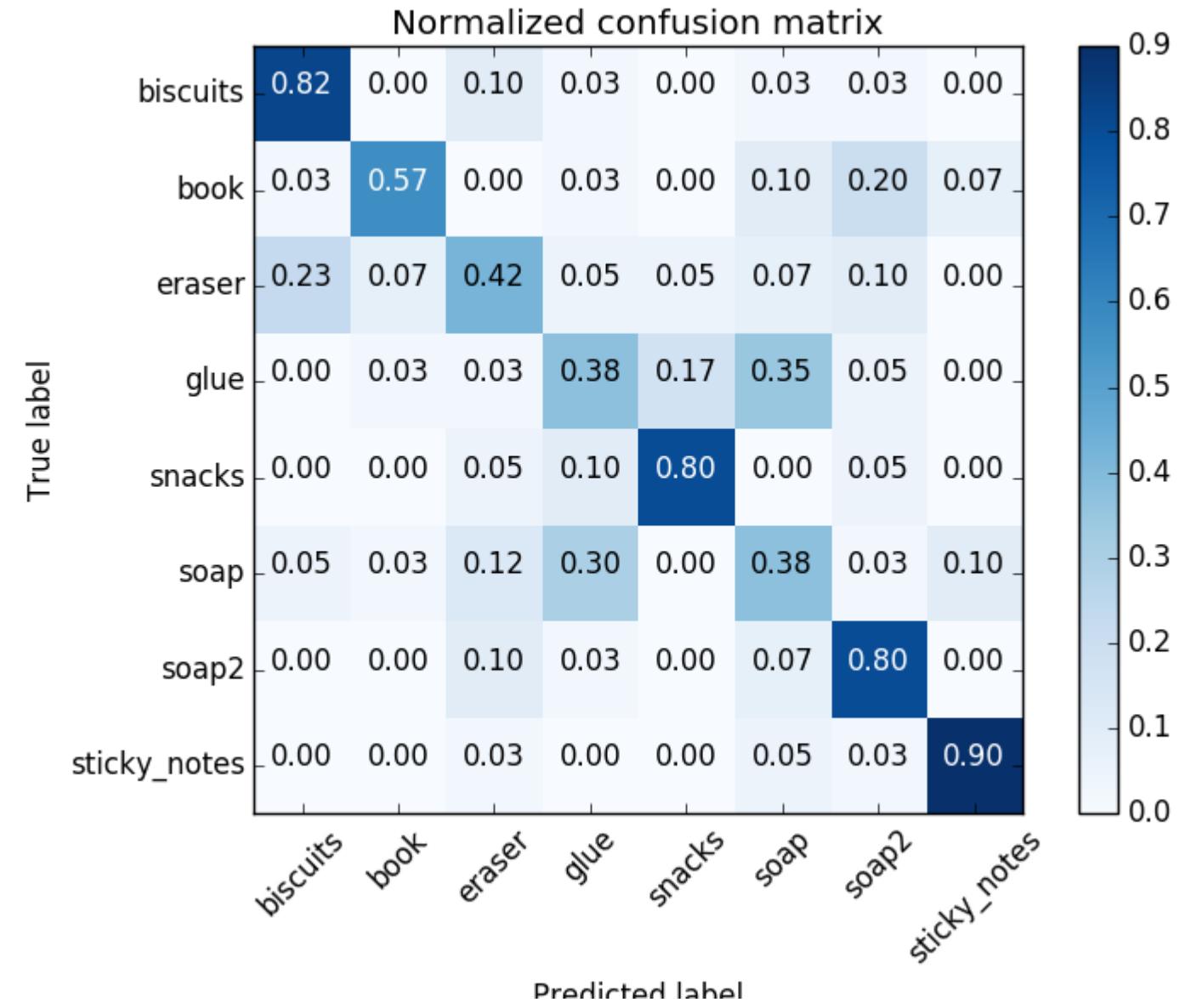
In a different environment 15 photos are taken to all the objects and their characteristics are captured.

SVM trained:

[Imagen: Confusion matrix]



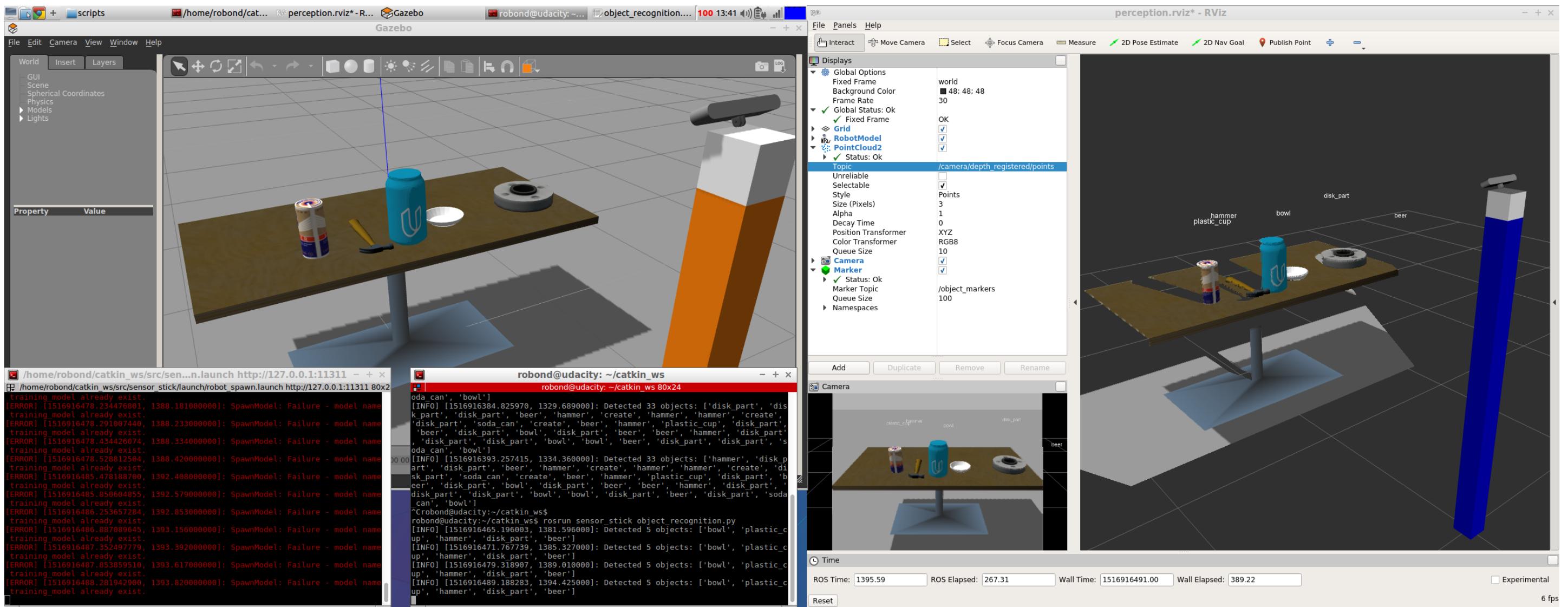
[Imagen: Normalized matrix]



Confusion matrix to SVM trained of project.

Object recognition:

[Imagen: objectsname]

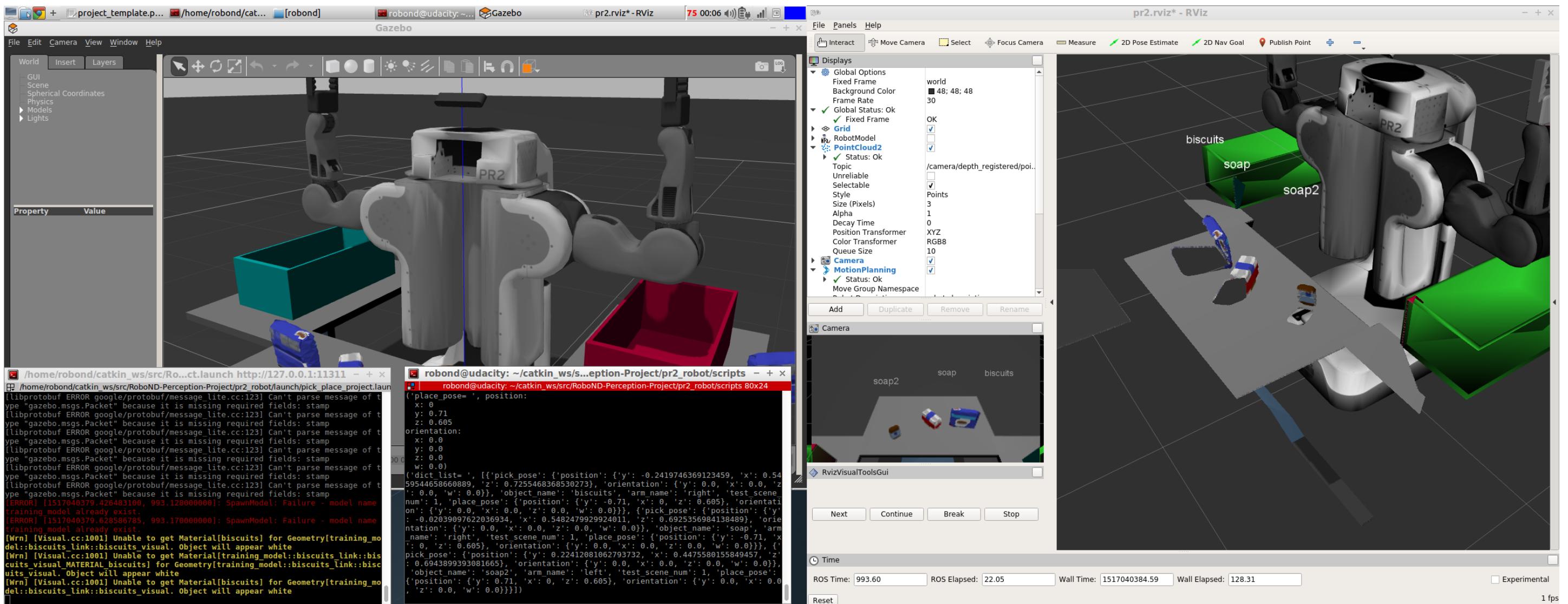


Recognized and labeled objects.

Pick and Place Setup

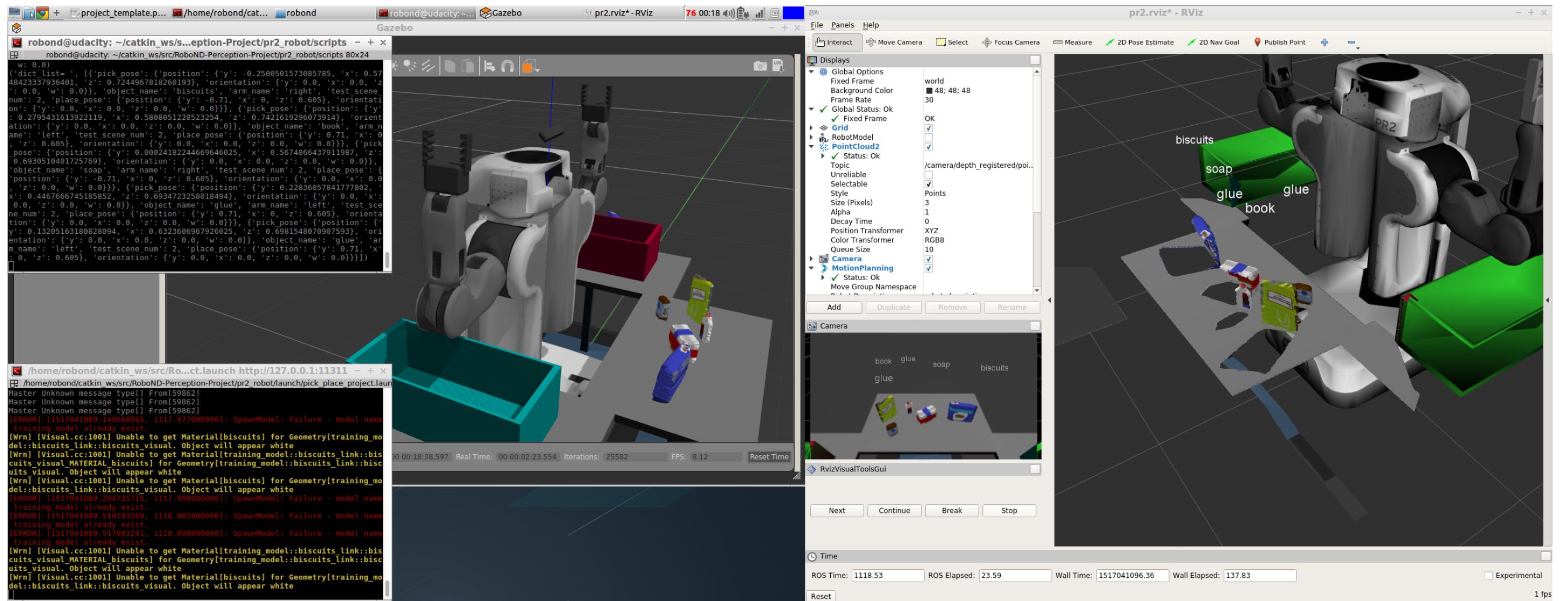
For all three tabletop setups (`test*.world`), perform object recognition, then read in respective pick list (`pick_list_*.yaml`). Next construct the messages that would comprise a valid `PickPlace` request output them to `.yaml` format.

[Imagen: world1]



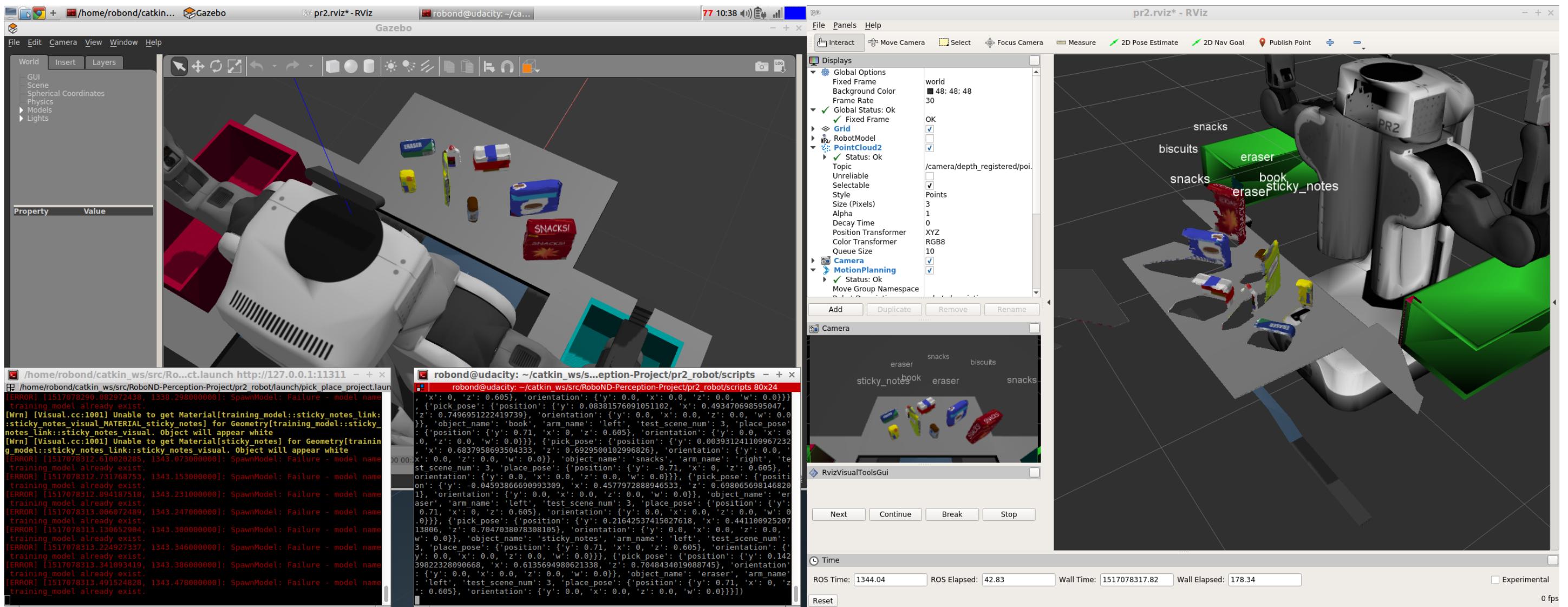
pick_place_project.launch: test1.world - pick_list_1.yaml project_template.py: models = 3 Objects - output file: output1.yaml

[Imagen: world2]



pick_place_project.launch: test2.world - pick_list_2.yaml - project_template.py: models = 5 Objects - output file: output2.yaml

[Imagen: world3]



pick_place_project.launch: test3.world - pick_list_3.yaml - project_template.py: models = 8 Objects - output file: output3.yaml

I loved this project. generate a cloud of points with color and depth and from there create a cluster, filter noise, capture features, train and most importantly recognize objects and all this by adding ROS to implement it in a pr2_robot. Fantastic.

Points to improve: improve the way to apply voxel to the environment. Improve the prediction technique to improve the confusion matrix. Improve the ways of the arm and avoid collisions.

Thank you very much enjoy it.

PD: the outputX.yaml files are in the following route RoboND-Perception-Project-Miguel\RoboND-Perception-Project-master\pr2_robot\scripts