Senior Capstone Notes

11/18/2015

Combine visual queues with robot arm handling to make object handling that is course correcting.

Change visual data into a point-cloud, so that you have references for points in space.

You can cluster point-clouds to represent objects.

The rear facing camera has a range from .6m to 3.5m.

Our project is to be able to identify that flat (such as a smartphone) objects are within a flat area.

Algorithm is up to us, but it is suggested to have an initial movement that gets us close and then use error correction.

Book: ROS by Example volume 2

They hope to learn things that need to be changed to the Move-It library that they can improve it for the OSS ROS community.

The camera software will be linked to us after the meeting.

Build a good implementation of object finding, which uses significantly less power in an optimal way.

Use PCL (point-cloud library), but optimize it.

Geometric vector mapping might be useful

Look into how Google Sketch Up handled modeling 3d from a pic.

It may be handy to use GPU processing.

Must be able to run the system effectively on an Intel Nuc

We have a guide for OpenCL reference.

Stages:

1. Segment the planes (floor, walls, etc.)
2. Adjust plane for the angle that you are observing or adjusting to
3. Figure out a way to compensate for errors in observation
4. Find a way to remove the plane, so that the objects that are in the plane are identified
5. Resolution constraints limit the ability to find objects at farther distances.
6. The camera should be able to identify an object 2.5m away.
7. ROS might provide a lot of the tools that
8. It may be very useful to use a virtual machine.