

CMRoboBits (15-491, Fall 2007) "Vision Warm-Up" lab (HW3)

Due Wednesday, October 3rd, at lab-time (12:30pm).

Description

In this lab you will navigate an iRobot Create through a maze by showing colored signs to a webcam and by using the robot's bumper sensor.

Your robot should have the following behavior:

- When starting your program, the robot should be stopped.
- Whenever your robot is stopped, it should look at the Webcam's Color Segmentation output where it expects to see one of three signals (in form of different colored papers that you will be holding):
 - If you show your robot the "go" flag (e.g. green paper) it should start driving forward and ignore all computer vision until it hits something with its front bumper. Once it hits an object with the bumper it should come to a full stop and start processing vision flags again.
 - If you show your robot the "left" flag (e.g. yellow paper), it should rotate left, in place.
 - If you show your robot the "right" flag (e.g. blue paper), it should rotate right, in place.
- Your rotations should be small enough so you can accurately align your robot with the next upcoming hallway of the maze (which will be randomly constructed during demo-day). The idea is that you keep showing your robot colored signs (e.g. "left left left left right left") until it is perfectly aligned with the next hallway of the maze and then give it the "go" signal to resume course. Obviously, you will need to have some sort of loop (or event-handler) to keep analyzing the output of the Color Segmentation service.
- When hitting an object with the bumper, it might be a good idea to have the robot move backwards a few centimeters before coming to a full stop. This will ensure that your robot can turn freely without scraping against the wall it just hit.
- Which three colors you end up using for the signs is up to you. You may find that some colors work better than others. Also, if you would like to add additional flags for more precise control or combinatorial flags (e.g.

yellow and blue flag in the same image means reverse), you can feel free to do so, but it is not required.

Demo / VPL / Write-Up

As usual, you should submit your VPL code and a write-up to your dropbox/hw3 directory. In your write-up, mention any particular design decisions you took and how they affected your result.

All groups will need to demo their program at lab-time. We will setup a random maze through which you have to navigate your robot.

To make it more fun, we will also time how long it takes your robot to complete the maze. The fastest group will earn the respect of their peers and the instructors, and possibly some extra credit points. Note that since this is a robot/human interaction assignment, you might want to train your own flag-showing abilities before the demo...

Technical Details + HowTo

BlueTooth

The BlueTooth connection of the iRobot Create is very similar to the scribbler, EXCEPT that the passcode for the Create is 0000 (not 1234).

Unlike the Scribbler, the iRobot Create does not automatically figure out the ComPort. You might need to look at the BlueTooth->ComPort tab (or device manager->ports) to figure out which ComPort to use before hitting the iRobot "Connect" button in the Web-Interface.

iRobot Create

To drive the iRobot Create, please use the Generic Differential Drive with the iRobot Create manifest.

To access its sensor you should be able to use the iRobot Generic Contact Sensors for which you might also need to load a manifest.

WebCams

There is exactly one webcam for each group. Please coordinate with your partner so that every group has access to a webcam at any point in time.

Please install the driver for the webcam before plugging it in. The driver CDs are located in the REL.

ColorSegmentation Service

Robotics Studio has several vision-related services. We strongly recommend that you use the "Colorsegmentation" one.

Make sure to import the manifest of the Color Segmentation service before using it. Once that is done you can start your program and access the calibration routines of your Color Segmentation service through the Web-Interface at <http://localhost:50000/colorsegment>

In the web-interface you will be able to train your colors by marking regions inside of the image. You then get to assign color-labels to your training-samples. To learn a bit more about the Color Segmentation service please read the "Courseware Tutorial 4 Part 2"

You should not need to use any C# in this lab and stick purely with VPL, although you might need to write some fairly complex looping structure in VPL to iterate through the result list of the Color Segmentation output.

WebCam Driver / Robotics Studio Bug

We have noticed that after installing the webcam driver, the run dialog of Robotics Studio will no longer show debug messages and claim that "The attempt to access the directory service of the DSS runtime was failed". We currently don't know how to fix this bug, and while it is certainly annoying you will still be able to see debug messages through the web-interface, which is sometimes preferable anyway because it doesn't truncate long horizontal messages.