

MIDTERM REPORT

ELISE MCELLHINEY, ADAM VAN HAL

1. PROGRESS

1.1. Devices.

- Imaged two identical Raspberry Pi 3's with ubuntu mate
- Installed openCV on Raspberry Pi 3's
- Reimaged the intel UP with ubuntu 16.04
- Software will be compatible with multiple devices so long as they have installed the dependencies of ROS and openCV

1.2. Vision.

- We have established that openCV runs successfully on the Raspberry Pi 3
- The Raspberry Pi 3 runs two simultaneous webcams in order to produce stereo images
- As of now, the images are too slow to be useful in navigation as the pair of images takes about 10 seconds to run. This is due to the fact that the cameras are reinitialized for each set of images. Implementation of ROS nodes to allow a single initialization and images to be produced periodically should improve the timings.
- If we want updates every 10cm and to have the robot traveling at the maximum 0.5 m/s we will need to update every 200 ms.

1.3. Robot.

- Robot car is implemented with a Junior Runt Rover chassis and motors and is currently controlled by an arduino uno.
- The arduino uno can receive commands through serial inputs.
- Commands are configured to allow for forward, backward and turning motions on the robot car.
- The robot car can travel about 0.5 m/s with the current implementation.

2. PLAN

4-8-2017.

- ROS installed and running on both the Raspberry Pi 3 and the Intel UP.
- OpenCV running on the Raspberry Pi 3 with ROS nodes
- Intel depth map camera working on the Intel UP

Date: April 2, 2017.

4-15-2017.

- Raspberry Pi 3 producing stereo depth maps using dual webcams
- Intel UP producing depthmaps
- Calibrate distance of depth map images
- Analysis of resolution of depth map images
- Timing analysis of production of depth maps

4-22-2017.

- Implement navigation system using depth maps for input
- Connect robot to navigation
- Have robot respond to navigation commands from controller
- Implement collision detection (Detect nearby surfaces and stop if one is too close)

4-29-2017.

- Improve navigation system and algorithms
- Utilize machine learning concepts for navigation
- Bug fixes

5-3-2017.

- Wrap up project
- Project presentation

3. DIVISION OF LABOR

With the scope of the project as it is, we expect to both work on most components of the project. The listing below outlines parts of the project that we claim and will be responsible for fixing if unexpected problems occur.

Elise

- Implementation of physical robot
- Controls that translate serial commands into robot actions
- ROS module implementation
- Intel UP board and depth camera implementation

Adam

- OpenCV interactions
- Raspberry Pi 3 and OpenCV processing of dual webcams into depthmaps
- ROS interactions with OpenCV