INDIVIDUAL LAB REPORT #10
11/12/2020

AWADHUT THUBE

Team G - The Pit Crew

Team Members: Alex Withers, Justin Morris

Individual Progress

For the final progress review, we wanted to integrate and test our entire system. Our goal was to perform a dry run of our Fall Validation Demonstration tests. The two tests that we had for this progress review are as follows:

1. Simulated Mission Test
2. Terrestrial Pit Edge Detection Test

This time around, we had to work together most of the time for integrating the various subsystems. In terms of the simulated mission test, I spent some time refining our decision-making code which allows the rover to stop. I worked with Justin to integrate our two conditions (unsafe slope and shear brink) in our simulated environment. I also made sure to add our image capture program inside the simulator. This allowed us to capture one image at each vantage point near the pit edge. Once this program was included, our simulated test was pretty much complete as we had already integrated the brinkmanship subsystem with the navigation and planning subsystem.

Most of our time in the last two weeks was spent integrating software and hardware in real life. I spent a lot of time integrating the updated heuristic for brinkmanship on the real rover. Our updated heuristic uses the IMU to keep identify unsafe slopes. I wrote a program that used the IMU data to transform our point cloud appropriately before generating the triangle mesh. In the spring semester, I had written a ROS node that used the stereo images from the realsense to generate point clouds. I reused this piece of code and filtered outliers from the point cloud. The outliers were filtered using the average position of the point cloud and the deviation of individual points from this average position. Additionally, to make the decision making more robust, the triangle mesh is divided into 9 parts which form a 3x3 grid. This allows identifying the location of the brink and the location of unsafe terrain with respect to the rover with sightly higher resolution. For progress review 11, I also had to convert the recorded rosbags from the rover's perspective into MP4 videos. I used OpenCV to achieve the same.

Fig: Images from our test at Gascola

When our brinkmanship subsystem was integrated with the navigation subsystem, Justin and I integrated the camera operation subsystem on the rover. This code is essentially a library that operates the pan-tilt motors and enables us to use the realsense camera to capture and save images when required.

Next, I configured our system for reducing setup time between experiments. I assigned static USB ports for our motor drivers. Additionally, I made a few changes to our computer settings to provide appropriate USB permissions for the motors and for the IMU. All different ROS nodes were also merged into one launch file. Finally, I also added a helper code to record the entire image stream as seen from the rover perspective. This code used the Pyrealsense2 library to log the stereo images and all related metadata directly into a rosbag.

Challenges Faced

This time, our team faced a significant number of technical and non-technical challenges during the buildup to our tests and also during the actual tests. First up, the point cloud generated from the stereo images was a lot noisier and required some filtering. Integrating different codes and making sure that all data was correctly transformed and passed between subsystems was also a challenge. We had to spend a lot of time running unit tests for individual subsystems.

In terms of non-technical challenges, some of the parts we ordered did not arrive in time for our first field test. That included a USB monitor which is very essential to visualize data out in the field. Another non-technical challenge was that we had to scout a different location to perform our tests. That's because many people were using the original site for recreational purposes. Also, It was difficult for three people to cover all the things we needed. We had to run our system, make sure everything was functional, record videos, and also have a dedicated person for holding on to the rope tied to the rover.

Team Work

Alex Withers: Alex made sure that the navigation and planning subsystem was functional on the real rover. He generated a low-resolution map with three different waypoints for the rover and set up the TEB local planner on the real rover for navigation. He had initially scouted the site for identifying potential locations where we could perform our tests. Alex has coordinated with the moon ranger team and our project mentors to make sure that we had permission to test at Gascola.

Justin Morris: Justin wrote the skeleton code for our camera operation subsystem. He got the rover running and tested our navigation subsystem in his backyard. Justin also supported me in integrating the different nodes which we have into one full system. Justin has made modifications to our pit edge detection heuristic to make it more robust when performing in real-life conditions.

Future Plan

Before the FVD, we plan to go to Gascola one more time. Our aim for this field test will be to perform some more dry runs and collect some data for analysis of the system's performance. Specifically, we want to see the effect of varying camera tilt on the stopping distance of the rover from the pit. We will also try and record more rosbags containing images from the rover's perspective. We would be able to use these rosbags later to visualize point clouds and triangle meshes as seen by the rover. We also wish to do some analysis with respect to the time required for covering multiple waypoints and the changes in the stopping distance of the rover with varying thresholds concerning the triangle mesh. We are working as a team to distribute the tasks among us. Another task on the list is to update the website for the upcoming review.