Individual Lab Report

Bikramjot Singh Hanzra
Team B – Auto Pirates
Teammates – Tushar Chugh, Shiyu Dong, Tae-Hyung Kim, William Seto
ILR05

November 25, 2015

Contents

List of Figures

1 Individual Progress

In the last week, I worked upon writing the code to display the RADAR data on the Occupancy Grid Map using the OpenCV [1] library. One of our fall validation experiment is to *show detected objects from RADAR image while showing recorded video*. So, I was assigned this task by my team. In the subsequent text, I will discuss how I approached this task.

Figure 1 shows the Occupancy Grid Map where the white pixels represent the river. The RED circle in Figure 1 shows the position of the boat in the Occupancy Grid Map. Shiyu and William working on building the Occupancy Grid Map with inputs from Tushar.

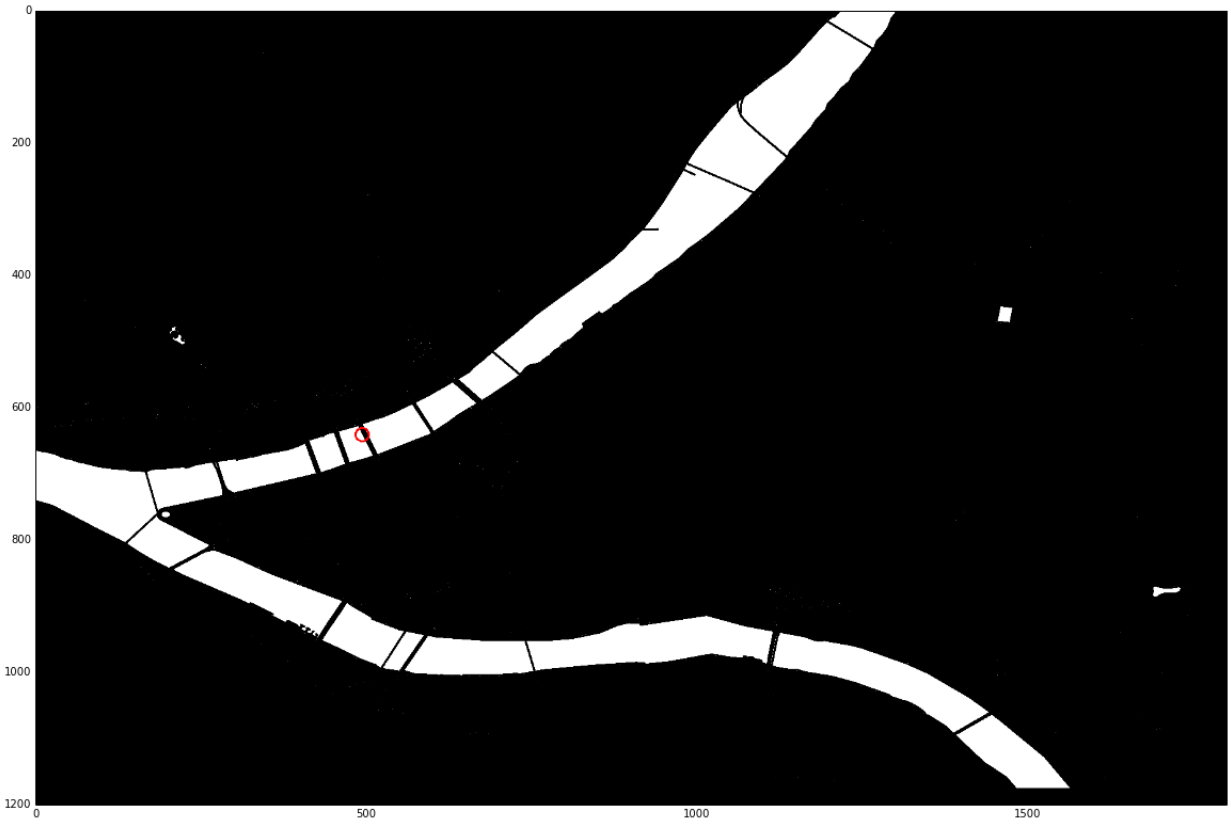

Figure 1: Location of the boat in the river. The RED circle shows the position of the boat in the Occupancy Grid Map.

The IMU provides us the `latitude`, `longitude` and `yaw` information of the boat. The IMU data is published on the `\recboat\span_pose` topic. The listings below shows the data published on the `\recboat\span_pose` topic.

```
header:
  seq: 294911
  stamp:
    secs: 1447777053
    nsecs: 481296021
  frame_id: ''
ver: 0
sec: 0
```

```
usec: 0
lat: 40.4468470896
lon: -79.9999770025
north: 4477834.92804
east: 584803.337073
z: 185.295959754
roll: 0.0301739536226
pitch: 0.0500648021698
yaw: 0.98569303751
vel: 7.6191444397
nsat: 6
ins_stat: 3
sol: 0
pos: 16
ins_str: Solution_Good
sol_str: Computed
pos_str: Single
```

So, I extracted the `latitude`, `longitude` and `yaw` information from the `\recboat\span_pose` topic. After getting this information, the next step was to render the RADAR data on the map. To do this, I needed to rotate the RADAR image in accordance with the map. The `yaw` information from the IMU was used to rotate the image.

Figure 2 shows the original raw RADAR data image where the white pixels represent the obstacles.

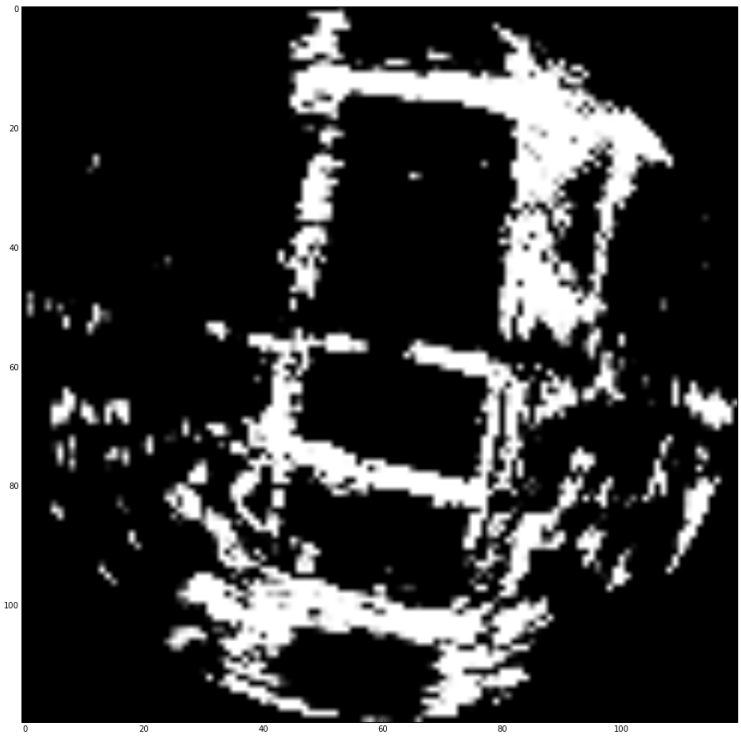

Figure 2: Original Radar Data Image showing obstacles (The image is downsampled by a factor of 5)

Figure 3 shows the rotated RADAR data image.

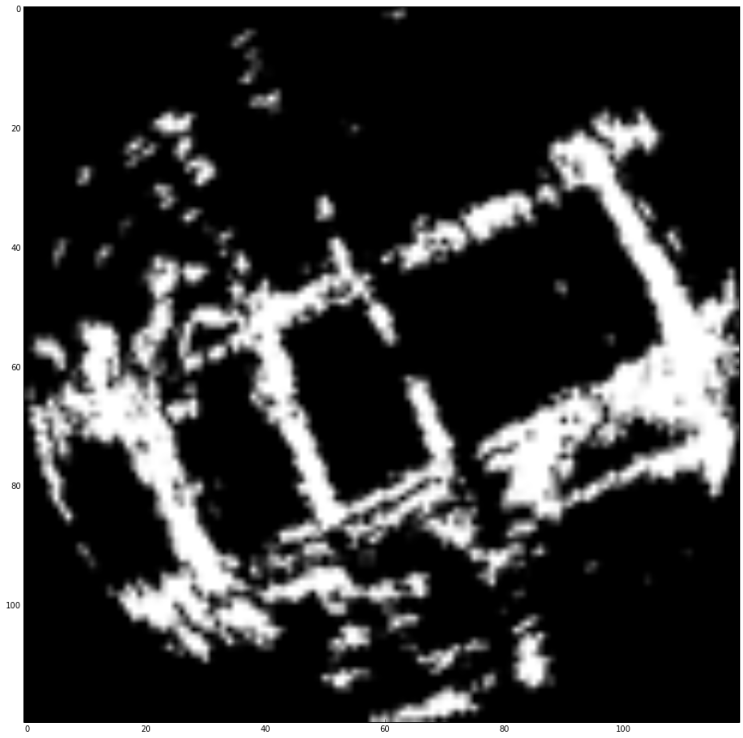

Figure 3: Radar Data Image after rotation (The image is downsampled by a factor of 5)

After rotating the RADAR image the next step was to render the RADAR data on the Occupancy Grid Map. To perform this task, I used the functionality provided by OpenCV's `imgproc` module. The `imgproc` module in OpenCV provides convenient functions to manipulate pixel values.

Figure 4 shows the Occupancy Grid Map with RADAR data rendered on it. The RED pixels in Figure 4 represent the RADAR data.

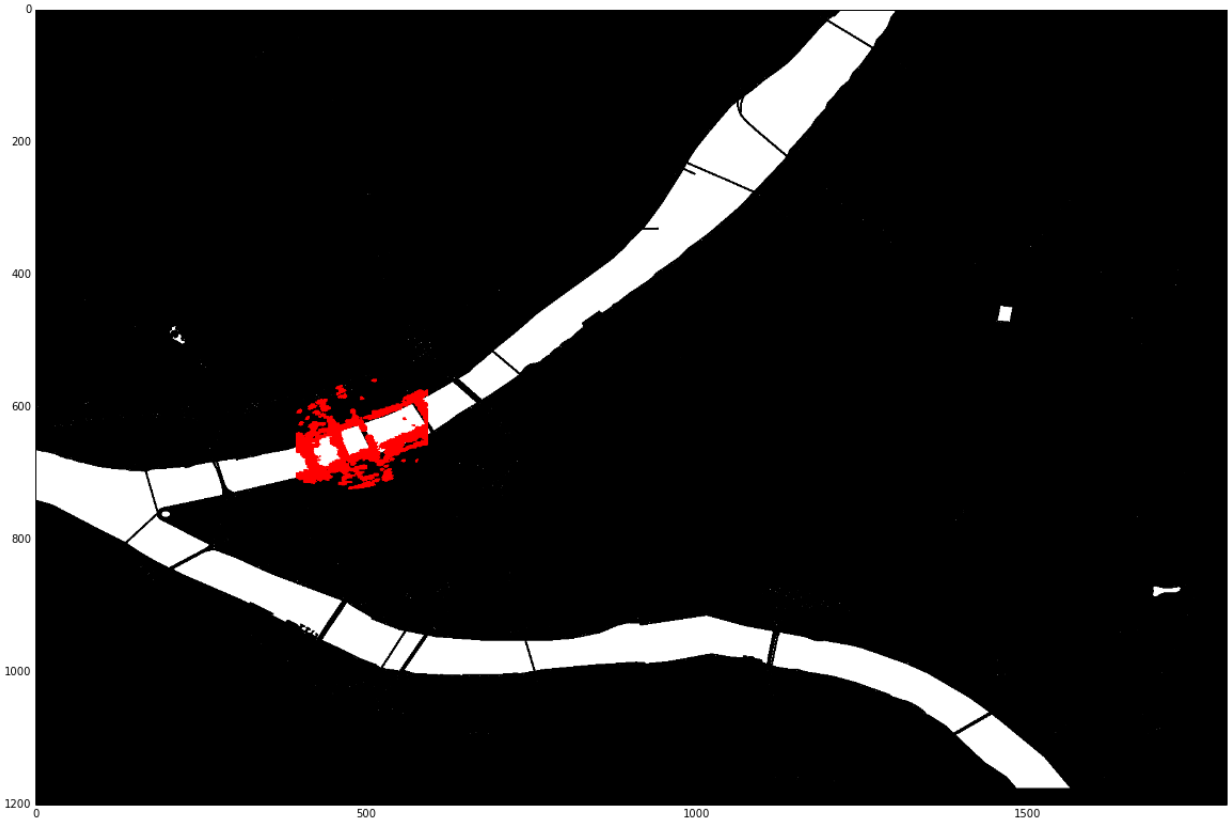

Figure 4: Grid Occupancy Map with RADAR data rendered on it

During the field test, I was assigned that task of recording the RADAR data, RGB camera data, and IMU data. I wrote the code for the same. We collected around 5GB of raw data during the field test.

2 Challenges

During the field test, we faced some problems while controlling the boat. Currently, the boat can be moved from one GPS location to another by passing a command specifying the locations. Once, the boat reaches the destination it starts to move in a circle around the destination due to GPS location errors. This created a lot of trouble while running the boat autonomously. We need to fix this issue before our next field trial which is scheduled to be in the next week. Also, this weekend we wanted to work on the boat to fix this problem but were unable to do so because the boat was shifted to the NREC garage due to the cold weather.

While rendering the RADAR data on the Occupancy Grid Map, there were issues that the RADAR data was not superimposing correctly with Occupancy Grid Map. Sometimes the actual bridge location is slightly at an offset from the position estimated by the code. This is due to errors in GPS locations. We need to incorporate some image processing functionality to mitigate this problem. I have a few ideas on this and will try to implement them this week.

3 Teamwork

The work done by the rest of team is as under –

- **Tushar Chugh** – Tushar worked on improving the GUI. He also worked on performing path planning using Occupancy Grid Map.
- **Shiyu Dong** – Shiyu worked on plotting the gps coordinate on the Occupancy grid mapping using `rviz`.
- **Tae-Hyung Kim** – Tae-Hyung worked on compiling the lattice planner in SBPL to connect with the `navigation` package.
- **William Seto** – William worked on creating the map in `qgis`.

We all worked collectively to perform the field test. Also, Tushar with inputs from Shiyu worked on soldering the PCB.

4 Work Overview for Coming Week

In the next week, I will be concentrating on writing the ROS node that updates the Occupancy Grid Map whenever new RADAR data is published. Also, I will be working with the team to finish the Fall Validations Experiments. We are mostly done for the Perception Fall Validation Experiments and some work needs to be done in the Path Planning Fall Validation Experiments. Also, we will be preparing for our second field test.

References

[1] `OpenCV` Official Website
`http://opencv.org/`