Individual Lab Report

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

November 26, 2015

1 Individual Progress

We have successfully carried on our first field test on Nov.17 and logged a lot of data on river including the radar data, camera image and IMU data.

In this week, I mainly worked on overlay the IMU data on the occupancy grid map and I have successfully plotted and updated the recorded GPS location of the boat.

1.1 IMU Data

Some preparation work is done before I can plot the boat position on the occupancy grid map.

The first thing is to figure out what we can get from the IMU data. It's shown as table 1

Data Type	Variable
float64	lat
float64	lon
float64	north
float64	east
float32	roll
float32	pitch
float32	yaw

Table 1: IMU Data

In the table, the lat and lon is GPS coordinate for latitude and longitude. And the north and east is UTM coordinate for northing and easting. The GPS coordinate and UTM coordinate are all used to describe the location and we can do transformation among them.

1.2 GIS, Map, and World File

William worked on how to transform a existing map into the occupancy grid map that we can know the GPS coordinate of each pixel. William suggested a good way to generate the map using a GIS (Geographic Information System) software called QGIS. Using that software, we were able to select an area from the map and also generate a corresponding world file. With the world file, we can tell the coordinate (GPS/UTM) of the top-left pixel and the length of each pixel in x and y direction of the map and the orientation of the map frame.

For example, one world file looks like this:

```
0.000043
0.000000
0.000000
-0.000043
-80.021197
40.474416
```

The meaning of this world file is listed as follows:

- Line 1: Pixel size in the x direction in GPS unit.

- Line 2: Rotation about y axis
- Line 3: Rotation about x axis
- Line 4: Pixel size in the y direction in GPS unit
- Line 5: x coordinate of the center of the upper left pixel in GPS unit
- Line 6: y coordinate of the center of the upper left pixel in GPS unit

1.3 Mapping the GPS Coordinate to the Pixel Value

Since we know the world file of the map we generated, it will be easy to map the GPS coordinate we recorded in the field test to the map in Rvis. The mapping equation is given as follows:

$$p_x = (x_i - x_0)/i_x \tag{1}$$

$$p_y = w - (y_i - y_0)/i_y \tag{2}$$

where p_x, p_y is the pixel position on the map, and pixel $(0, 0)$ means the left-bottom pixel.

x_i, y_i is the GPS measurement of each time.

x_0, y_0 is the GPS location of top-left pixel given by the world file.

i_x, i_y is the pixel size of x and y direction in GPS unit.

w is the pixel width of the image.

1.4 Render the Map and Plot a Marker in RVIS

To render the map, we need a bit map file and a map.yaml file shown as below:

```
image: map_bri.bmp
resolution: 1
origin: [0.0, 0.0, 0.0]
occupied_thresh: 0.65
free_thresh: 0.196
negate: 0
```

In the map.yaml file, we can specify the bit map image we will use, the resolution and the origin of the map.

Use *rosrun map_server map_server map.yaml* will enable us to render the map in RVIZ.

To plot the marker in RVIZ, we need to specify the marker size, orientation, color, frame, etc in the following code:

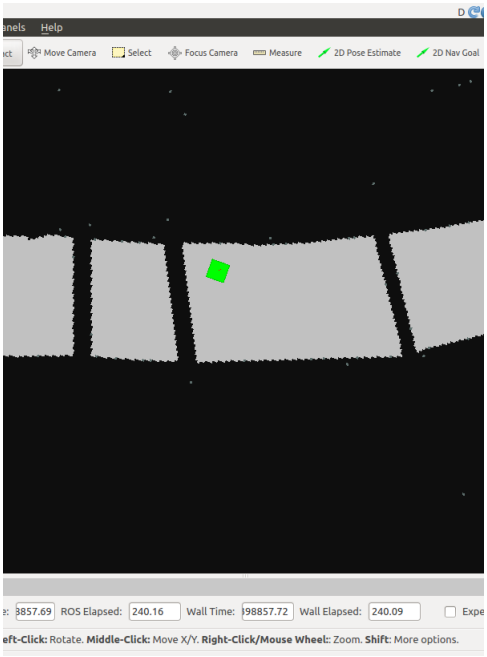
```
points.header.frame_id  = "/my_frame";
points.header.stamp = ros::Time::now();
points.ns  = "points_and_lines";
```

```
points.action  = visualization_msgs::Marker::ADD;
points.pose.orientation.w  = 0.0;

points.id = 0;
points.type = visualization_msgs::Marker::POINTS;

points.scale.x = 8;
points.scale.y = 8;

points.color.g = 1.0f;
points.color.a = 1.0;
```

1.5 Subscribe to the IMU Topic and Publish to RVIS to Visualize

After we initialize the marker and map in RVIZ, the next thing to do is to subscribe to the rostopic for imu, send the value to the marker we made, and publish the marker to RVIZ.

The final output is shown as figure 1, where you can see a point marker is moving on the occupancy grid map.



2 Challenges

The challenge I faced for this week is mainly when I try to map the GPS coordinate into the pixel on the image.

So the first thing is that it's important to set the resolution in the map.yaml file. If the resolution

is set to 1, that means one grid is corresponding to 1 meter, so we don't need to make any changes for the coordinate. If it's not 1, we need to resize the pixel coordinate we calculated.

Another challenge I faced is that the map file uses different expression with plotting. In plotting, we view the bottom-left point as the origin and in the map file, we view the top-left point as the origin.

3 Team Work

- Tushar improved the GUI so now we are able to send waypoints commands in the GUI. He also ran the SBPL library in the occupancy grid map.
- Bikram worked on display the radar data on the occupancy grid map with the OpenCV library.
- TaeHyung worked on planning the field test and he also tested path planning in SBPL and connect it to the ROS navigation package.
- William worked on getting the occupancy grid map and word file based on the QGIS program.

4 Future Plan

For the team, we need to finish planning the path on the occupancy grid map and integrate everything together for fall validation.

For myself, I need to work with the perception team and try to remove more noise on the radar image but remain small obstacles like buoys.