Team F: FALCON EYE

Individual Lab Report 4

Progress Review 3

Rahul Ramakrishnan

Team

Pulkit Goyal
Yuchi Wang
Pratibha Tripathi
Danendra Singh

November 10, 2017

1 Individual Progress

My primary contribution to the progress was to perform obstacle detection using Velodyne LiDAR and map them in Rviz. Another major task was the conversion of the GPS data into NMEA format and publish the coordinates as a topic in ROS for it to be accessed by Husky node. I also worked with Pratibha for the fabrication and mounting of parts designed previously in CAD.

1.1 Velodyne Obstacle detection

We finally got the Velodyne!! Now that we were able to play with real-time pointcloud data from Velodyne, I tested the driver created last week and it was working as expected. We could visualize the pointcloud as shown in Fig.1 without any hassle. There wasn't any noticeable lag between the display in Rviz and the changes in the environment.

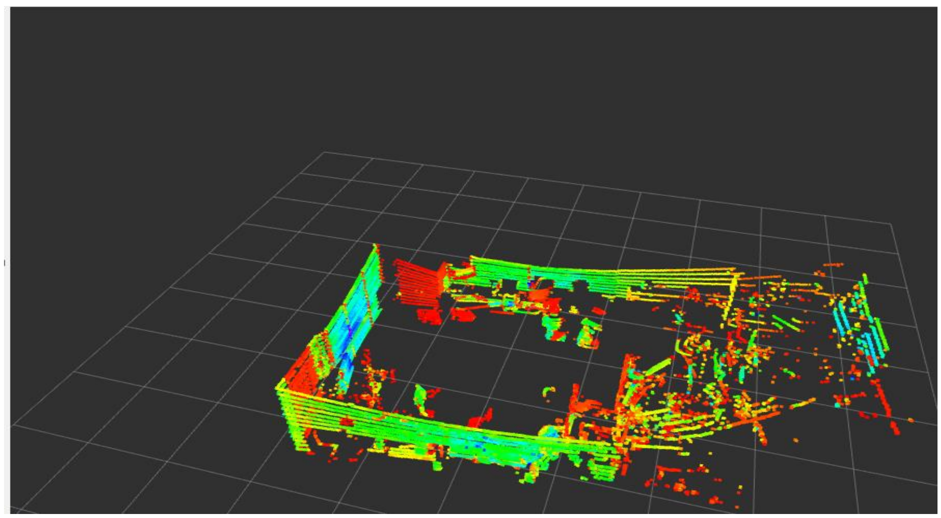

Fig.1 Pointcloud visualization of Tele-supervised autonomous robot's lab

For the obstacle detection, I decided to go with the Velodyne_heightmap. This algorithm considers everything in the environment in Velodyne's field of view, above the height specified by us as an obstacle. As in the case of Velodyne driver implementation, few parameters needed to be tweaked to obtain optimal detection of the obstacles. The detected obstacles were mapped across the same dimensions as the pointcloud map and visualized in Rviz as in Fig.2.

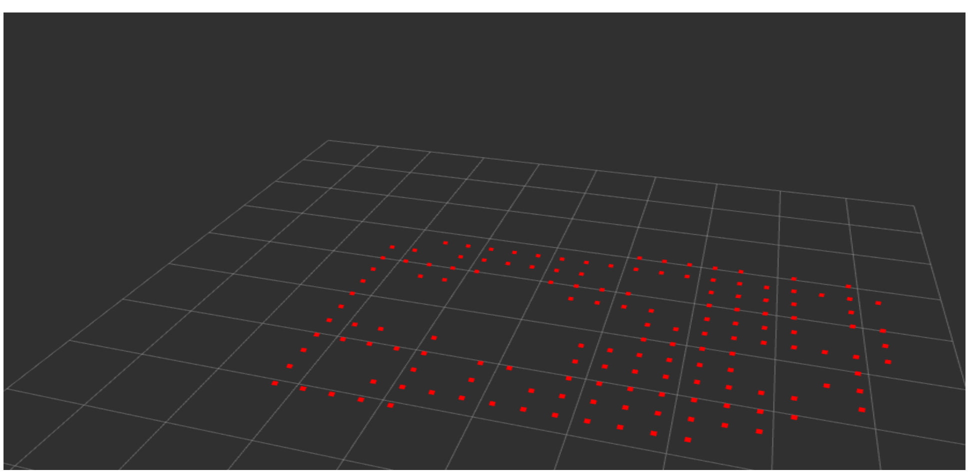

Fig.2 Map of obstacles detected in Fig1

This obstacles map (Fig.2) with little more work should suffice for navigation avoiding obstacles. But since we are planning to embed full autonomy into husky, we need a better detailed map of our surroundings which we are planning to proceed for in the spring semester. Also, the boundaries of the clear path have also been visualized as shown in Fig.3.

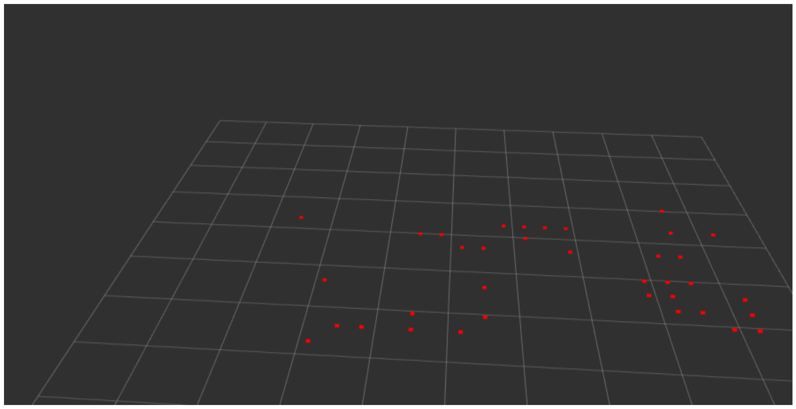

Fig.3 Map of clear path boundaries detected in Fig1

1.2 GPS data conversion

Yuchi had previously worked on acquiring the GPS data from Radiolink SE100 via Arduino in GNSS format which was the default format that the GPS was publishing. I used a driver *nmea_navsat_driver* present online to acquire GPS data into ROS. The driver gave me the GPS data as displayed previously in Arduino serial monitor.

There was another node in the same driver that is supposed to give the data converted to required latitude and longitude format but that did not work. This was because the format of our GPS data was different than the one the driver was meant to be used for. Hence, I created a similar node that performed the same calculation for conversion of the GPS data using the format that our GPS gave and published the desired GPS data as a topic so that it can be accessed by the Husky node for its navigation purpose. Fig.4 shows the published GPS data.

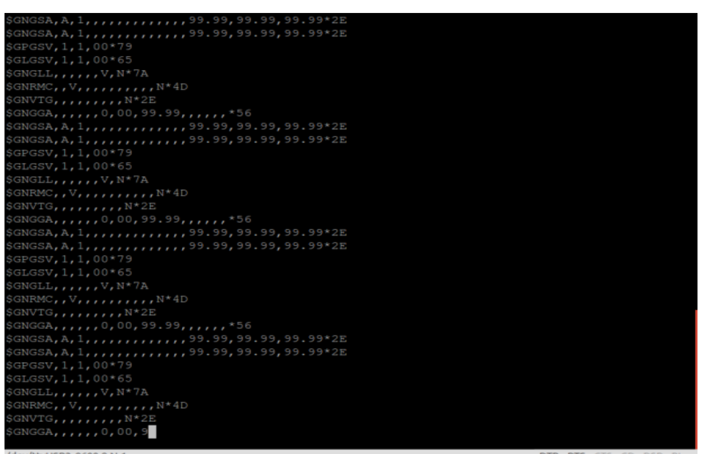

Fig.4 GPS data from Radiolink SE 100

1.3 Mechanical Fabrication and Mounting

Pratibha and I used the pre-existing mounts we got from NREC, modified them to our need and put them together to fit our design and stability needs. We fabricated the mount required for Velodyne, drilled holes appropriately and mounted it firmly in the front part of the husky. During this process we ensured minimum vibration of the mounts to reduce motion artifacts for data acquired from the Velodyne. We mounted the point gray camera and the platform to carry the drone. All these parts were made in accordance to the size constraints designed in CAD which ensures complete field of view for LiDAR and sufficient take-off space for the drone. Fig.5 shows the mechanical design of the system.

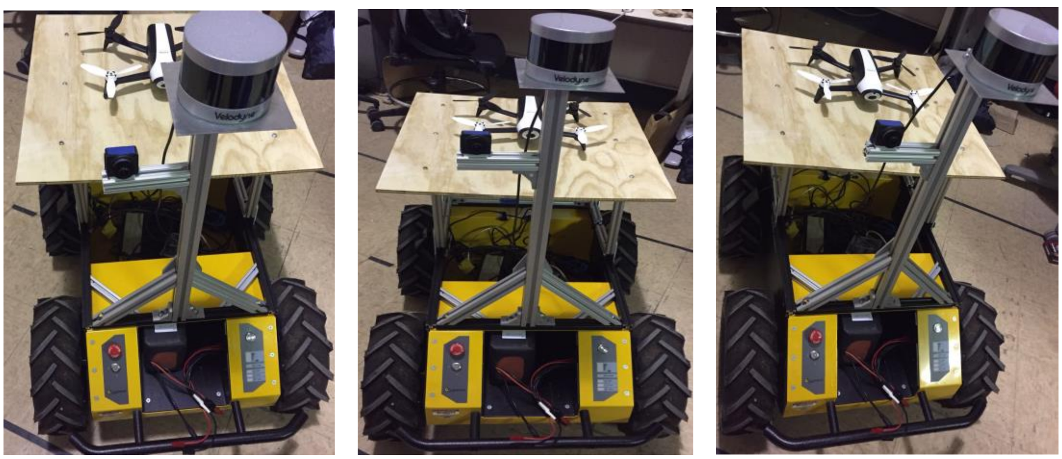

Fig.5 Mechanical mounting for Velodyne, drone and camera

2 Challenges

Technically even the simplest of work turns out to be challenging every single time. To quote some main instances, mounting of the mechanical parts took us 9 full hours. Even though the setup was well planned and systematically executed, they gave us a hard time dealing with small nuts and bolts. It made me realize never to under-estimate a mechanical job.

It took me quite a while to figure out the reason why the nmea_driver did not give us the converted co-ordinates as no error was displayed while compilation. Solving it was easier than figuring out the problem.

Dealing with the Velodyne data has been a challenge of its own considering the number of parameters associated with each individual variable. Say for obstacles, it’s height acts as a threshold, whose resolution matters for efficient detection which is in turn dependent on the rotational frequency of the lasers in the puck. Getting hold of these is going to take a while but hopefully it wouldn’t be a challenge anymore in the near future.

3 Teamwork

Yuchi and Danendra took up the work of high level control of the drone given a GPS coordinate and were able to do so until the drone crashed due to loss of connectivity. Pulkit and Yuchi setup the common network for the entire system to be connected to a single network. This is a vital part of our project. They have been successful in connecting the husky and remote computer to a common network but aren't able to connect the bebop 2 to the network as acts only as a host. Pratibha and I worked on the fabrication and mounting of the mounts. Pratibha also worked on the PCB design assisted by Danendra. Pulkit and Pratibha worked on the fusion of IMU and encoder data and acquiring that data into Husky nodes. They successfully visualized them in Rviz.

4 Future plans

I will work alongside Yuchi on perfecting the localization of April tags with respect to the home frame since it is very unstable now. This is extremely important as our system's performance is dependent on this. Pulkit and Pratibha will work on the network part trying to get the bebop 2 connect to the common network as a client. Yuchi and Danny will continue to work on the flight of drone given the GPS locations. This is a part of our FVE and hopefully will be completed by next progress review.