Harikrishnan Suresh

Team C: Fly Sense

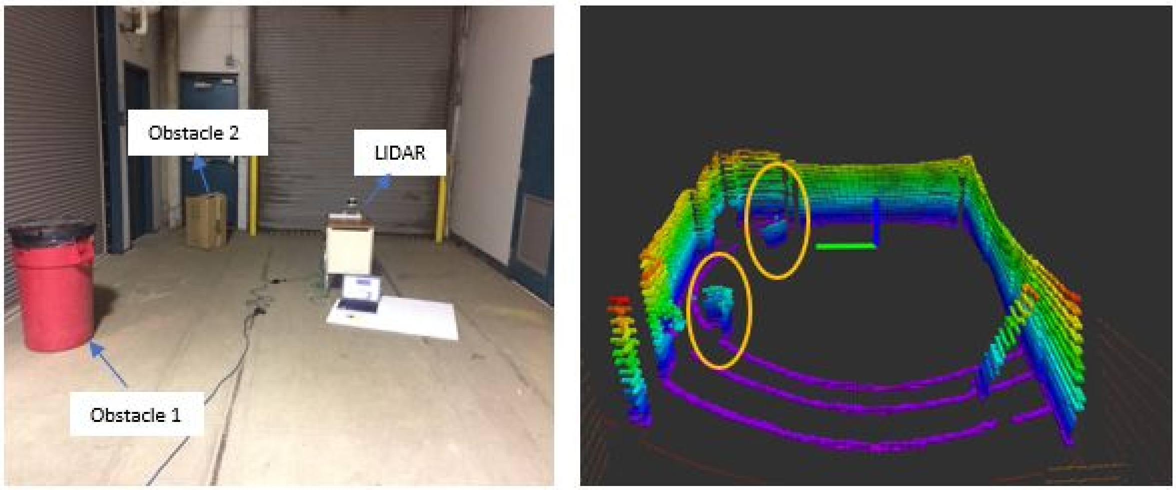
Teammates: Shivang Baveja, Nicholas Crispie, Joao Fonseca, Sai Nihar Tadichetty

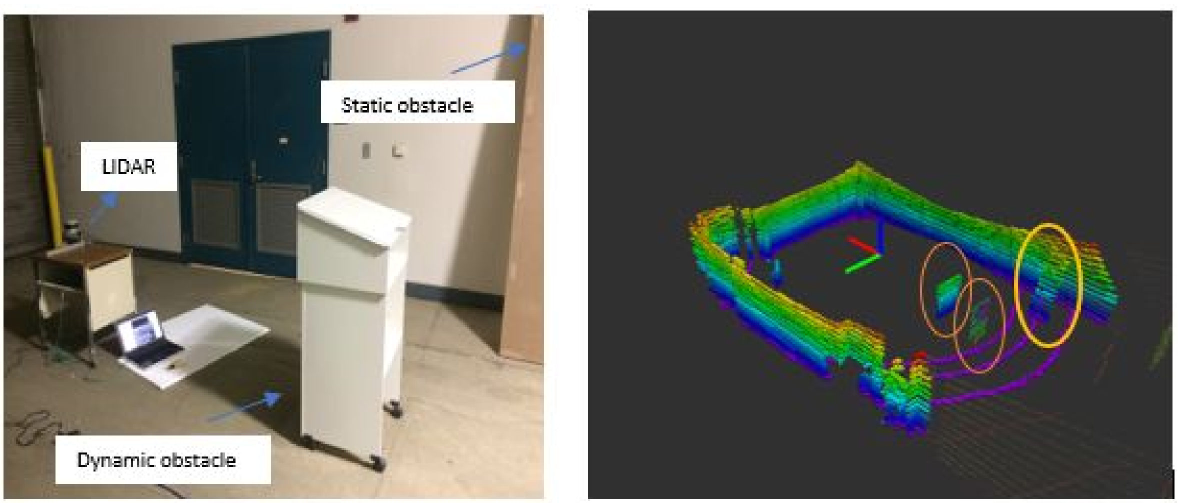
ILR02

Oct. 20, 2017

Individual Progress

This week, I was primarily involved in getting the 3D map generated for various samples of LIDAR data in different scenarios. A series of tests were conducted outside the MRSD lab to collected the data, which will be described in section 3. I was successful in getting a complete 3D map generated for all the samples using the octomap_server node in ROS. This also involved segmentation of the environment into occupied cells and free space. The visualization of point cloud data and 3D map was done in Rviz, and is shown in Figure 1 and Figure 2.





The key takeaways from the above experiments were:

1. Octomap_server was successful in detecting the walls and all the static obstacles
2. The octomap_server failed to generate the moving obstacle in real time
3. 3D mapping range was very limited
4. Random markers were generated, which need to be filtered out
5. Floor detection parameters need to be optimized

Along with this, I also looked into other possible approaches to this problem. I felt a possible solution was to implement real time detection and tracking of all the obstacles in the environment, and then perform some form of 2D overlay over all those obstacles. This would be a more feasible method to represent all the obstacles in the environment, including the fast moving ones. I looked into some of the work done by Dr. David Held in this field, mainly in real time 3D segmentation and robust anytime tracking. Since his ROS code for the same is available as open-source, I decided to discuss with our project mentor Vishal before taking any decision on the same.

Challenges faced

1. The octomap algorithm failed to generate 3D representations for the dynamic obstacles. This is seen in Figure 2, where the 3D markers of the moving obstacle are seen separate from the point cloud data.
2. In order to improve the update frequency of the octomap and achieve real time mapping, the parameters in the configuration file were modified based on suggestions found online. Parameters like point cloud hit/miss probability, point cloud range max/min, ground plane filter etc. were changed This did not produce any huge improvements in the obstacle mapping problem. Since I was not completely aware of the algorithm and its code, I did not understand the impact of various parameters on the algorithm. As a result, I decided to read the paper "Hornung, Armin, et al. "OctoMap: An efficient probabilistic 3D mapping framework based on octrees." *Autonomous Robots* 34.3 (2013): 189-206" based on which the code was developed.
3. The Velodyne PUCK was also not able to capture obstacles of very small sizes like cables, which will be a huge drawback for us as we plan to focus on the tail rotor problem. As a result, we need to look into the performance of Velodyne HDL-64E which Near Earth Autonomy uses in its helicopters. We also need to check options like use of camera for obstacle detection.

Team work

Tests outside MRSD Lab

My primary collaboration was with the sensing and integration team (Shivang and Nick). A series of tests were conducted outside the MRSD Lab to evaluate the performance of the Velodyne PUCK and collect samples of point cloud data (in bag files). These samples could be used to test various algorithms we plan to implement till our Velodyne sensor is obtained from the sponsor. The scenarios in which the tests were conducted are listed below:

- Environment with only static obstacles, but of different sizes
- Adding a dynamic obstacle (moving podium) to the above environment

Meeting with Vishal

We had a meeting with Vishal to give an update on the work done done so far and also get some feedback on the various approaches to tackle the problem.

The approaches listed by us were:

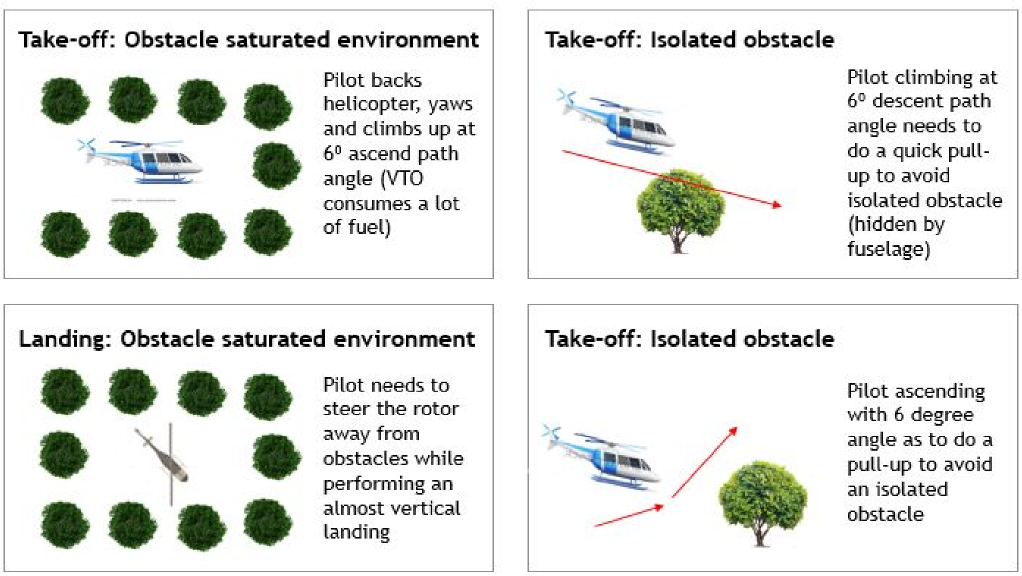
1. 3D mapping and segmentation, which then gives all the obstacles in the environment

2. Real time obstacle detection and tracking, and perform 2D overlay on the tracked obstacles

Since I had already implemented the 3D mapping approach, we presented it to him and discussed the challenges we faced. He suggested us to try with different grid sizes of the octree and determine the ideal grid size that produces a 3D map of reasonable accuracy but in real time. We also got a positive feedback on the 3D mapping approach, as our problem focuses only on obstacle representation and does not require accurate tracking. The sensor would give us details of all the obstacles in its range, irrespective of whether it is static or dynamic.
We were also provided with a few materials to read, which may prove useful in solving our problem.

Meeting with Lyle Chamberlain (Near Earth Autonomy)

Lyle Chamberlain shared his experiences based on his research on the tail rotor problem. He had a very similar concept for the helicopter assistance, but focussed on providing visuals on the Garmin display. He seemed to be fascinated about the possibility of Head-Mounted Displays (HMDs), and suggested us to invest more time on getting the interface ready. Based on his suggestions, we also listed down 4 different scenarios where the bird's eye view mode will be tested.



Discussion on AR

Nihar and Joao performed a series of experiments to evaluate the effectiveness of the Hololens for our application. Due to some of the issues like limited field of view, discomfort for pilots due to the weight and failure of voice detection in noisy environments, they concluded that the Hololens would not be the best solution. After a team discussion, we decided to procure the Epson BT-300 and ordered it.
They also met with some of the experts in CMU like Jean OH, Aaron Steinfield who gave us positive feedback on the approach taken by us regarding HMDs. They also got some insight into the pilot preferences, and decided to go ahead with developing a sound recognition system that would work in the helicopter's noisy environment.

Future work

For the next lab demo, I aim to improve the performance of the octomap. By reading the paper and understanding the code, more clarity will be received on the various parameters that affect the performance of the octomap. We also plan to work with the LIDAR data provided by Near Earth Autonomy, and implementing octomap_server for the same. We will perform more tests with lidar to establish how much buffering of data will be required to detect smaller obstacles at far away distances.

Our team will focus on development of basic interface with the Hololens till the epson is procured.