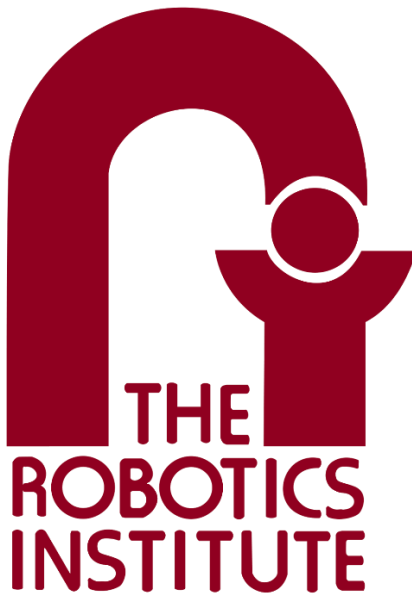# Individual Lab Report #4

VIVEK GOPAL RAMASWAMY

TEAM E- BEYOND SIGHT

Team Members: Chien Chih Ho, Pengsheng Guo, Oliver Krengel, Rohit Murthy, Vivek Gopal Ramaswamy

10th November 2017

Individual Progress:

As a part of the team, I was responsible for the following

1) Camera-LIDAR calibration.
2) Setting up Jetson Tx2.

Progress in detail:

1) Camera-LIDAR calibration
To get started, I had done the following

a) Overlaying ZED camera and LIDAR on Rviz
This was done, to see how effectively the data from the camera and LIDAR can be combined to gain more information.   For achieving this, I had written a tf publisher, which could subscribe to the topics published by ZED and LIDAR and publish the transform from the ZED frame to LIDAR frame as shown in fig 1 and fig 2.
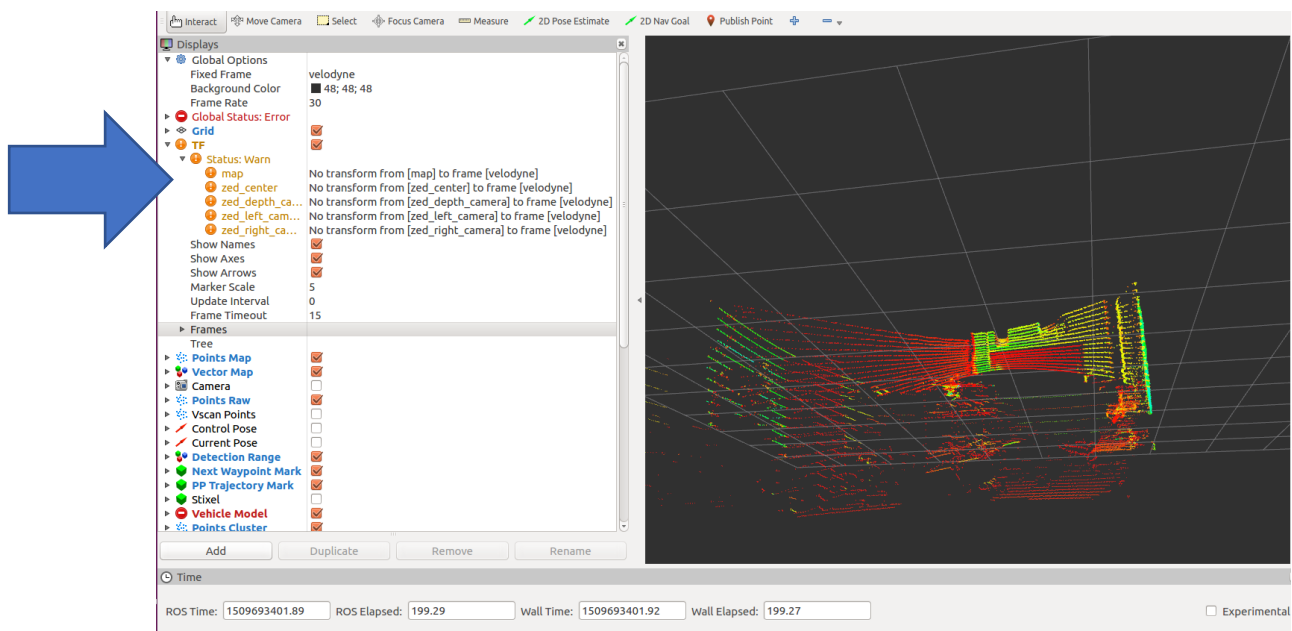

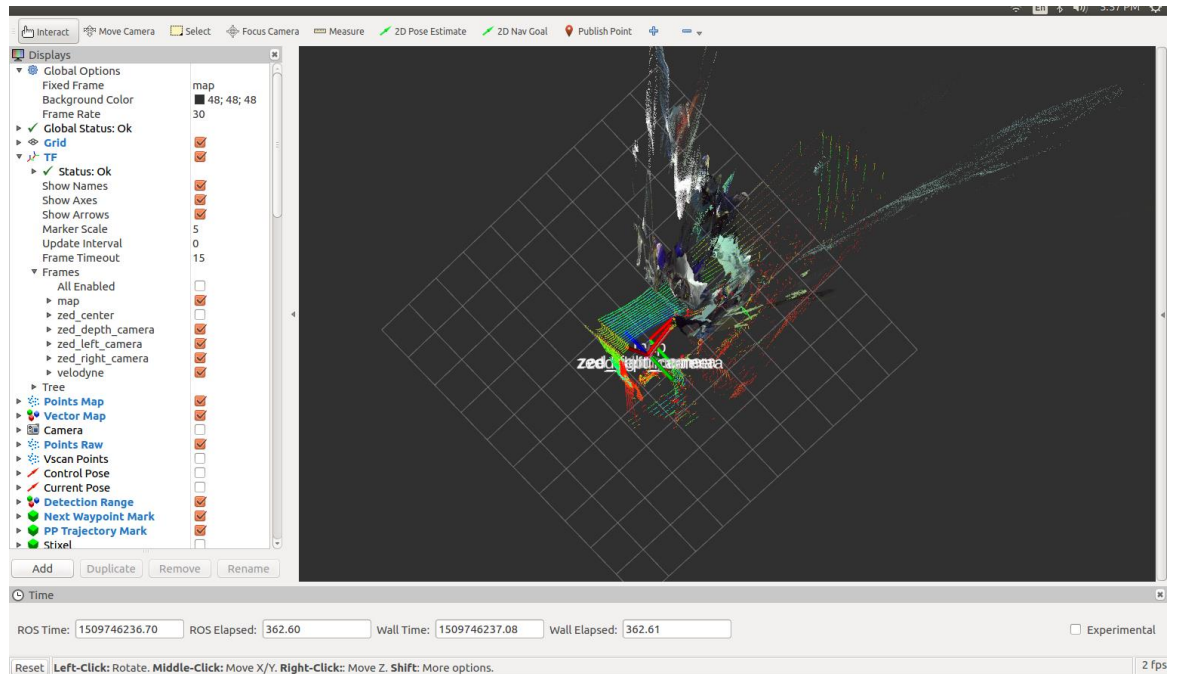Figure 1: Visualization of Camera and Lidar with inappropriate transforms

Figure 2: Inappropriate overlay of the camera and LIDAR in Rviz.

b) Using the Autoware toolkit for calibration

Autoware is an opensource tool [1], which can be used to calibrate ZED and LIDAR. The Autoware tool was set up by my teammate Chien Chih Ho. I had followed the manual provided by them to do the calibration. The calibration required the camera and LIDAR to be fixed in a place and required a large enough checkerboard, which could be detected by both sensors. I had used the large checkboard, which was already present in the B level NSH.

The following steps were involved in this process:
1) Setting up the sensors and the checkerboard.
2) Selecting the sensors to be fused.
3) Including the calibration file for LIDAR.
4) Recording the raw data from the sensors with the checkerboard at different positions and orientation in a bag file.
5) Playing the bag file and simultaneously grabbing frames, where the checkboard is not blurred and is clear enough for calibration.
6) Selecting Points in the checkerboard in the grabbed frame and mapping it to the checkboard visible on the LIDAR point cloud as shown in fig 3.
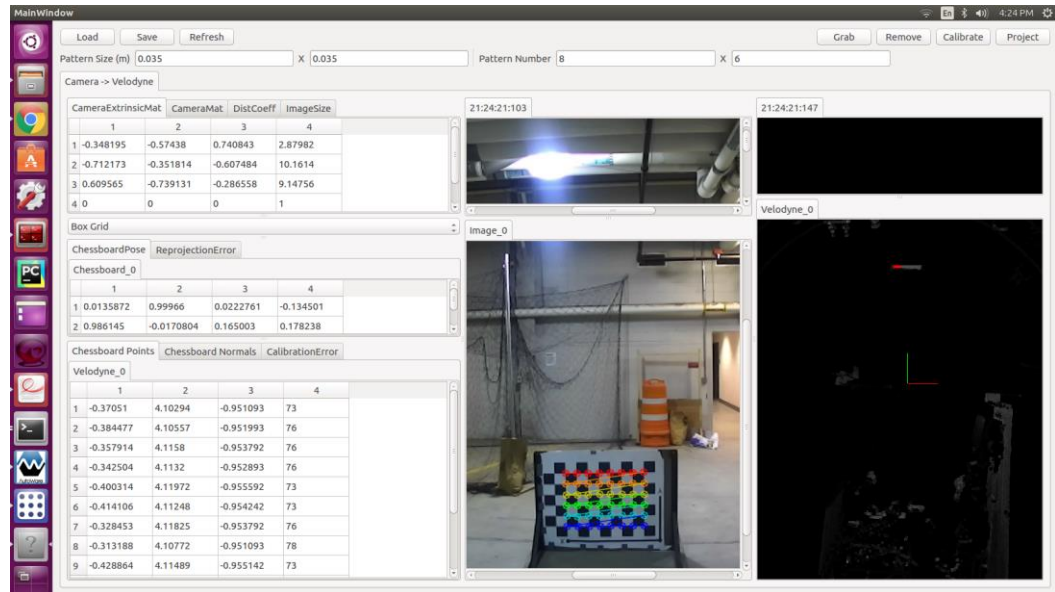
Figure 3: Mapping points from the checkerboard to LIDAR PCL.

7) Visualizing using the image plugin on Rviz provided by Autoware.

The reason for switching to a different method will be discussed in the challenges section.

c) Using 3D-3D correspondences

For the most part of this, I had referred to this paper [2], which had a unique technique of fusing the sensor using 3D-3D point correspondences with Aruco marker tags.

This method was suggested by our mentor, Professor David Held. For implementing this method, I had done the following things

a) Generated two square ArUco marker tags with dimensions of 205mm, and 10 mm padding each.
b) Made two square boards with the dimensions of 30 cm each.
c) Hanged out the boards at a 45-degree angle approximately, from a suspended wire as shown in fig 4.

Figure 4: Sensor Setup with ArUco Tags

d) Had set up the ROS packages [3] provided by authors of the paper referred above.
e) Had updated the config.txt which consisted of the intrinsic parameters of the ZED camera, image width, image height, tolerances in x-axis and y-axis.
f) Calculated the extrinsic parameter between the camera and LIDAR.
g) Launched the Velodyne.
h) Launched ZED camera.
i) Launched the Arcos mapping packages, provided to detect the ArUco tags.
j) Launched the transform. launch file for calibration.

2) **Setting up the Jetson Tx2**
I had followed the steps provided on their official website [4] for setting up the board and had installed Ubuntu 16.04 LTS on it.

Challenges faced

There are a couple of challenges which I faced during the development of the project.

1) **Calibration**

For calibrating using the Autoware toolkit, it needed at least 60 frames in which the checkerboard is rotated and translated in every possible direction. Though I had done this, it takes in a lot of time and thus was very difficult for me to spend more time in trying to get a perfect calibration. My partial results, which I got from using Autoware are shown in fig 5. The alternative method was suggested by Professor David Held, using 3D-3D point correspondences.
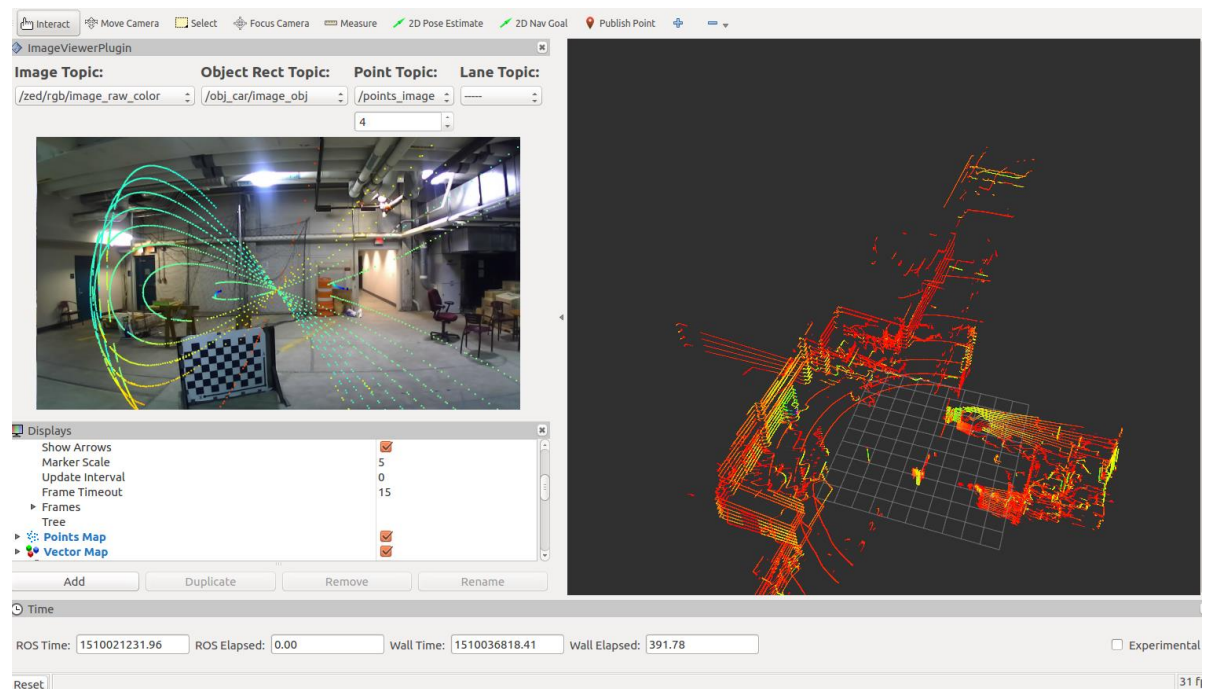

Figure 5: Calibration of LIDAR and camera.

2) **3D-3D correspondences**

Setting up the ArUco tags in such a way that it does not move was difficult to me, as the board always got drifted away. To solve this, I added tolerances in X, Y, Z axis to counteract the board movement.

The current issue which I am facing is that, when the ArUco package, and the lidar_camera calibration package is launched, I am not able to visualize the point cloud from the LIDAR. On digging deep into the problem, I found that the topics published

by the ZED and the LIDAR are not synchronized. To solve this, I had written a script with the help of Chien Chih Ho, using message filters, which I have not tested, it yet.

3) **No HDMI output from Jetson**

Getting an output from the HDMI port of the Jetson Tx2, was a challenge. After I spent a considerable amount of time, I got to know that, the HDMI port gets disabled during the bootup process. The solution was to unplug the HDMI cable from the port and let board bootup. Once this is done, the display output was seen on the monitor.

Team Work:

Past two weeks were intense, as we ourselves had a deadline for completing the perception pipeline, and making some progress on the getting the vehicle to move, with waypoints. We held multiple discussions, regarding our FVE test plan and also for getting updates on the areas which we had to really work in order to get the system fully integrated.

One problem, which we as a team faced was that, since everyone in our team had access to the common computer, they installed all the necessary files required for the development of the things they were doing. But this had a problem that the some of the installed libraries started conflicting with each other and thus was difficult for us to understand and debug it.

After discussing with the team, we came up with a solution in which we dual booted the system with 16.04 LTS. One OS would be used for perception pipeline and the other would be used for the prediction pipeline.

Vivek Gopal Ramasamy: LIDAR-Camera calibration and setting up the Jetson Tx2.

Chien Chih Ho: He was responsible for the single shot pedestrian detection on a live feed and also for helping me out in the calibration process.

Pengsheng Guo: He was responsible for optimizing the algorithm for prediction based on LSTM model and for training the model with different datasets.

Oliver Krengel: He was mainly responsible for the waypoint navigation for the vehicle and for developing the PCB along with me and Rohit.

Rohit Murthy: He was responsible for the schematic design and PCB. Apart from that, he was responsible along with Pengsheng, to implement algorithms for prediction using LSTMs in pytorch and also for understanding the basics of data processing.

Current Project Progress

In terms of the project, we have

1) Perception pipeline almost ready except for the LIDAR-CAMERA calibration.
2) Prediction pipeline is still in progress.
3) Advanced waypoint navigation on the Team D's car.
4) Designed PCB for the power distribution board.

Future Plans

1. To Optimize the algorithm based on social LSTM for better accuracy.
2. Write a ROS node to interface the tracking and the LSTM modules
3. Complete the sensor fusion, so that perception pipeline can be closed
4. Work on improving the localization of the car or set up a controlled environment for the car to localize correctly.

References

[1] https://github.com/CPFL/Autoware-Manuals

[2] https://arxiv.org/pdf/1705.09785.pdf

[3] https://github.com/ankitdhall/lidar_camera_calibration

[4] http://developer2.download.nvidia.com/embedded/L4T/r27_Release_v1.0/Docs/Jetson_X2_Developer_Kit_User_Guide.pdf?vcEds9XE5uyKqilEPzeWcWaoUzGPLKH3th2SeXAhCW6uCZsi3pf4ERBWCBB7ufwsM1fc9yIWEn8a0FUzdV-HPkOIC6mIxJS6sdM8QcWsNSthqpI_u8YZJBvQsrX93aeXaM-UZgVBOXKgD2BxabgbuPROxPJzhxhOHYSQ6JSWM48Ry5yfL0SYuQ