Report Date: 04/29/2022
To: ematson@purdue.edu, ahsmith@purdue.edu, lhiday@purdue.edu and lee3450@purdue.edu
From: FarmVroong

- Jueun Mun (cindy4741@khu.ac.kr)
- Seongil Heo (tjddlf101@hufs.ac.kr)
- Jiwoong Choi (jwtiger22@khu.ac.kr)
- Jiwon Park (overflow21@khu.ac.kr)

Summary

In this project, the tasks could be largely divided into hardware and software. First of all, the hardware were tested, such as a single board computer and sensors: ultrasonic sensor, GPS, 3D camera, and 2D Lidar. This week, it was decided which sensor to used in this project. Second, in the software part, there are main phases for driving the Autonomous vehicle: i) Simultaneous Localization and Mapping (SLAM), that builds a consistent map of the unknown environment and know the location of the vehicle; ii) path planning, that chooses the optimal trajectory; The main goal of this week was the theoretical understanding about SLAM.

What FarmVroong completed this week:

- Setting the system and sensor

  - Jetson TK1

    Mobile robots require wireless connections, therefore Robot Operating System (ROS) is essential for robot programming. Since it was unable to install ROS on Raspberry Pi 4 Model B, NVIDIA Jetson tk1 was used. After flashing Linux For Tegra 21.1.0 (L4T), it turned out that there were some limitations.
    - Ubuntu 16.04 cannot be installed on L4T 21.1.0 generally
    - After the force installation of Ubuntu 16.04, its GUI was broken
    - There was no Wi-Fi chipset onboard, it was required to put an external Wi-Fi dongle to use the wireless network.

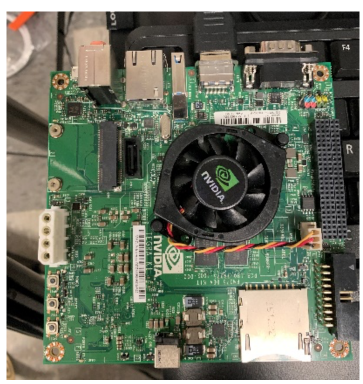


Fig. 1. NVIDIA Jetson TK1

Due to the limitations, Dr.Matson ordered NVIDIA Jetson Nano.

  - Sensor

Autonomous vehicle needs various sensors to perceive an unknown environment and to decide how to drive. Therefore, the sensors have been tested for checking the data with the laptop and the raspberry pi.

- Ultrasonic Sensor (HC-SR04)

  HC-SR04, the ultrasonic sensor offers the ability to detect the distance of 2cm to 400cm using the reflection of sound. This sensor has been checked for its ability to collect normal data that can be used for our projects in real-world environments. The electronic circuit was manufactured as shown in Fig 2. and experimented by using raspberry pi 4 B. The distance with obstacles could be calculated through the time interval between sending trigger signal and receiving echo signal. Therefore, the time was measured and divided by 2. According to our experiment, the sensor could detect obstacles within 200cm.

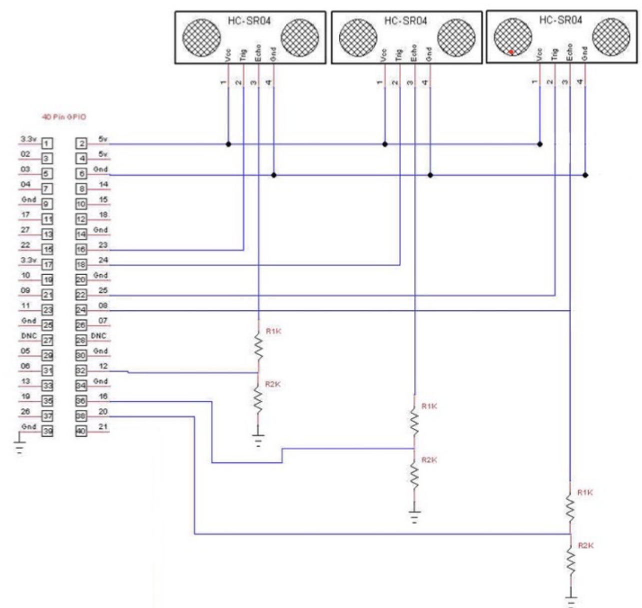


Fig. 2. The electronic circuit with three ultrasonic sensor and raspberry pi GPIO pin. Raspberry pi GPIO is shown on the left and ultrasonic sensors is shown on the top of this figure.

- GPS

  BU-353-S4 is a USB GPS receiver that features a highly sensitive, low power consumption chipset. [7]

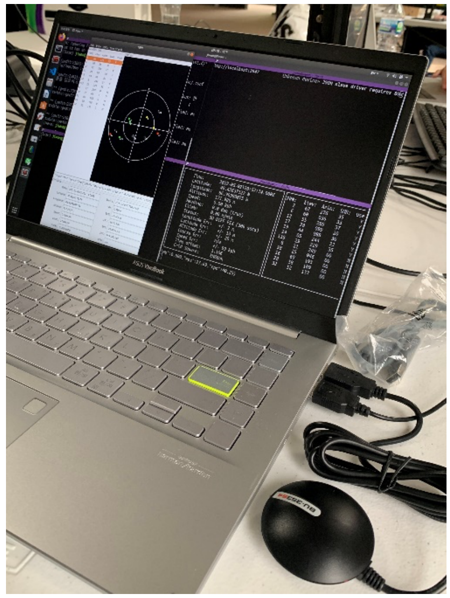
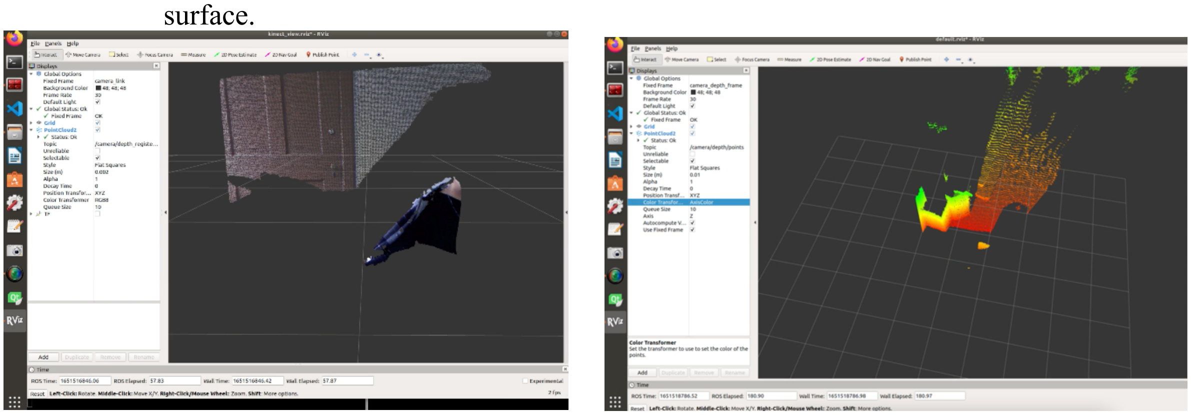


Fig. 3. Testing GPS on Ubuntu 18.04

gpsd is the GPS service daemon for Linux via serial, USB port, and Bluetooth to use the GPS information. Compared to the actual GPS data from Google Map, this GPS sensor showed accurate GPS location of the K-Square building.

- Kinect XBOX 360

Kinect is a RGB-D (Depth) camera produced by Microsoft. It uses infrared projectors and detectors to create depth map through time-of-flight calculations. At first, Kinect was regarded as useful in that Kinect can create a depth map of a scene like camera, in contrary to 2D LiDAR which can only detect one depth map in a planar surface.

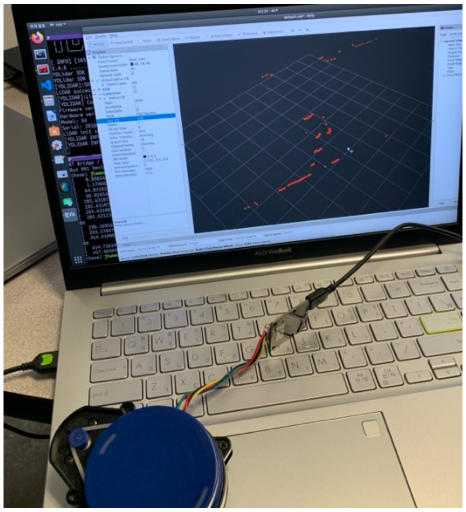


(a) (b)

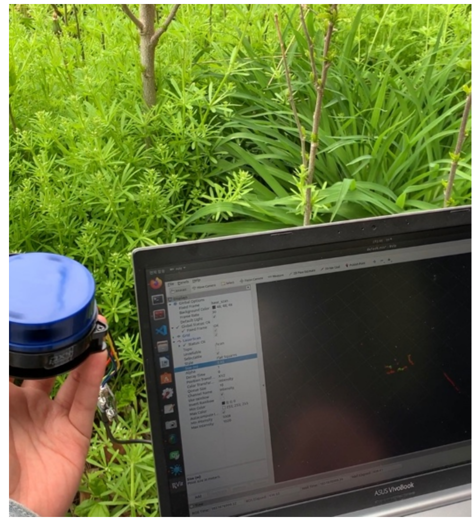
Fig. 4. (a) RGBD image accuired from Kinext Xbox 360. (b) Point Cloud Data visualized on ROS-Visualiztion Tool (Rviz)

- YDLiDAR X4

YDLiDAR X4 is a 2D LiDAR which has 10 m , 360 degree of ranging radius. Data achieved by YDLiDARwas visualized by Rviz.



(a)



(b)

Fig 5. (a) Testing YDLiDAR in indoor environment. (b) Testing YDLiDAR in outdoor environment, especially in the middle of the bush.

The data showed that the YDLiDAR is robust when dealing with flat surface like wall, however, it was difficult to recognize surrounding environment like bush or grass. Therefore, the research for using LiDAR in more unrefined environment has to be conducted.

- Researching the SLAM

SLAM is the process by which a mobile robot can build a map of an environment and at the same time use this map to deduce its location. SLAM consists of two parts, localization and mapping.

- Localization

First of all, Localization is the process which decides the position and direction of car based on the map. There are two ways for it, Global localization and Local localization. Global localization uses external information to decide the present location. For example, there are Global Navigation Satellite Systems (GNSS) and previously known landmark, detected by LiDAR. Local localization is comparing present posture (or position) with past posture (or position), such as wheel odometry, Inertial Navigation System (INS), and LiDAR.

The most popular localization ways, which are adequate to outdoor SLAM, are LiDAR based localization and Camera based localization.

- LiDAR-based localization

LiDAR-based localization mostly uses scan matching. Scan matching is the technique which searches geometric arrange, which two results of scan can be overlapped optimally. This can be applied to detect the loop closing, which means the car revisits the past location. Loop closing results in the correction of odometry.

The most cited technique of scan matching is Iterative Closest Point (ICP). It minimizes the point-to-point distance. There are similar ways with ICP, which use point-to-line, and feature-to-feature, as well.

Each scan matching technique is effective in different situations. Therefore, it is necessary to compare which one is most effective in the farm. For example, feature-to-feature scan matching is only effective in the environment which has many landmarks.

◆ Camera-based localization

Camera-based localization finds the position and direction of the car not only from the start point but also from the global map. There are two ways of camera-based localization, feature-based method and direct method. Feature-based method extracts the essential features from the present observation, such as corner and edge, and compares them to the known features in advance. Meanwhile, direct method uses the information of the observation directly, which is appropriate to the environment which has low texture.

ORB-SLAM [2] is an example of feature-based method. ORB SLAM advanced from a parallel tracking and mapping (PTAM) [3] which is a simple and effective real-time method, however, it limits to small scale operation and this occurs some limitations. PTAM has a lack of loop closing, adequate handling of occlusion, and low invariance to viewpoint of the relocalization.

ORB SLAM, based on the main idea of PTAM, builds scale-aware loop closing and uses the covisibility graph for the large-scale operation. ORB SLAM provides bundle adjustment (BA) with the following.

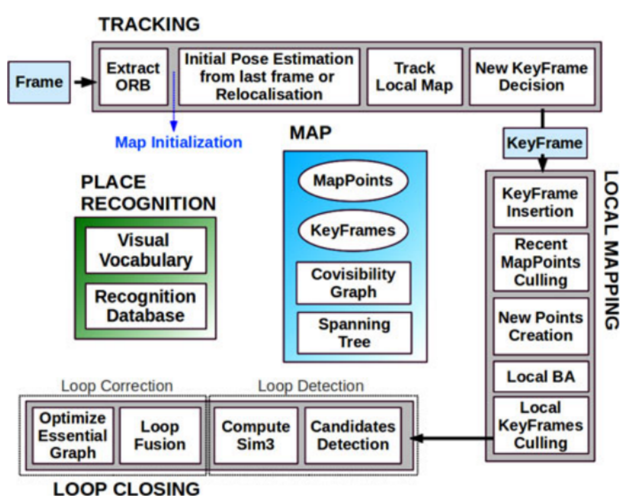


Fig. 6. ORB SLAM system overview. There are three part in the algorithm, tracking, local mapping, and loop closing.

1. Using the same features for the tracking, mapping, relocalization, and loop closing
2. Using covisibility graph [4] for real-time large-scale operation
3. Real-time loop closing based on the optimization of a pose graph
4. Real-time camera localization with significant invariance to viewpoint and illumination
5. A model selection for the planar or nonplanar scenes
6. A survival of the fittest approach to map point and keyframe selection

■ Mapping

There are two prime mapping algorithms in SLAM, filtering-based approach and optimization-based approach. Before understanding two algorithms, it is necessary to know the types of map.
In addition, SLAM is divided to two problems, online SLAM and full SLAM. Online SLAM only estimate recent posture and map. However, full SLAM estimate the global trajectory and map.

◆ Types of map in SLAM

Occupancy gird map and relation map are prime example in SLAM. Occupancy grid map is the most popular map in SLAM. It decomposes the environment into square cell and each cell includes the probability of occupancy.

Meanwhile, relation map defines the relation between the elements of the environment. Concretely, pose-constraint map is well used in the graph-based SLAM. It includes the posture of the car (position and azimuth), and gradually express the map by graph. Each element (or node) is connected by edge, which is usually based on the odometry measurement value. Edge means the constraint between poses.

◆ Optimization-based Algorithm

Optimization-based Algorithm works using smoothing principle. It uses whole set of posture and measurement value from start and finds the set of trajectory which has the high probability of matching with whole set. Therefore, it can be a good solution to the full SLAM, that it is the state-of-the-art in SLAM

Optimization-based Algorithm consists of front-end and back-end. At the front-end, the relevant features are extracted from the sensor data, and data-association is conducted. In addition, data-association is divided to local (short-term) one and global (long-term) one. Local data-association traces the features between two continuous measurement value. Odometry correction is the representative example of local data-association. Global data-association searches whether the car revisits the location that it previously visited, which is loop closing. Likewise, Back-end part conducts the process of finding the optimal constitution which maximizes the consistency of all observation. Maximum A Posteriori (MAP) is the prime problem of it

There are two main types of optimization-based algorithm. One is graph-based algorithm and bundle adjustment (BA).

● Graph-based algorithm

Graph-based algorithm realizes the pose-constraint map. Each node of the graph means the pose and location of the car, and the edge which connects nodes means the space constraint (movement and rotation). The construction of the graph and data-association is conducted at the front-end, and maximum likelihood estimation is conducted at the back-end. Maximum likelihood estimation is solved by gradient descent method (GD), Gauss-Newton, or Levenberg-Marguardt.

In case of Camera-localization, each node, which is named key-frame, represents present state. The interval of key-frame has a trade-off between performance and accuracy. If the number of key-frames increases, operation time gets longer.

- Bundle adjustment

Bundle adjustment is the simultaneous refining technique which joint optimize the 3D coordinates and parameters. It minimizes the difference between the position of the observed landmark and each predicted position of landmark. The value function is the sum of error of landmark-to-pose and error of pose-to-pose. Therefore, if the landmark-to-pose error is ignored, bundle adjustment is reduced to graph-based algorithm.

- Depth estimation

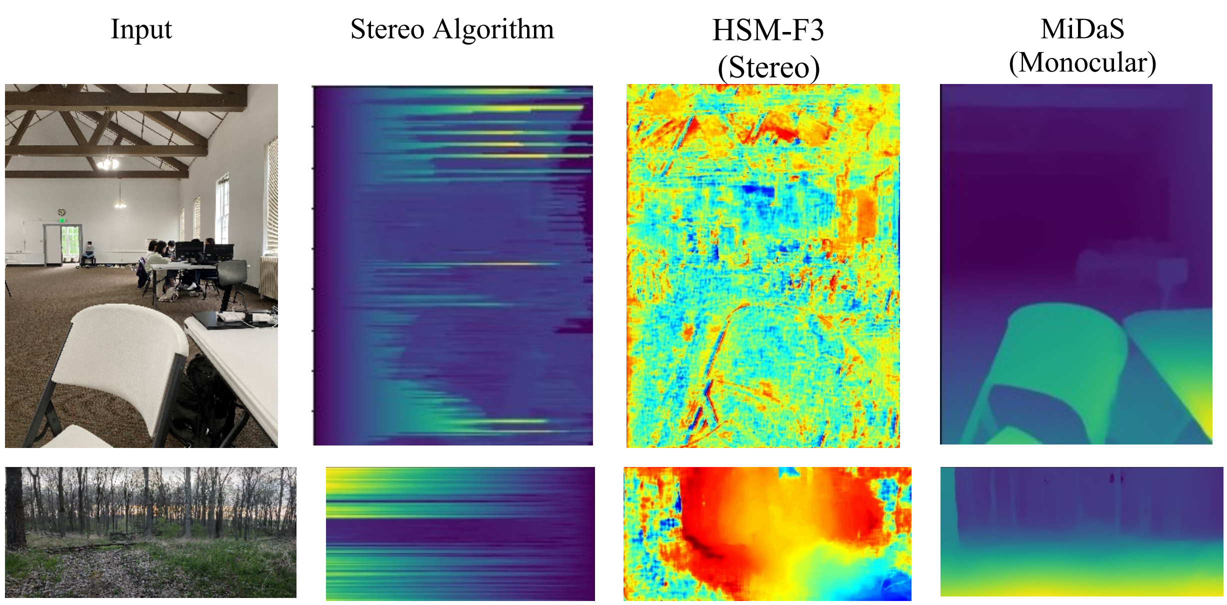


Fig. 7. Estimate the depth prediction with the various methods.


Fig. 8. Depth prediction with ADDS-DpehtNet.

When mapping the local area, obstacle avoidance is an important task. Using edge detection for detecting the obstacle had a bad result. For this reason, we changed the definition of the obstacle which is placed in front of the vehicle. At first, using the stereo depth algorithm for depth estimation had a bad result. Using the algorithm may have a computational burden and poor time complexity. Accordingly, we try to use the model-based depth estimation [4], [5], [6]. The results of using monocular and stereo depth estimation models show in Figure 7, 8. MiDas [4] had the best result, however, the accuracy of the depth is still not enough for avoiding the obstacle.

Things to do by next week

- Find the way to use both camera and Lidar simultaneously in SLAM.
- Give different importance to road and off-road in path planning.
- Decide which algorithm to use for SLAM out of filtering (Kalman filter, particle filter) or optimization (graph, bundle adjustment)
- Filtering-based algorithm while mapping

Problems or challenges:

- The performance of the 2D LiDAR is not good enough. Therefore, the algorithms that can maximize the utility of the data acquired from 2D LiDAR has to be contrived.
- The common algorithm in SLAM is the filtering algorithm such as Extended Kalman Filter (EKF), however, this algorithm doesn't suit for the environment that we are dealing with. Therefore, the research for another algorithms like Graph-based SLAM has to be conducted.

References

[1] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99–110, Jun. 2006.

[2] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: A Versatile and Accurate Monocular SLAM System," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, Oct. 2015.

[3] Y. Kameda, "Parallel Tracking and Mapping for Small AR Workspaces (PTAM) Augmented Reality," *The Journal of The Institute of Image Information and Television Engineers*, vol. 66, no. 1, pp. 45–51, 2012.

[4] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards Robust Monocular Depth Estimation: Mixing Datasets for Zero-Shot Cross-Dataset Transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 3, pp. 1623–1637, Mar. 2022.

[5] G. Yang, J. Manela, M. Happold, and D. Ramanan, "Hierarchical deep stereo matching on high-resolution images," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[6] A. C. S. Kumar, S. M. Bhandarkar and M. Prasad, "DepthNet: A Recurrent Neural Network Architecture for Monocular Depth Prediction," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018, pp. 396-3968

[7] “BU-353-S4 - USGlobalSat BU-353-S4 USB GPS Receiver” *navstore.com* . https://navstore.com/usglobalsat-bu-353-s4-usb-gps-receiver.html (accessed May 06, 2022)

[8] “Gpsd - OpenStreetMap Wiki,” wiki.openstreetmap.org. https://wiki.openstreetmap.org/wiki/Gpsd