HPC for Vehicle Autonomy Final Project

Realtime mapping, localization, path following, and object detection using **LGSVL**, **Autoware** and **LogiTech G29**

Team Number: 9

Zhi Fang Tan **Lingyi Song** **Kexuan Zhai**

Basic Goals and Achievements

- **Implementation of a virtual environment for autonomous drive testing in LGSVL**
- **Implementation of the communication between LogiTech G29 and LGSVL**
- **Communication between LGSVL and Autoware**
- **Point cloud mapping for a virtual map in LGSVL**
- **Automatic path following simulation**
- **Real-time object detection using yolo3 in virtual environment**

Team Roles

Lingyi Song

1. Real-time object detection using yolo3.
2. Software debugging.
3. Automatic path following.
4. Communication between Autoware & LGSVL
5. Organization.

Zhi Fang Tan

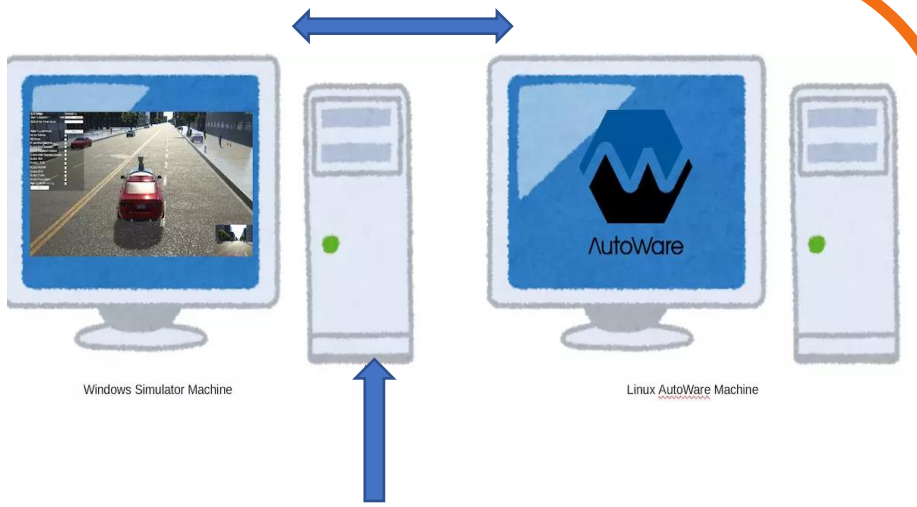
1. Communication between Autoware & LGSVL.
2. Path building.
3. Automatic path following.
4. Software debugging.
5. Simulation Implementation.

Kexuan Zhai

1. Software environment building and hardware implementation.
2. Point cloud mapping.
3. Software and computer drivers debugging.
4. Information and reference collecting.
5. Tests implementation and recording, and report writing.

Experiment Equipment

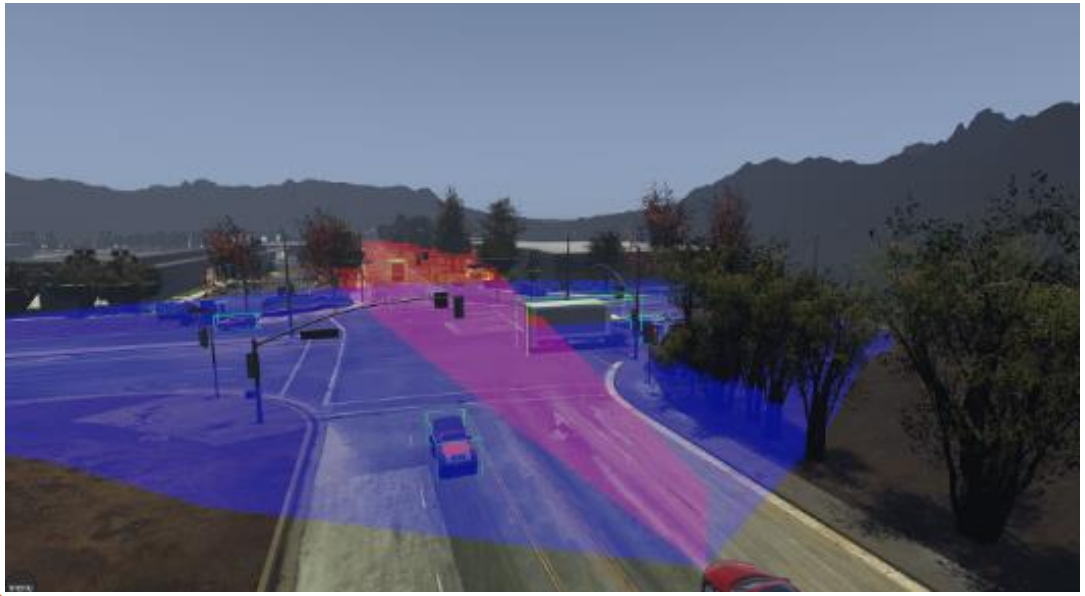
- **Software: LGSVL, Autoware**
- **Operating system: Windows and Linux**
- **Packages: CUDA10, cuDNN, CUDA toolkit**
- **Hardware: Logitech G29**



LGSVL Simulator

- A software for building virtual environment

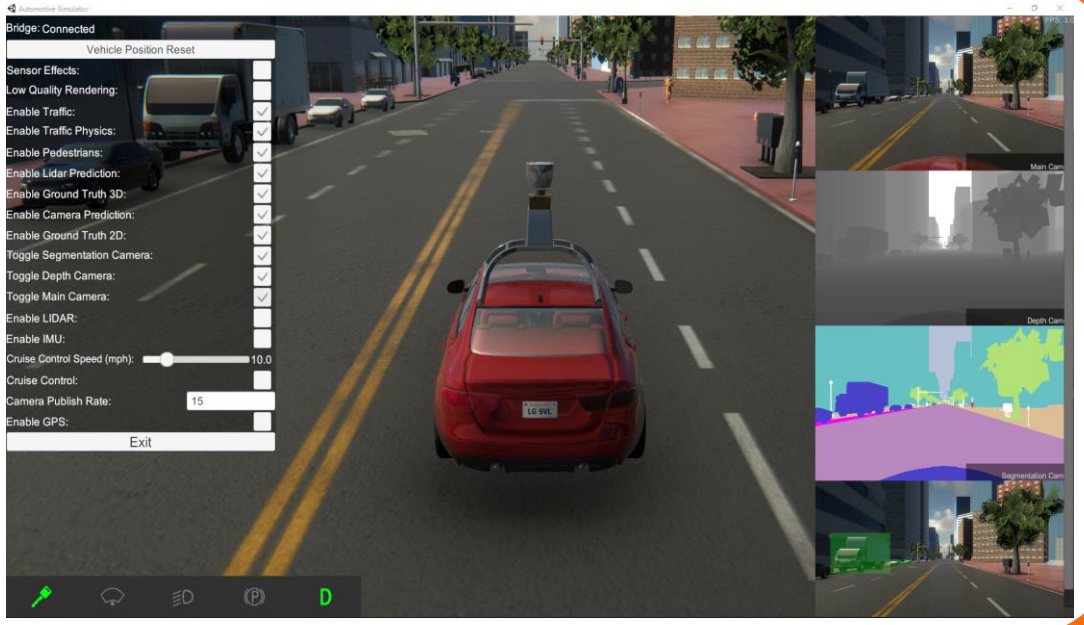
LG Electronics America R&D Center has developed an HDRP Unity-based multi-robot simulator for autonomous vehicle developers. It provides an out-of-the-box solution which can meet the needs of developers wishing to focus on testing their autonomous vehicle algorithms. It currently has integration with Autoware and Baidu's Apollo 5.0 and Apollo 3.0 platforms, can generate HD maps, and can be immediately used for testing and validation of a whole system with little need for custom integrations. They aim to build a collaborative community among robotics and autonomous vehicle developers by open sourcing our efforts.



The capability of LGSVL

- LGSVL is a powerful and practical simulation software

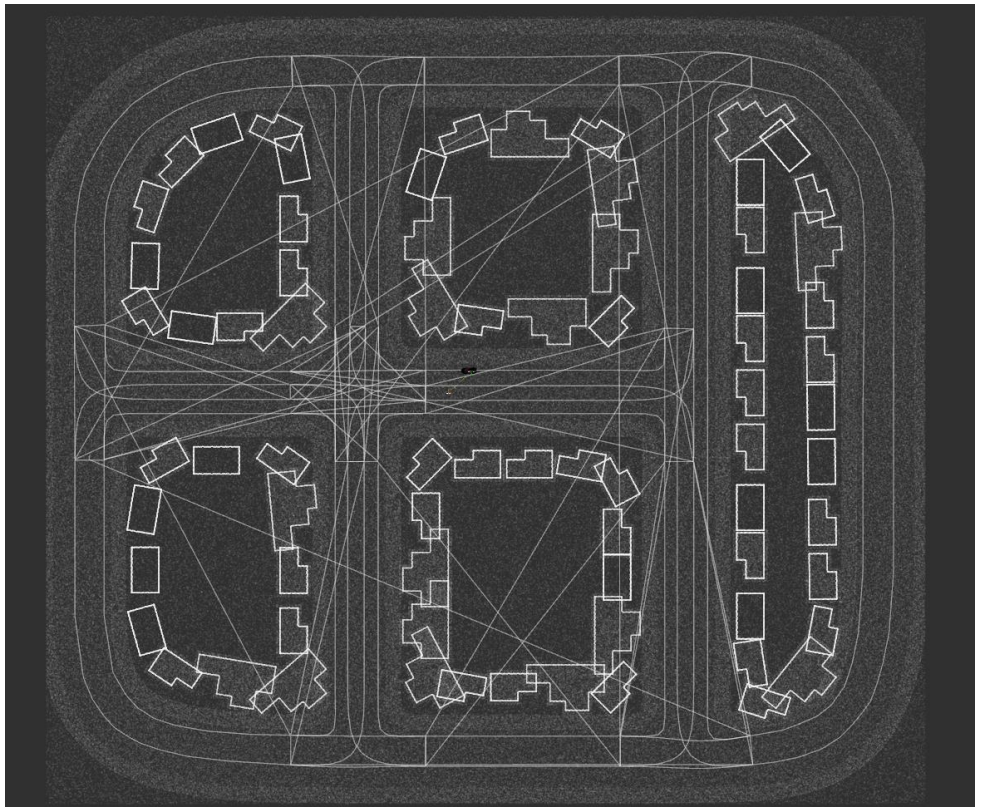
LGSVL can run a vehicle model in different maps and under different conditions. The vehicle models can output virtual data from lidar, radar, different cameras, etc. It also has a vehicle dynamics system. We can do various simulation and tests with it.

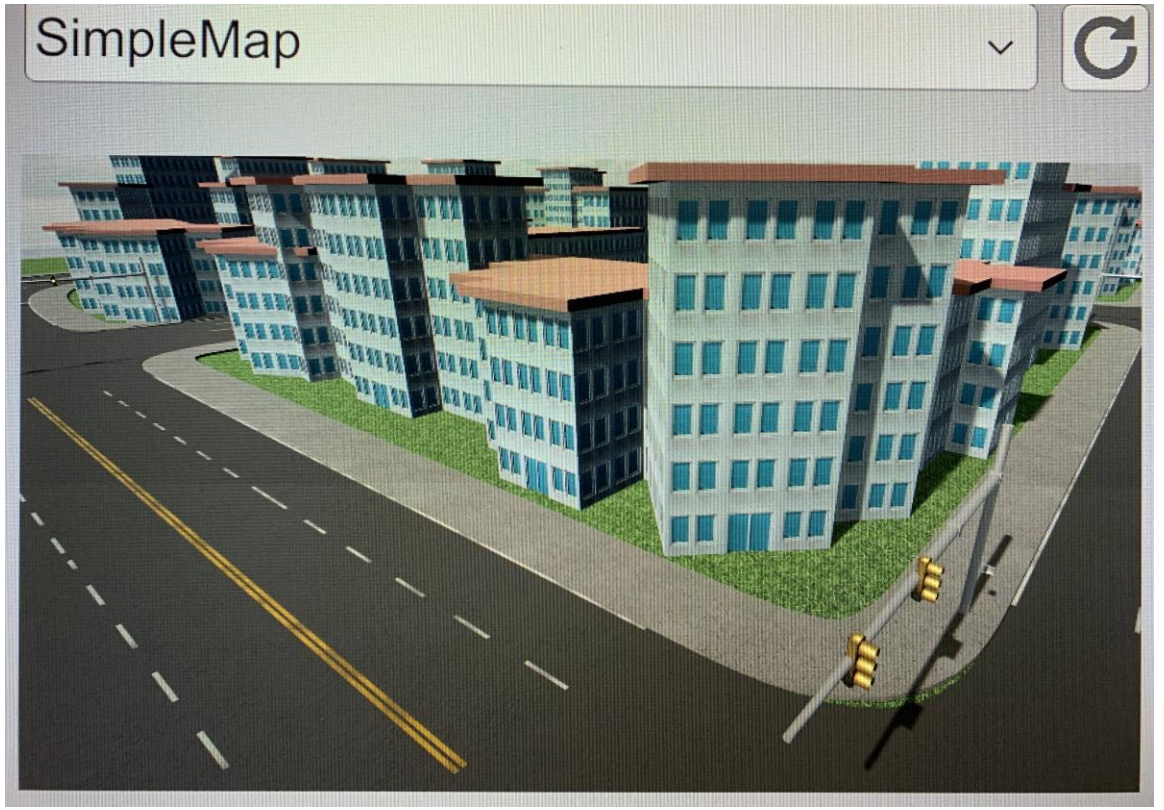


Virtual environment

- Simplemap in LGSVL

All the tests and demos are running under Simplemap.

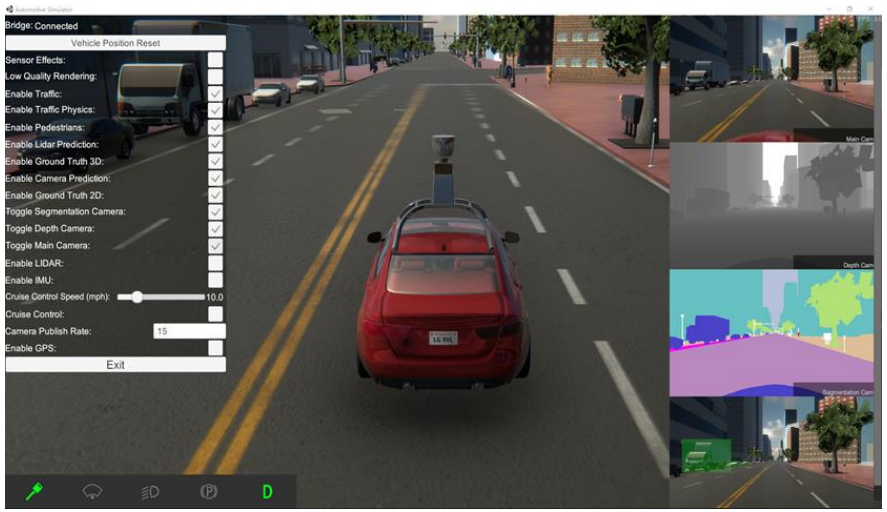




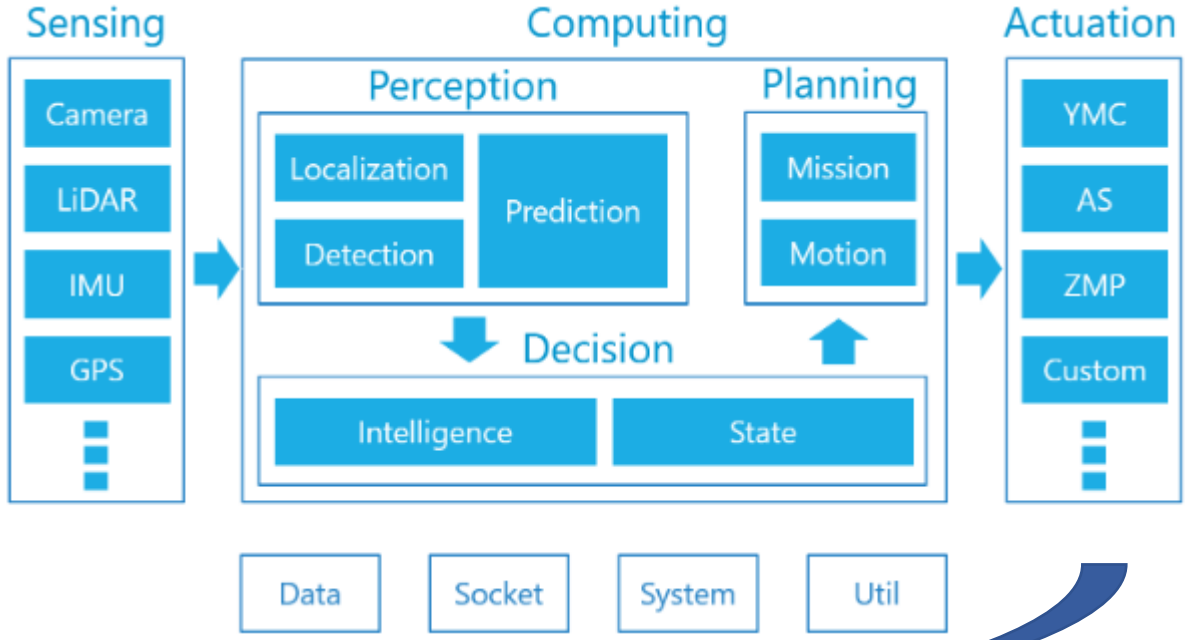
Platform configuration

- Autoware + LGSVL

Computing, data analyzing, and vehicle controlling are implemented in Autoware.



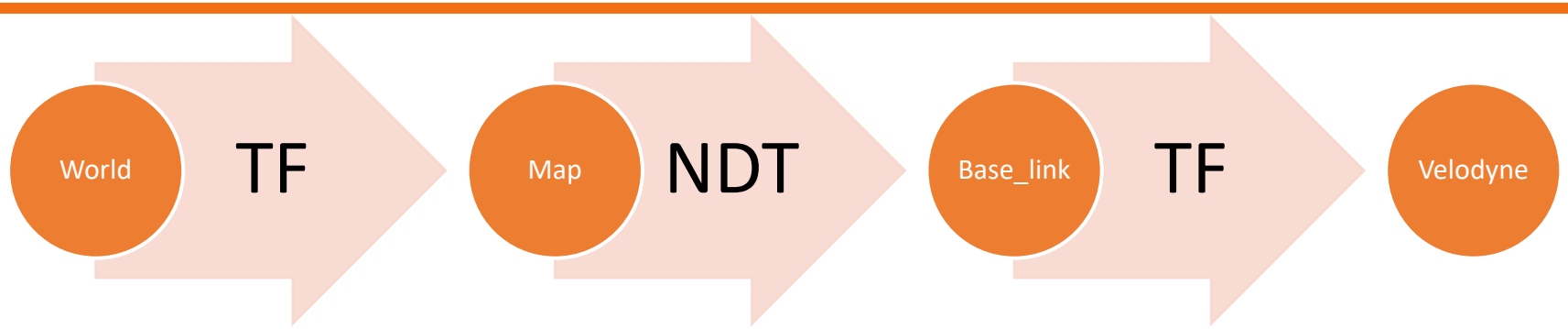
Sensor data

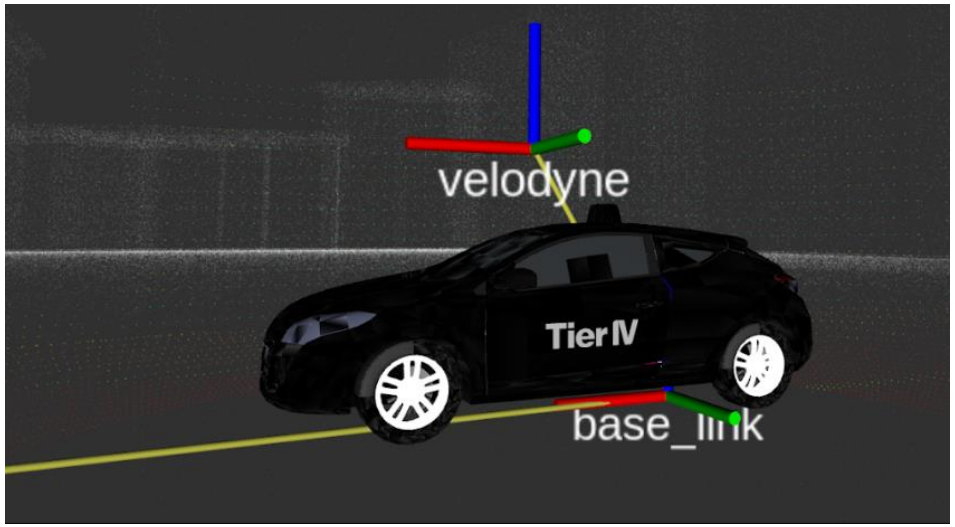


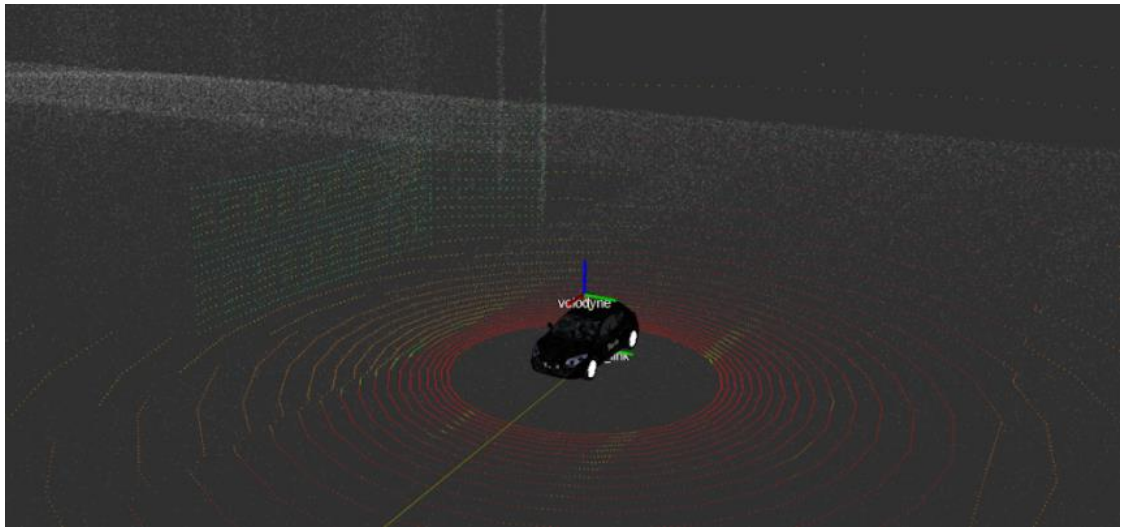
1. Map building

- Coordinate system in Autoware

Four coordinate systems: World, Mapping, Base_link, and Velodyne.







1. Map building

- Point cloud mapping using NDT Mapping

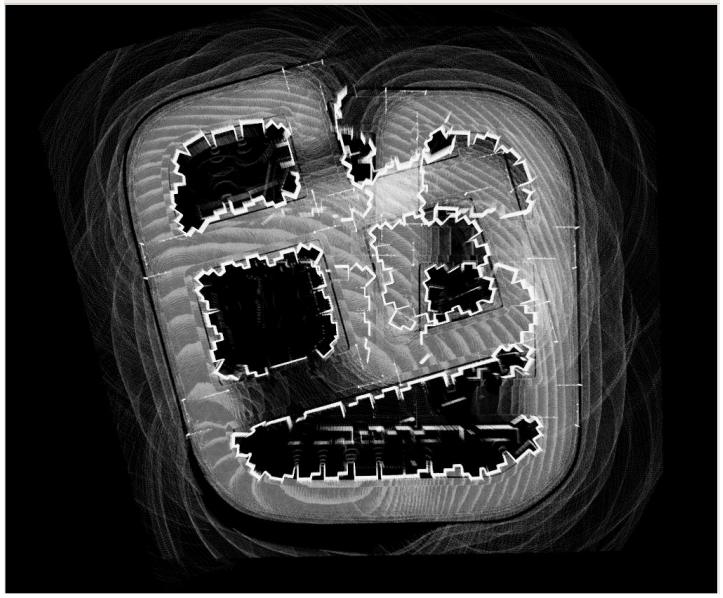
Built a point cloud map through lidar data acquired from a virtual map.

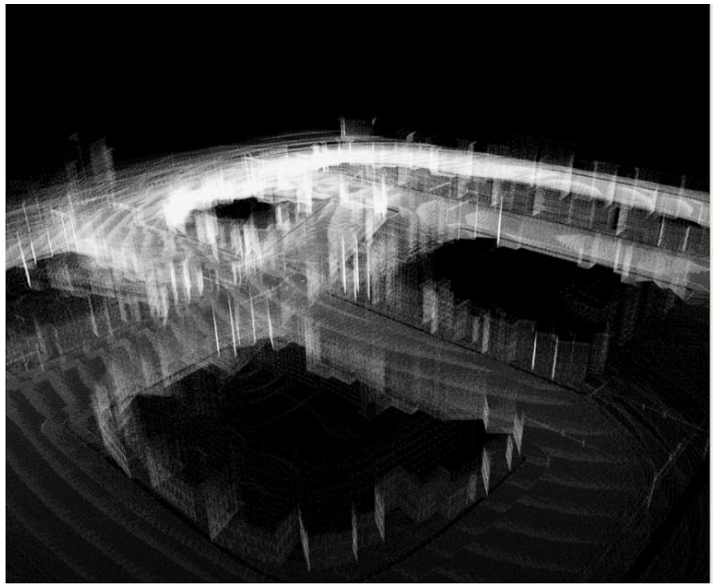
1. Map building

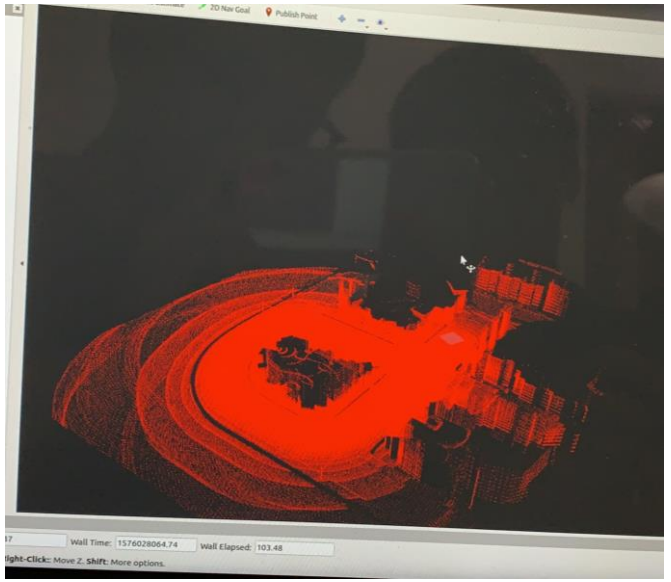
- Possible improvements

The map built needs to be filtered and optimized.

- No loop closure.
- No filtering.
- No optimization.



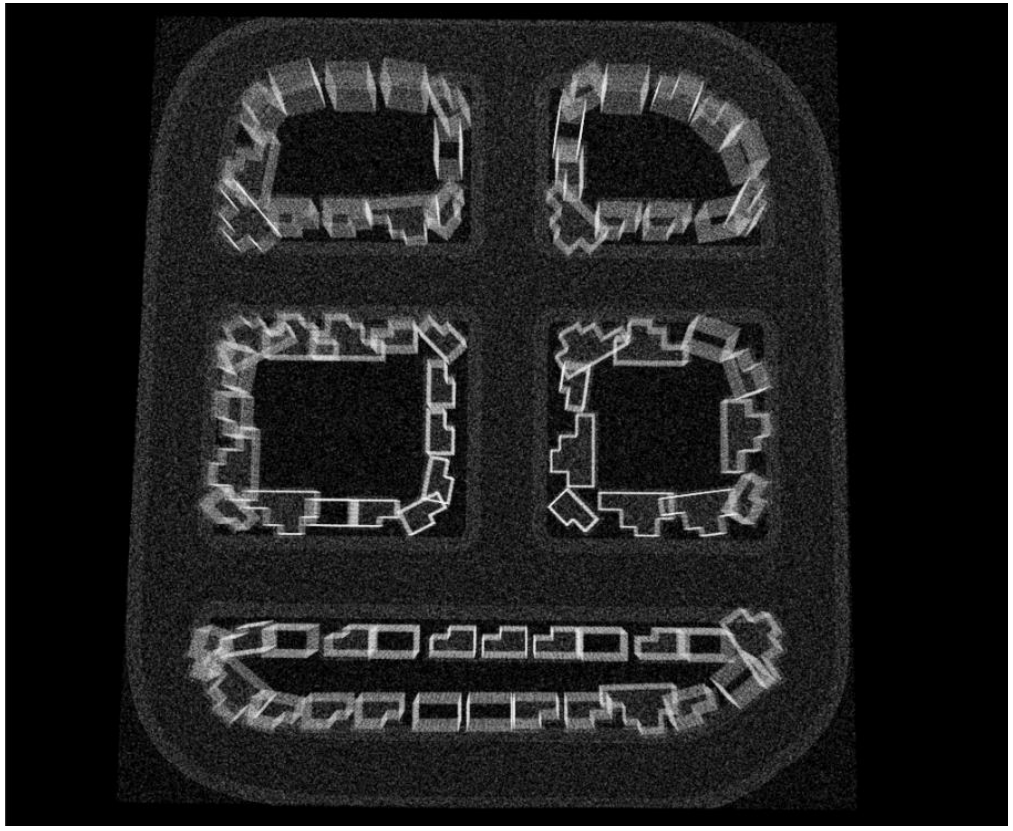


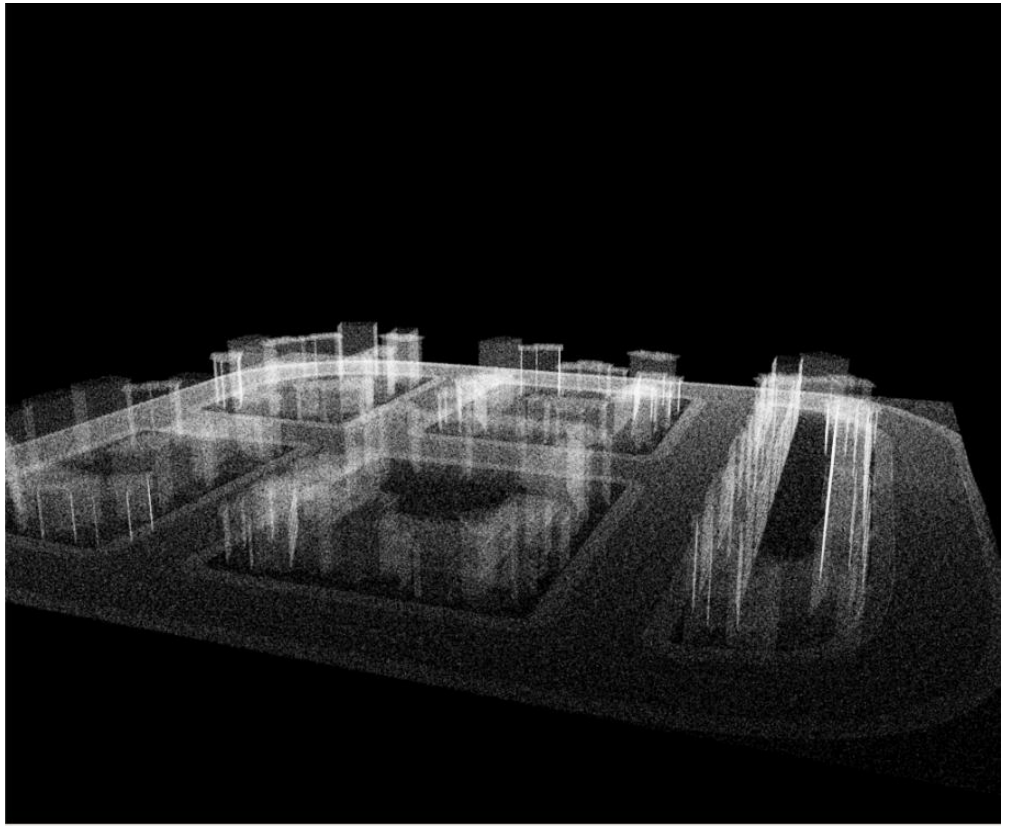


1. Map building

- Final map for Simplemap

For better implementation, point cloud map is taken from online resource.

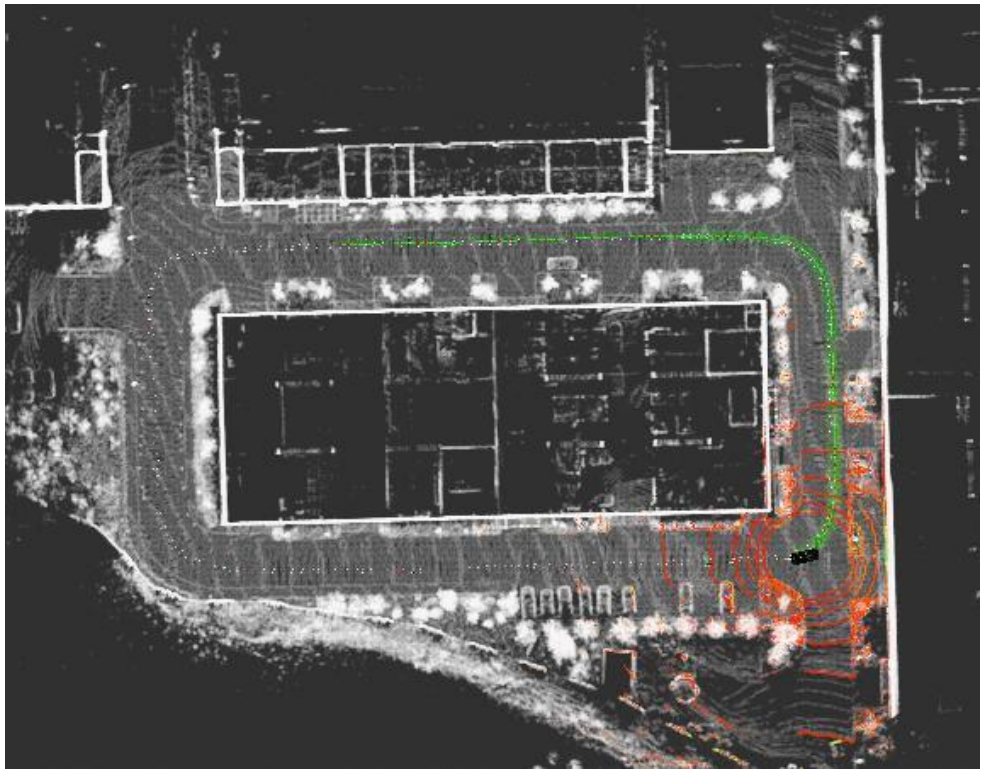


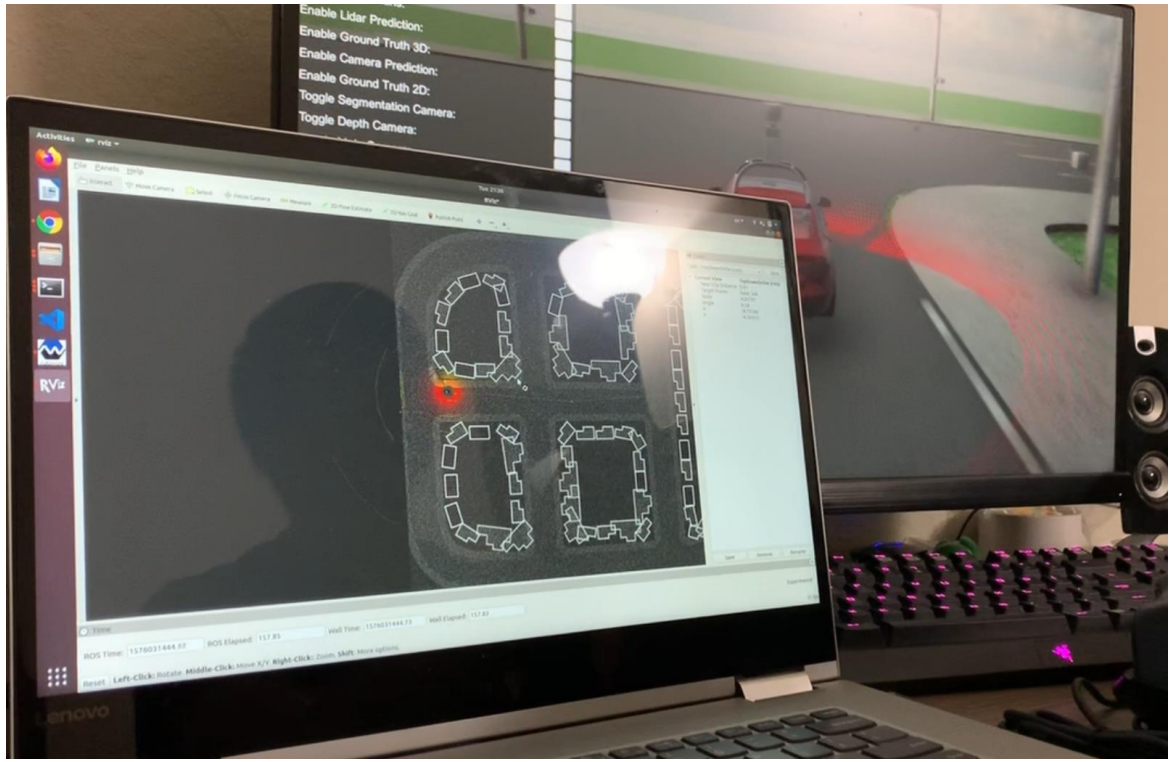


2. Localization and automatic path following

- Localization using NDT Matching

Localization in the points-cloud map though lidar.

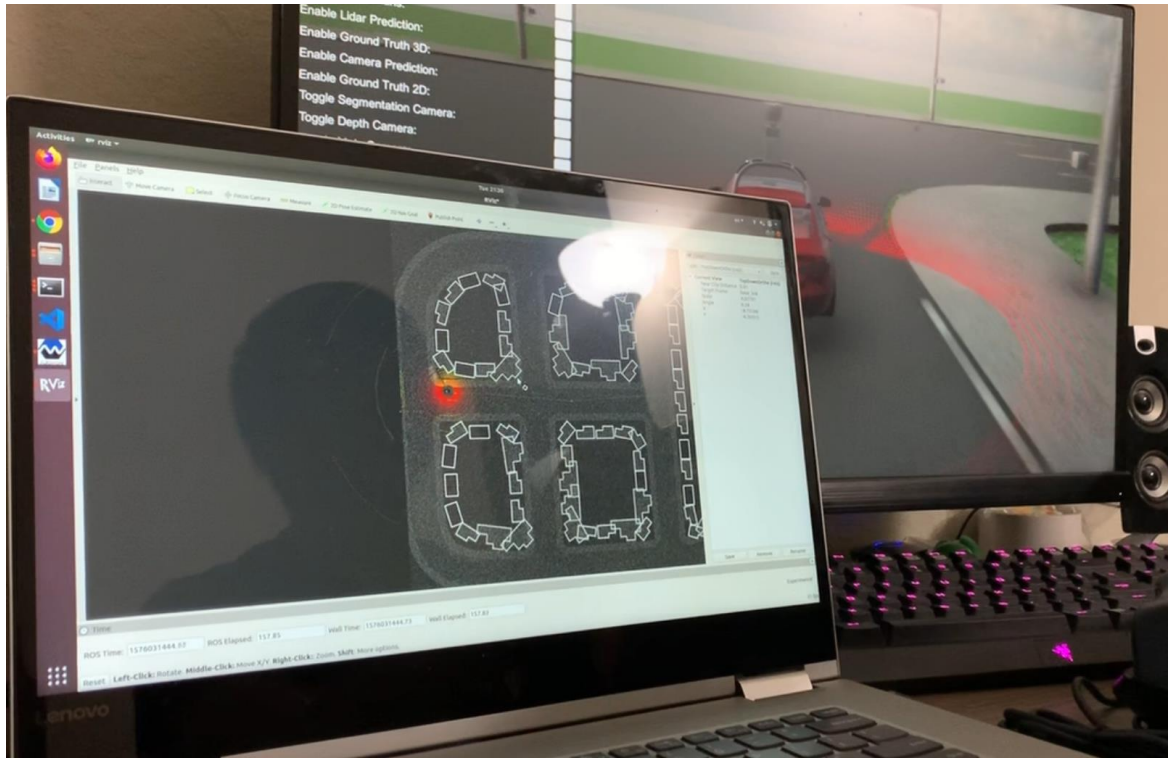




2. Localization and automatic path following

- Localization in point cloud map.

Demo.

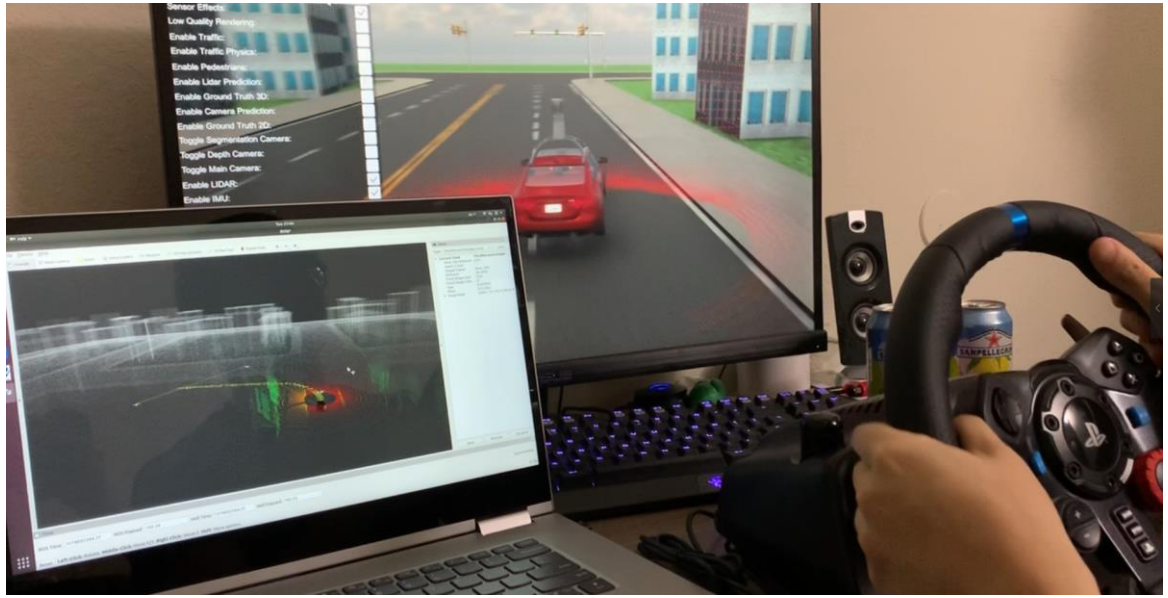


2. Localization and automatic path following

Pure Pursuit path following

The vehicle will automatically follow the path we recorded before.

- The path is recorded as .cvs file, with the x, y, z coordinates, the yaw rate and the speed of the car.
- The limitation of lateral acceleration can be set as we want.
- We can add GPS as the supplement for localization.



1. Set coordinate systems transformation.
2. Load the map.
3. Activate localization.
4. Set filtering and limitations.
5. Run the simulation data.
6. Monitoring the driving process.

2. Localization and automatic path following

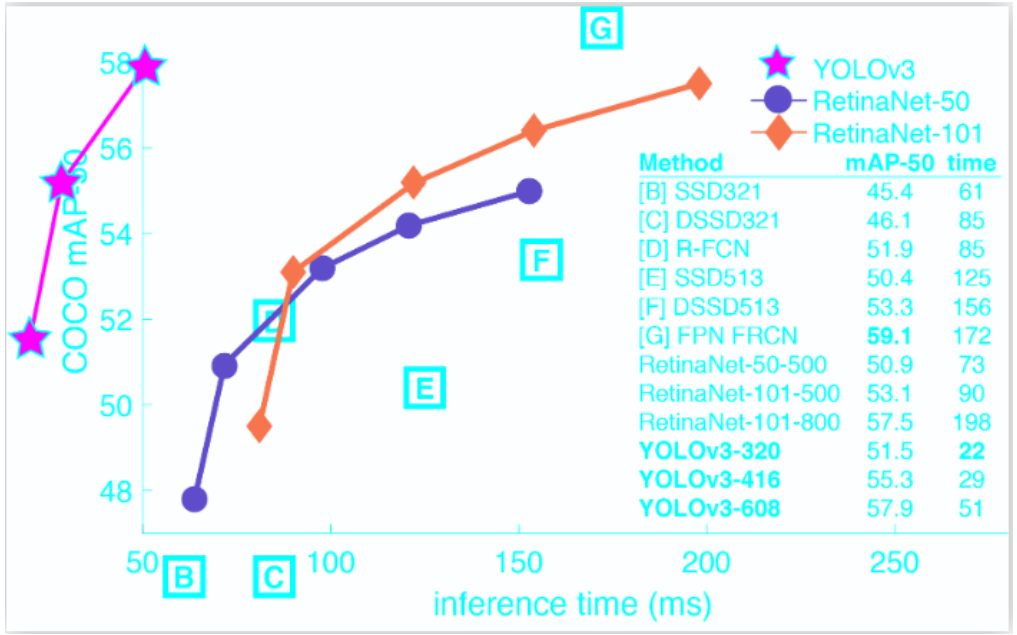
- Pure Pursuit path following

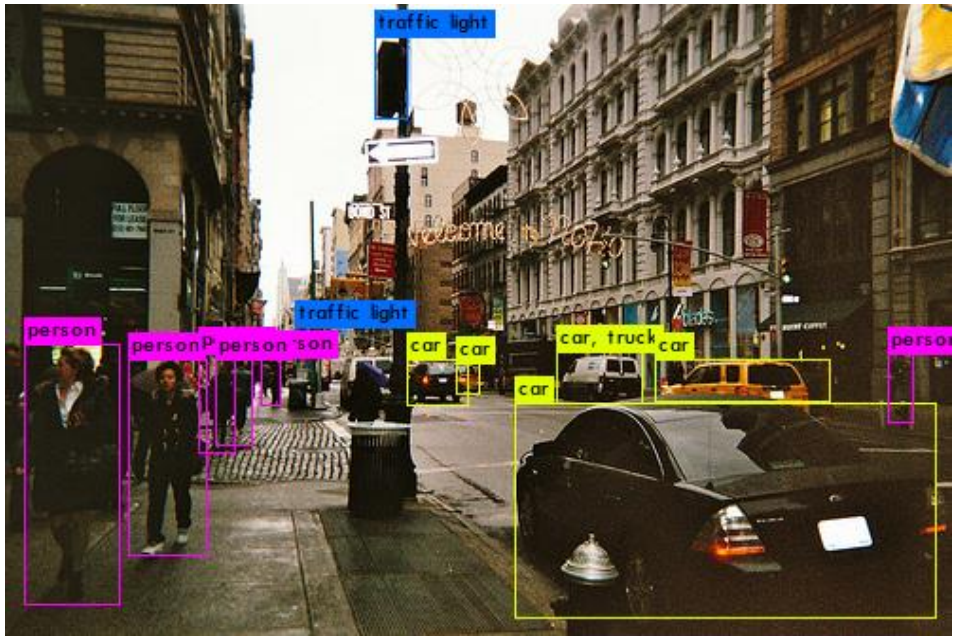
Demo.

3. Real-time object detection

- Yolo3

Yolo is a deep learning detection algorithm based on CNN.

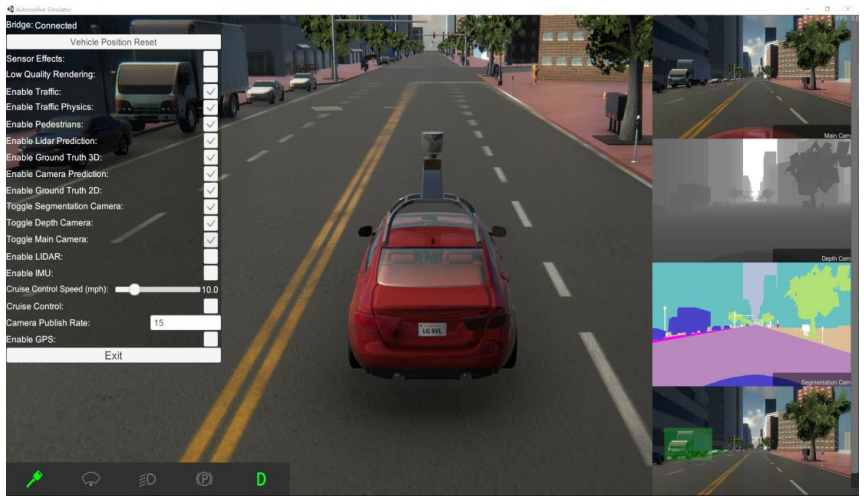


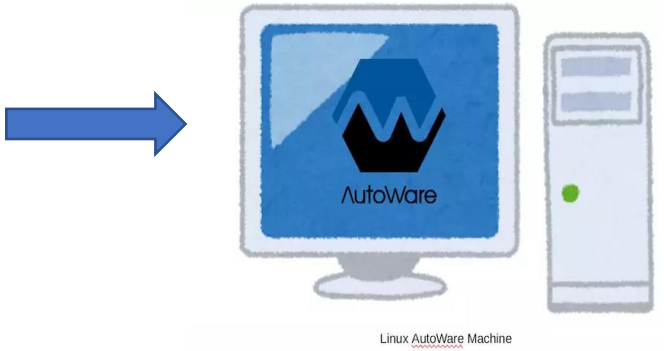


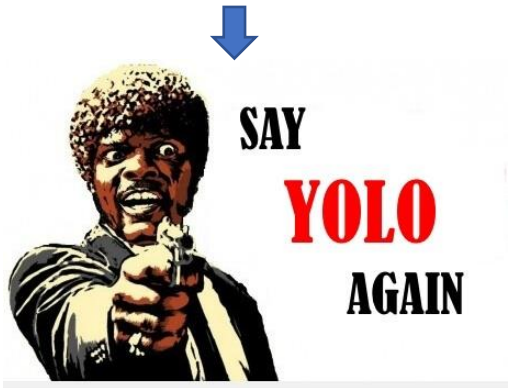
3. Real-time object detection

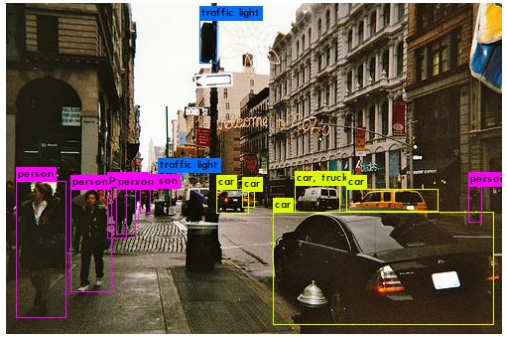
- Running yolo3 in our platform.

Running real-time object detection using the RGB camera in LGSVL.





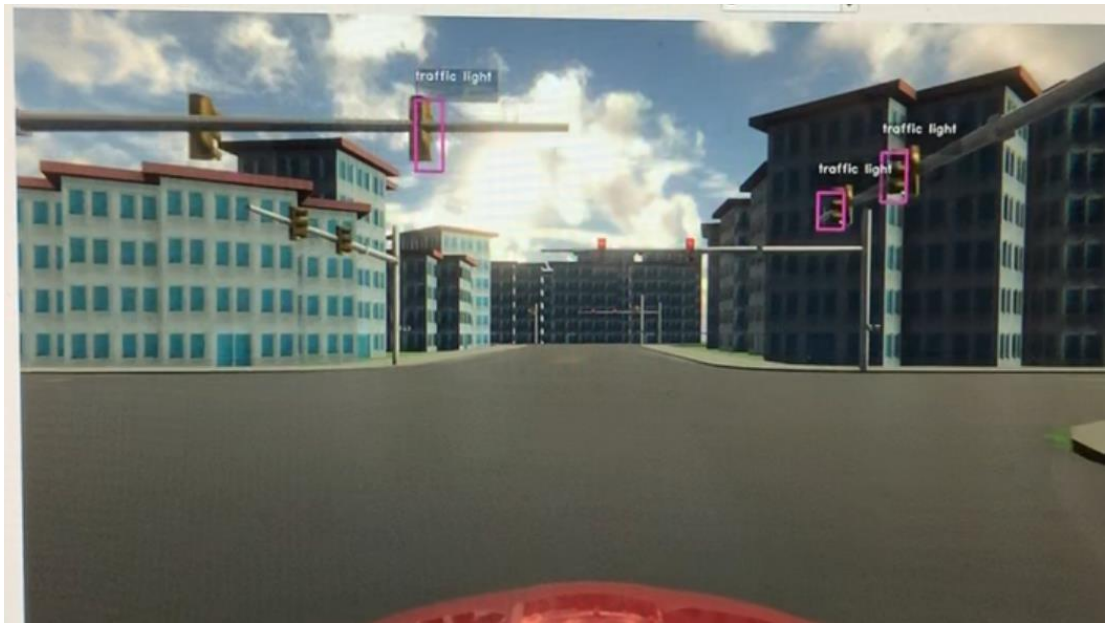


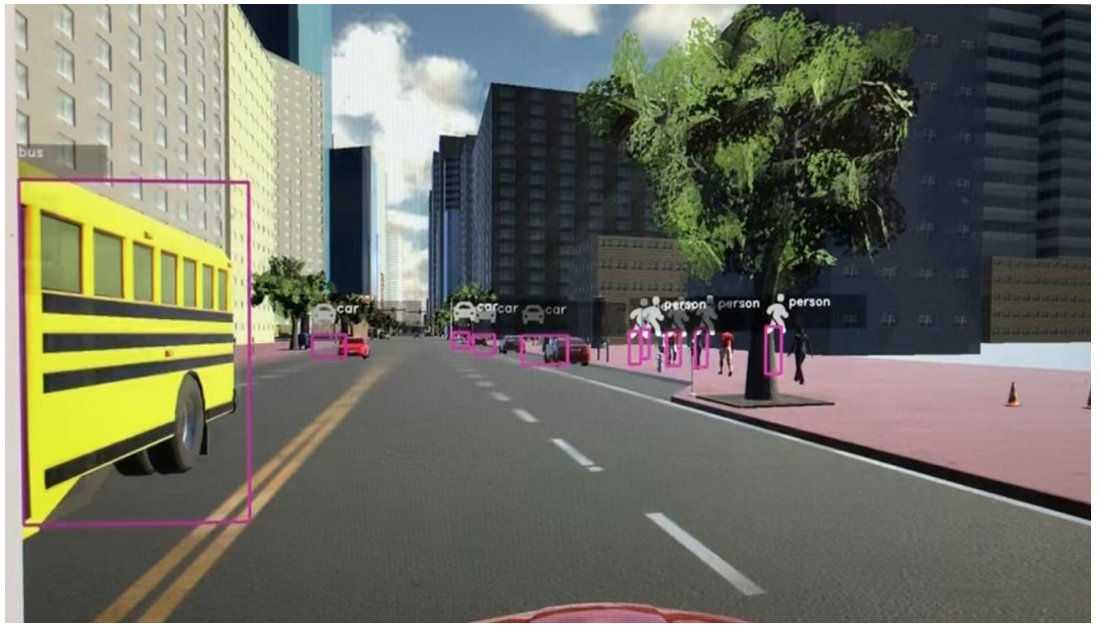


3. Real-time object detection

- Real-time object detection and recognization.

Demo.





Outcome and Conclusion

- Outcomes
  1. Point cloud mapping.
  2. Point cloud localization.
  3. Path following using pure pursuit.
  4. Real-time object detection.
- Conclusions
  1. The potential of LGSVL and Autoware.
  2. All these work can be transferred in real vehicles and real situations.
  3. Large amount of data and implementation of HPC in process.

3. Capability of our platform

- Capability of LGSVL and the platform.

Testing different vehicle model.

Hyundai Nexo 2018

Jaguar XE 2015

Lexus RX 2016

Lincoln MKZ 2017

3. Capability of our platform

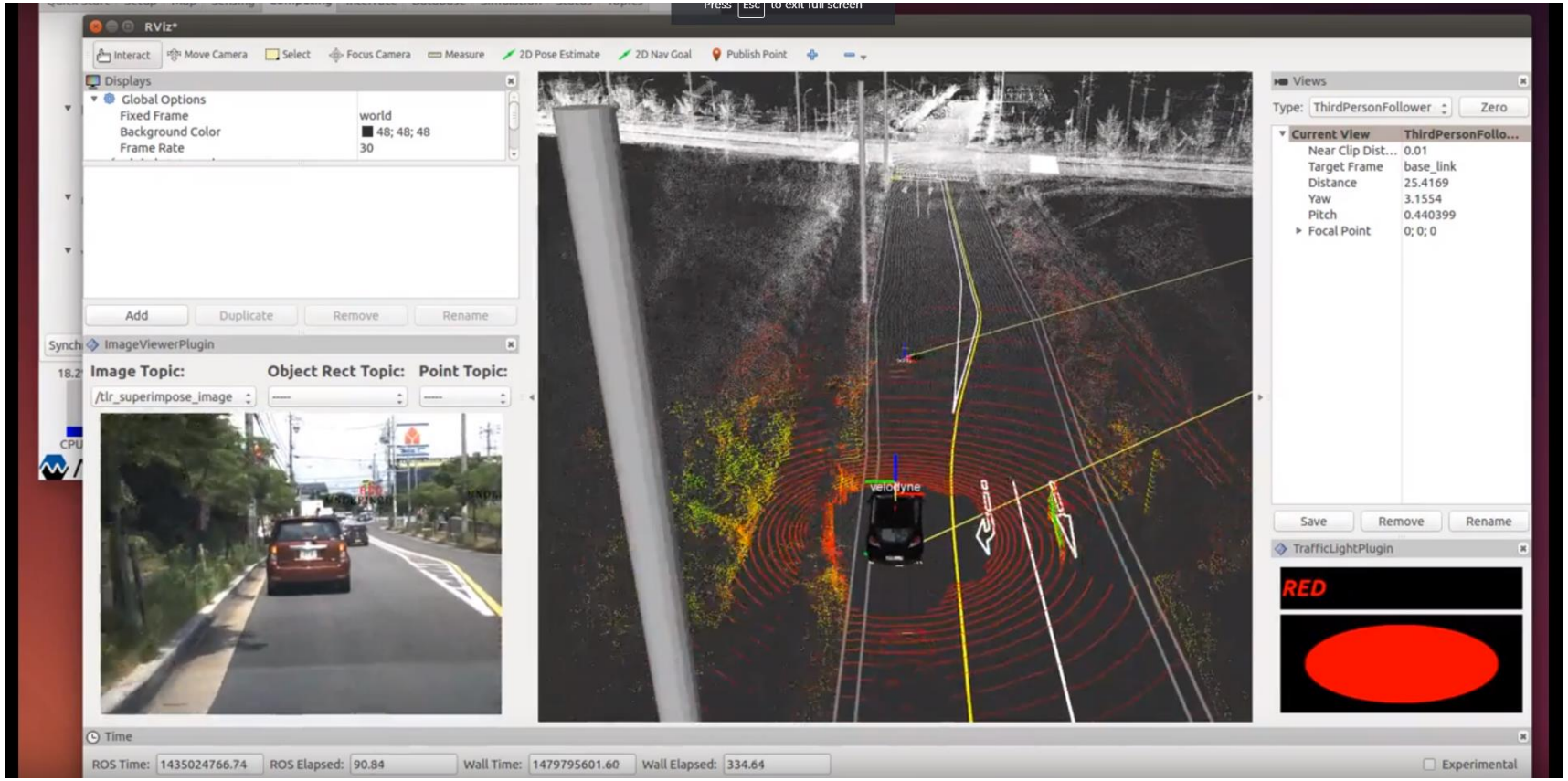
- Capability of LGSVL and the platform.

Building new maps and testing environment.

3. Capability of our platform

- Capability of Autoware and the platform.

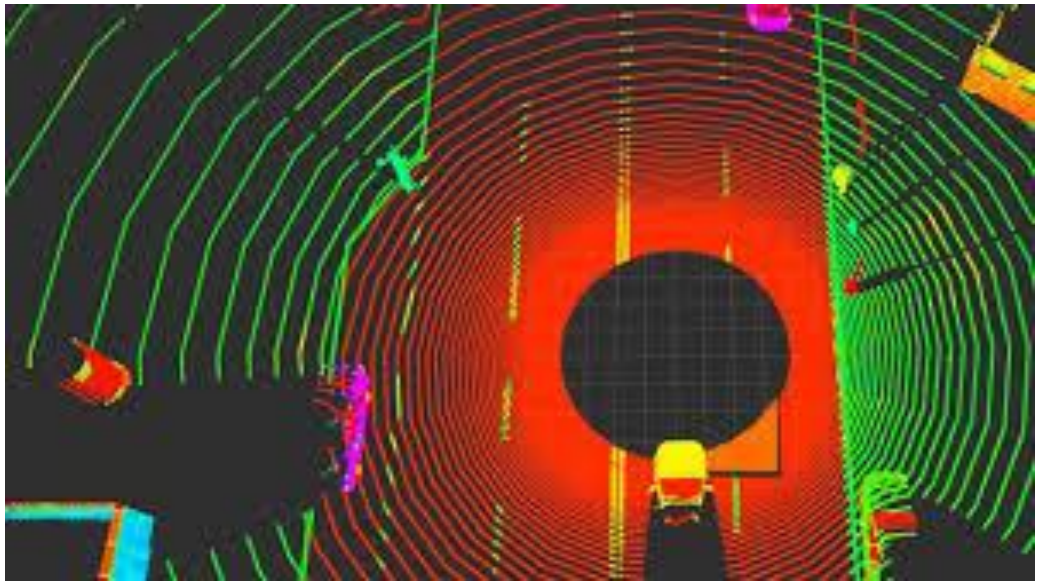
Traffic lights detection.



3. Capability of our platform

- Capability of Autoware and the platform.

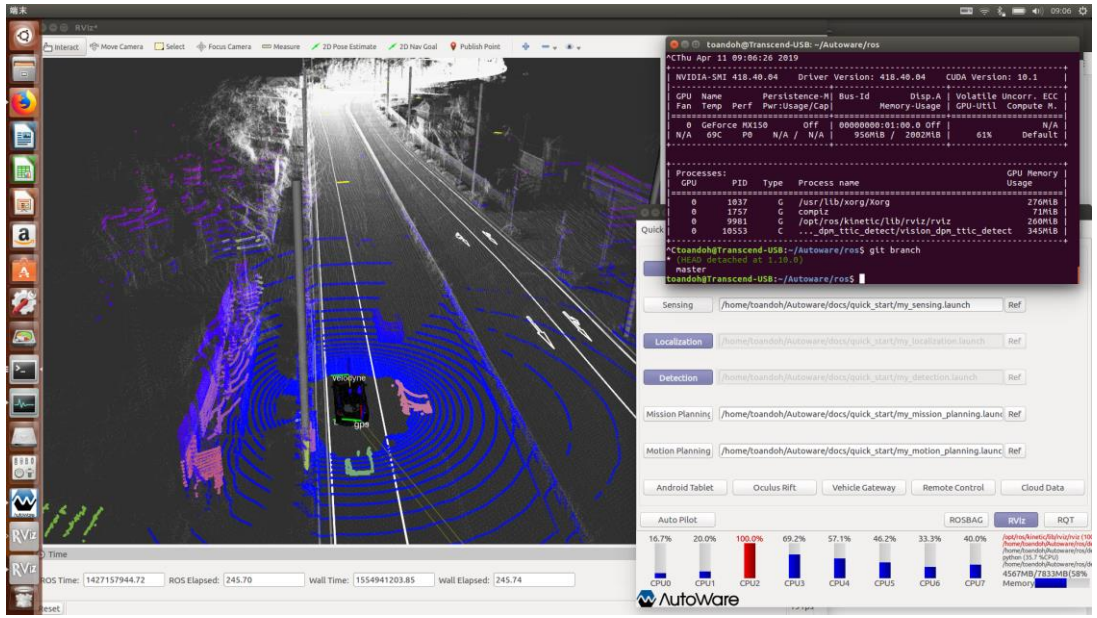
Testing and develop algorithms.



3. Capability of our platform

- Capability of Autoware and the platform.

Can be transferred to real vehicles.



Possible Future Work

- **Traffic lights detection / motion planning / …**
- **ADAS algorithms testing and development.**
- **Achieve rea-time implantation of the project**
- **Achieve L5 autonomous driving vehicle simulation**
- **…….**

- Thanks!

•Q&A