# Polyverif Setup And Installation guide

Below you can find instructions on how to setup the self-driving demo of AWSIM simulation controlled by Autoware. The instruction assumes using the Ubuntu OS.

## Required PC Specs:

Please make sure that your machine meets the following requirements in order to run the simulation correctly:

| | |
|---|---|
| OS | Ubuntu 22.04 jammy 64-bit |
| CPU | 11th Gen Intel® Core™ i7-11700K @ 3.60GHz × 16 |
| GPU | NVIDIA GeForce RTX 4070 |
| Nvidia Driver (Ubuntu 22) | 535.161.07 |

## Localhost settings:

The simulation is based on the appropriate network setting, which allows for trouble-free communication of the AWSIM simulation with the Autoware software. To apply required localhost settings please add the following lines to ~/.bashrc file

```
if  [ ! -e /tmp/cycloneDDS_configured ]; then sudo sysctl -w
net.core.rmem_max=2147483647 sudo ip link set lo multicast on touch
/tmp/cycloneDDS_configured
fi
```

and these lines to ~/.profile or in either of files: ~/.bash_profile *or* ~/.bash_login:

```
export ROS_LOCALHOST_ONLY=1
export RMW_IMPLEMENTATION=rmw_cyclonedds_cpp
```

Note: A system restart is required for these changes to work.

# 1.Download and Run AWSIM Demo Binary

a. Install Vulkan Graphics Library

```
$ sudo apt update

$ sudo apt install libvulkan1
```

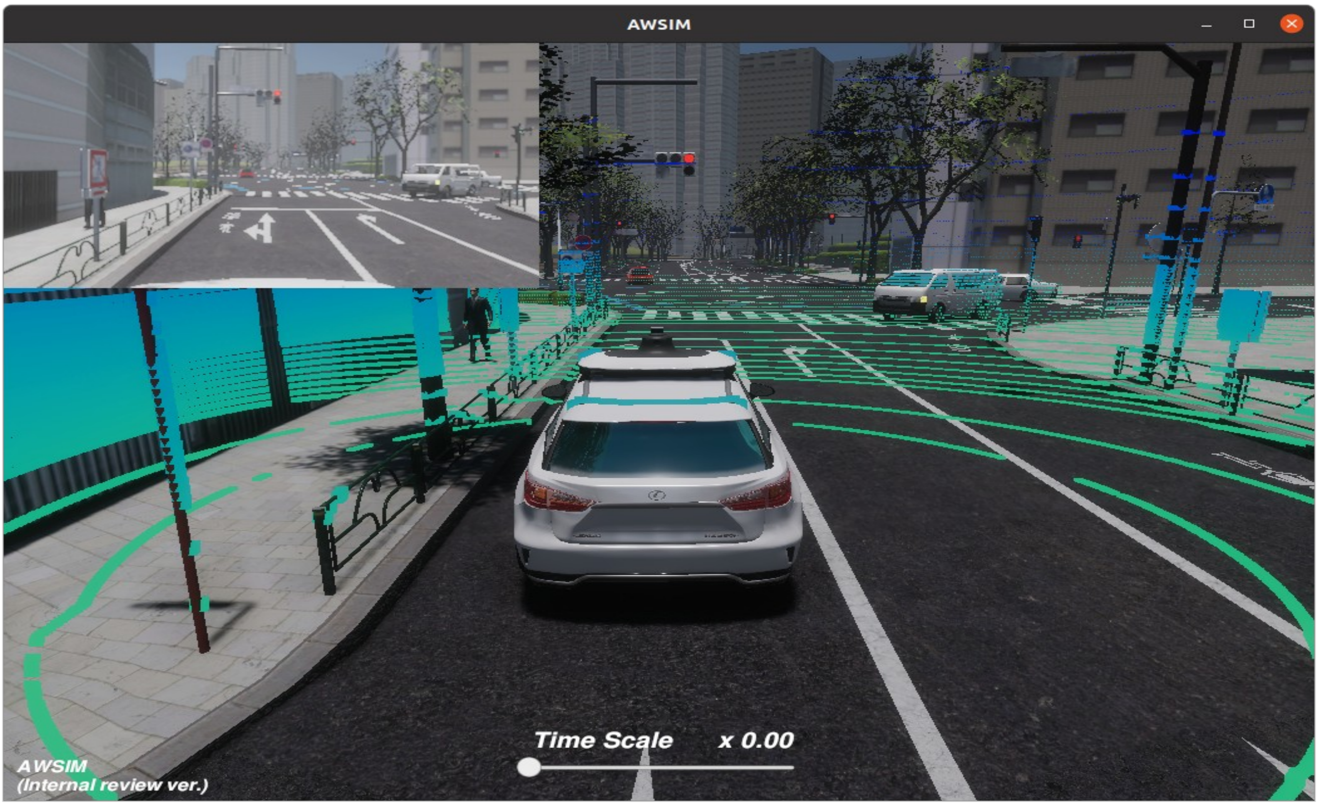
b. In order to configure and run the PolyVerif framwork Run following cmd's

```
 $ mkdir $HOME/adehome/ && git clone --branch PolyVerif-AWSIM --single-
branch https://github.com/MaheshM99/PolyVerif.git $HOME/adehome/

$ wget --directory-prefix="$HOME/adehome" --content-disposition
'https://drive.usercontent.google.com/download?
id=17p4Zne7auBrYpiz1TOMvHVQv3gagt-7E&confirm=t&uuid=a8b3a20c-7bf4-4f3e-
9a1d-e1713d611152' && unzip "$HOME/adehome/AWSIM_v1.2.0.zip" -d
"$HOME/adehome/"
```

c. Launch AWSIM

```
$ ./adehome/AWSIM_v1.2.0/sim.x86_64
```



## 2. Launching Autoware

In order to configure and run the Autoware software with the AWSIM demo, please:

1.Download map files (pcd, osm) in home and unzip them.

https://github.com/tier4/AWSIM/releases/download/v1.1.0/nishishinjuku_autoware_map.zip

2.Clone **Autoware** in home directory and move to the directory.

```
$ git clone https://github.com/autowarefoundation/autoware.git
$ cd autoware
```

3.Switch branch to awsim-stable

```
$ git checkout awsim-stable
```

4.Configure the environment

```
$ ./setup-dev-env.sh
```

5.Create the src directory and clone external dependent repositories into it

```
$ mkdir src
$ vcs import src < autoware.repos
```

6.Install dependent ROS packages

```
$ source /opt/ros/humble/setup.bash
$ rosdep update
$ rosdep install -y --from-paths src --ignore-src --rosdistro $ROS_DISTRO
```

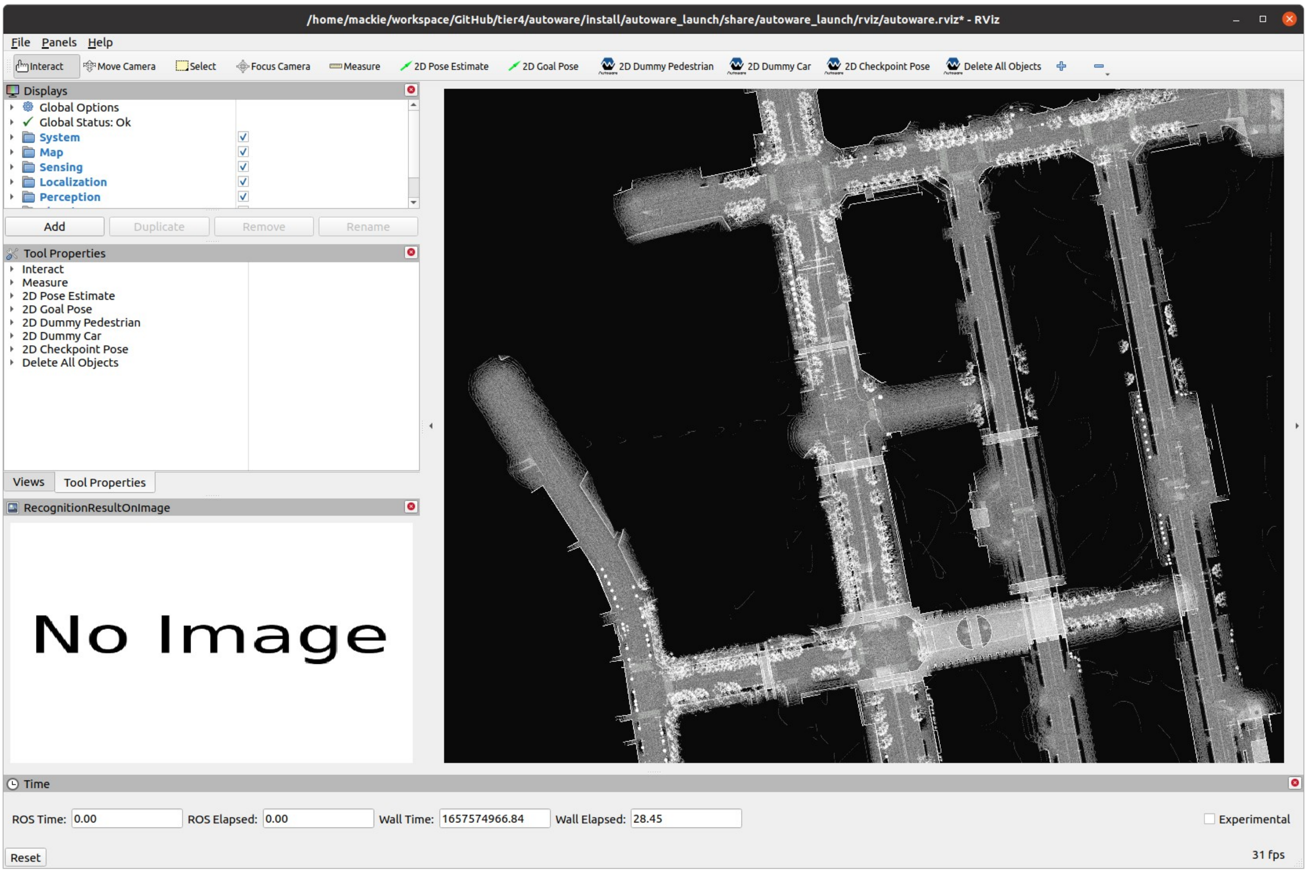
7.Build the workspace

```
$ colcon build --symlink-install --cmake-args -DCMAKE_BUILD_TYPE=Release
-DCMAKE_CXX_FLAGS="-w"
```

8.Launch Autoware.

```
$ source install/setup.bash

$ ros2 launch autoware_launch e2e_simulator.launch.xml
vehicle_model:=sample_vehicle sensor_model:=awsim_sensor_kit
map_path:=<your mapfile location>
```

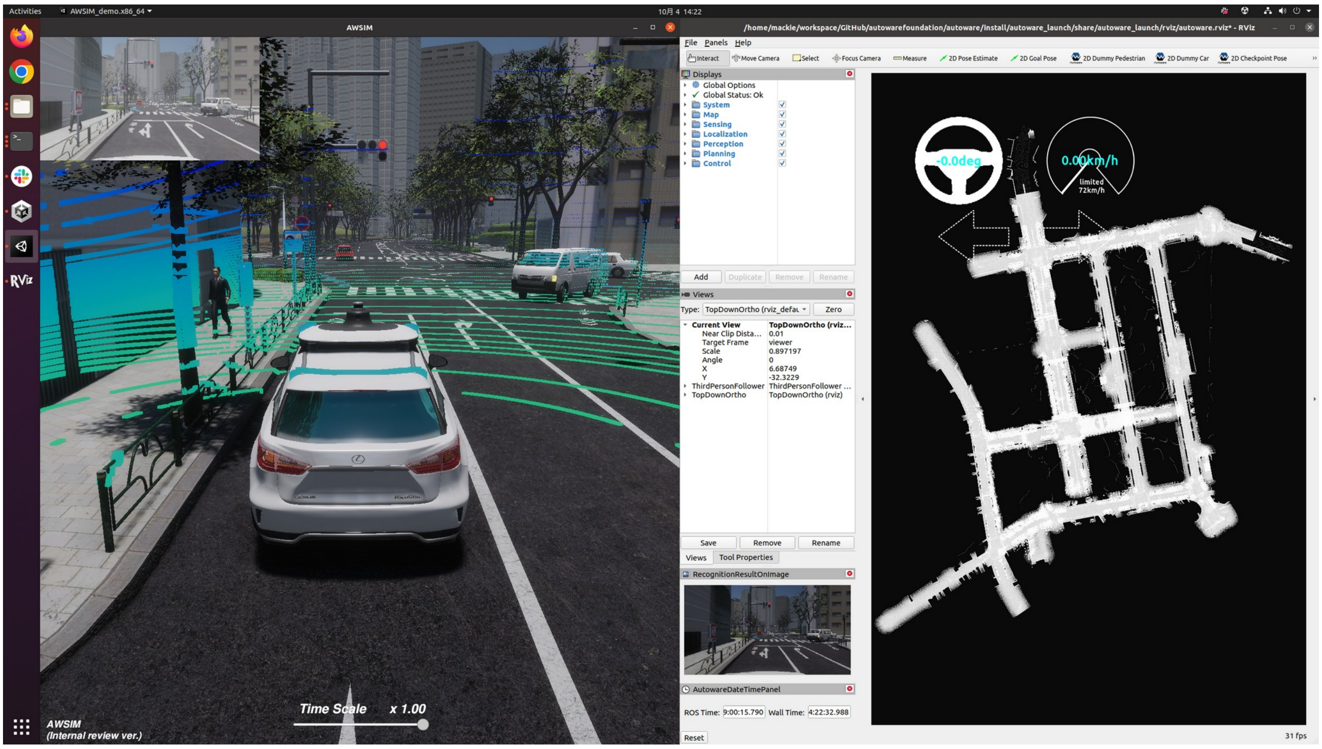


/home/mackie/workspace/GitHub/tier4/autoware/install/autoware_launch/share/autoware_launch/rviz/autoware.rviz* - RViz

ROS Time: 0.00 ROS Elapsed: 0.00 Wall Time: 1657574966.84 Wall Elapsed: 28.45

# 3 Let's run the self-Driving simulation

1.Launch AWSIM and Autoware according to the steps described earlier in this document.

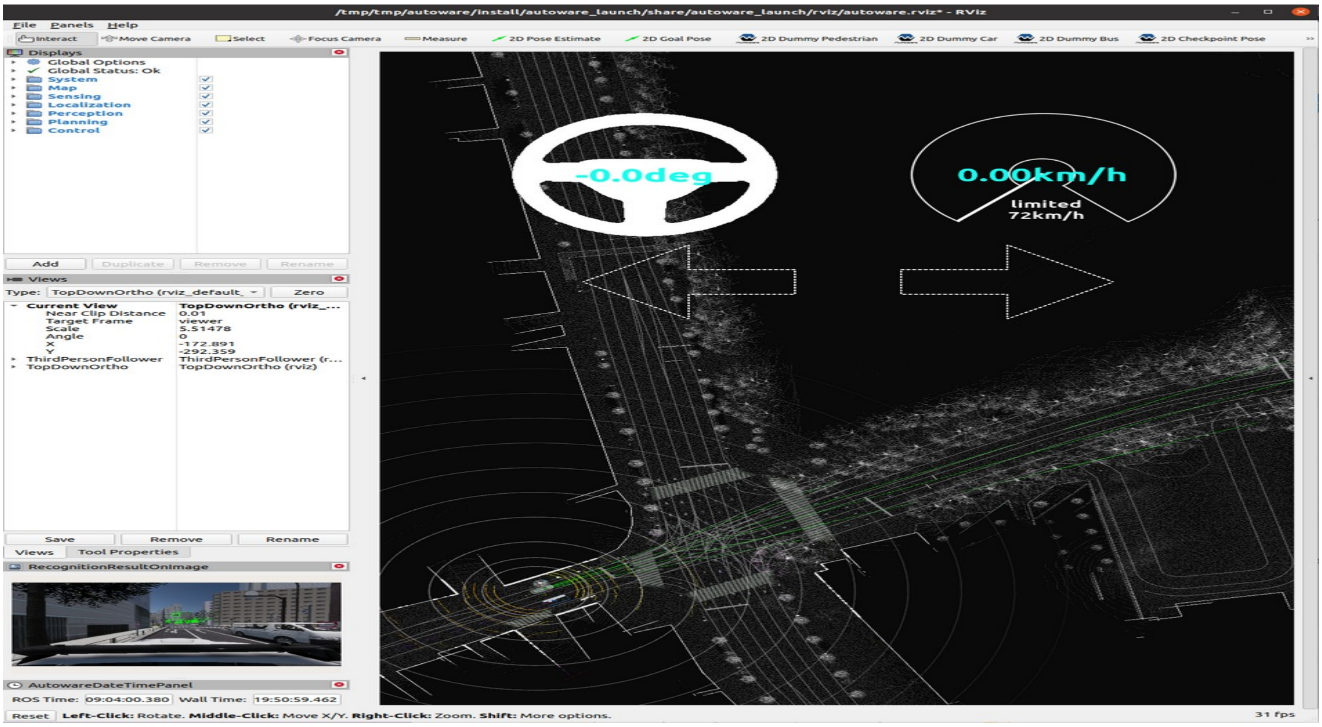


Type: TopDownOrtho (rviz_defau

| Current View | TopDownOrtho (rviz... |
|---|---|
| Near Clip Dista... | 0.01 |
| Target Frame | viewer |
| Scale | 0.897197 |
| Angle | 0 |
| X | 6.68749 |
| Y | -32.3229 |
| ThirdPersonFollower | ThirdPersonFollower ... |
| TopDownOrtho | TopDownOrtho (rviz) |

ROS Time: 9:00:15.790 Wall Time: 4:22:32.988

2.The Autoware will automatically set its pose estimation as presented below.



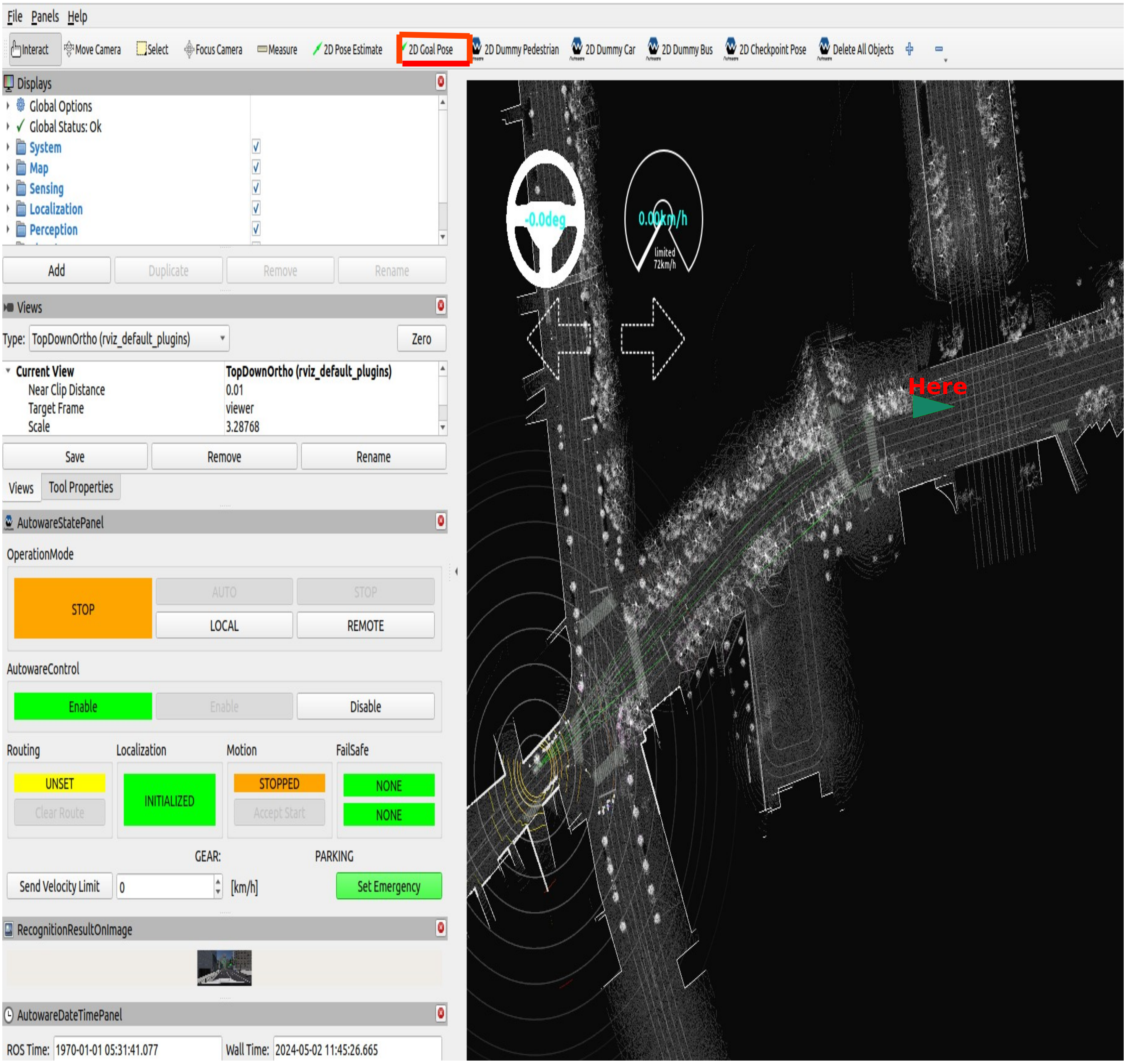
Type: TopDownOrtho (rviz_default_

| Current View | TopDownOrtho (rviz_... |
|---|---|
| Near Clip Distance | 0.01 |
| Target Frame | viewer |
| Scale | 5.51478 |
| Angle | 0 |
| X | -172.891 |
| Y | -292.359 |
| ThirdPersonFollower | ThirdPersonFollower (r... |
| TopDownOrtho | TopDownOrtho (rviz) |

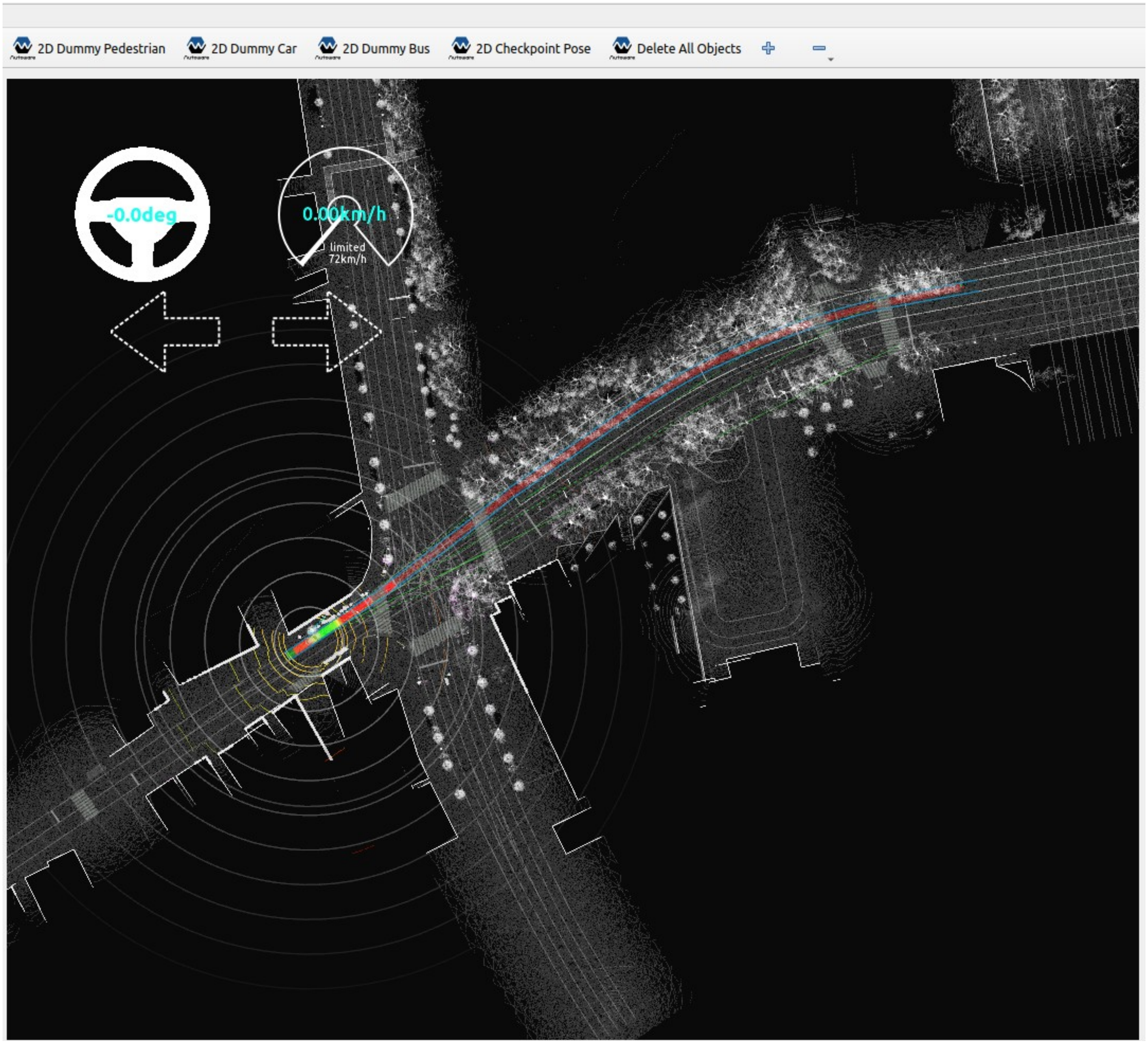
ROS Time: 09:04:00.380 Wall Time: 19:50:59.462

Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click: Zoom. Shift: More options.

3.Set the navigation goal for the vehicle.

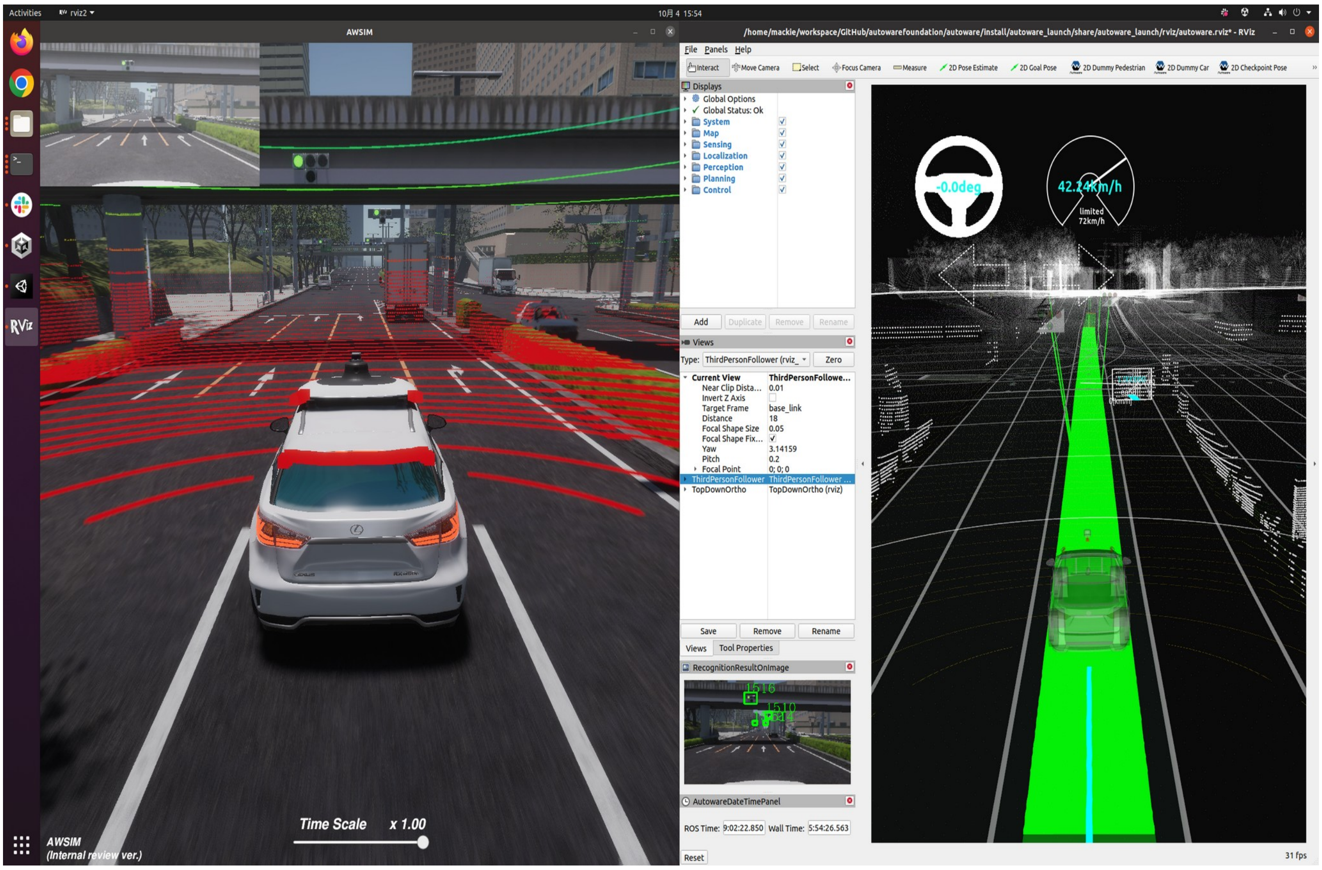


The generated path can be seen.



4. Enable self-driving.

```
$ cd autoware
$ source install/setup.bash
$ ros2 topic pub /autoware/engage autoware_auto_vehicle_msgs/msg/Engage
'{engage: True}' -1
```



Type: ThirdPersonFollower (rviz_

| | |
|---|---|
| Current View | ThirdPersonFollowe... |
| Near Clip Dista... | 0.01 |
| Invert Z Axis | □ |
| Target Frame | base_link |
| Distance | 18 |
| Focal Shape Size | 0.05 |
| Focal Shape Fix... | ☑ |
| Yaw | 3.14159 |
| Pitch | 0.2 |
| Focal Point | 0; 0; 0 |
| ThirdPersonFollower | ThirdPersonFollower ... |
| TopDownOrtho | TopDownOrtho (rviz) |

ROS Time: 9:02:22.850 Wall Time: 5:54:26.563

The self-driving simulation demo has been successfully launched !!!!!!