PolyVerif User Guide

Follow the below steps How to use PolyVerif Using Scripts

In order to configure and run the PolyVerif Pipelines,please download AWSIM_WS file in home dir and unzip it using below link.

https://drive.google.com/file/d/1RTTourPVnBTn0NuZ4WgL5v5d6YHLV5uY/view?usp=sharing

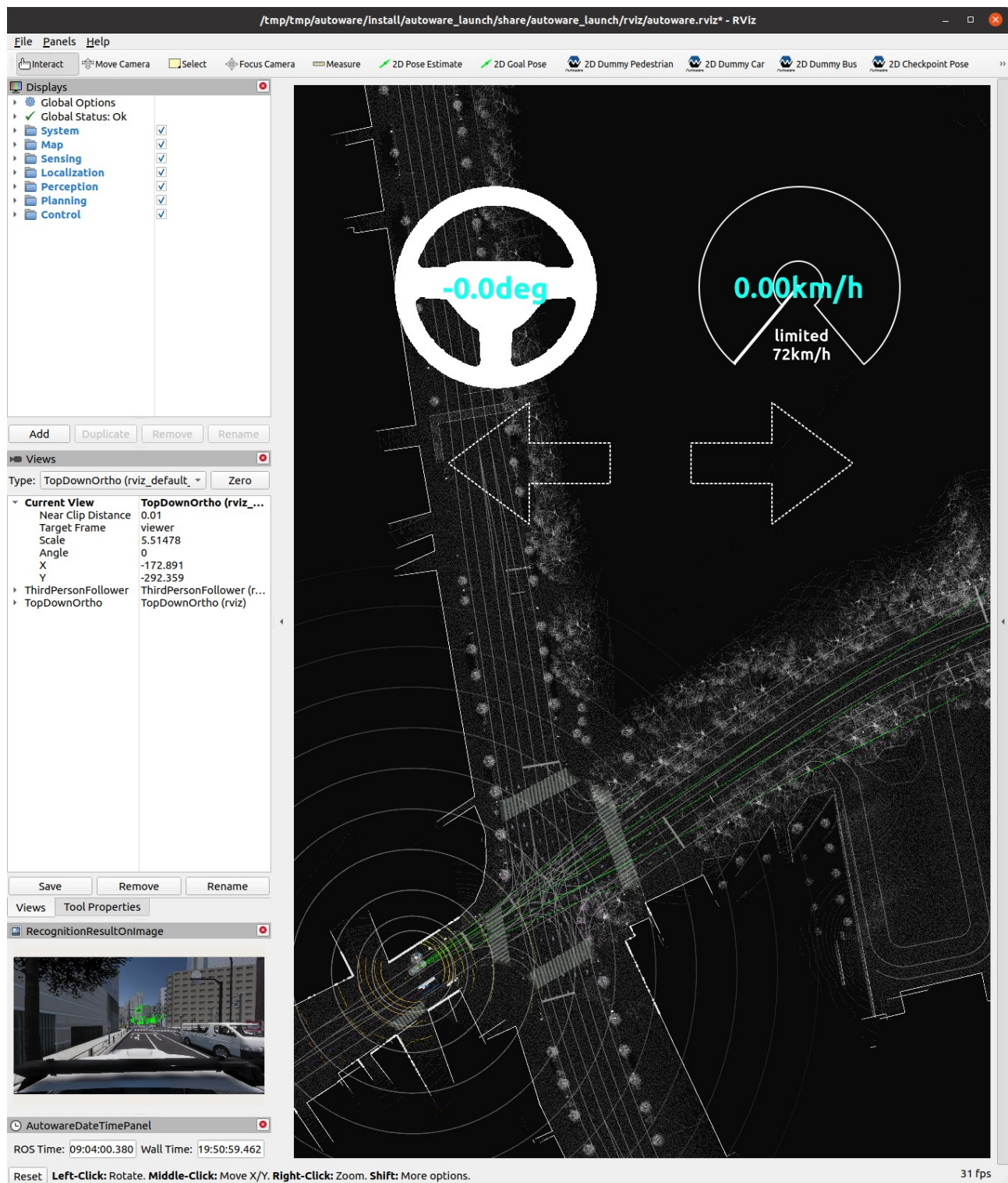1.Open a new terminal, change dir to AWSIM_WS and launch AWSIM and autoware using below cmds

```
$ cd AWSIM_WS

$ ./launch
```

2.Wait for while to launch awsim and autoware,once launched you can observe all necessary componet get intiliased in both simulator and rviz (map,ego,ego position,sensor etc).
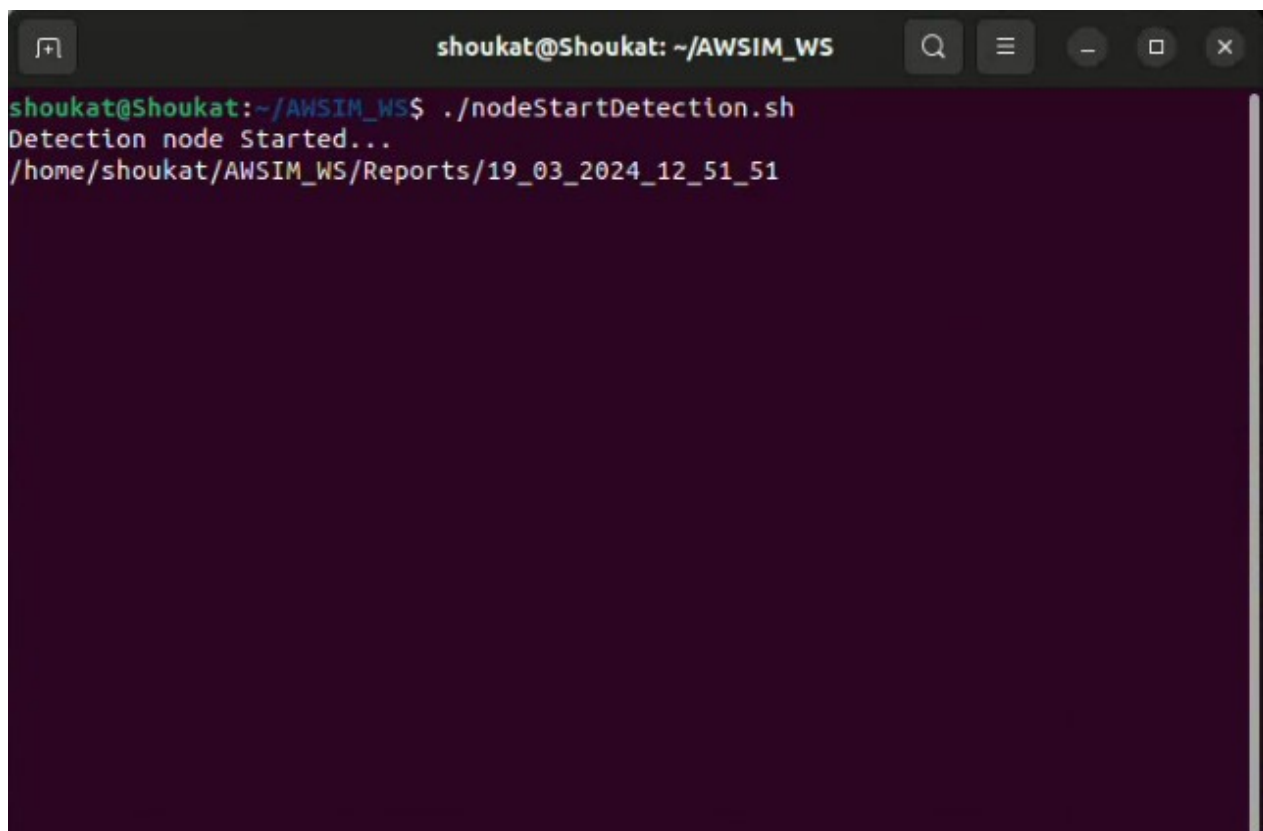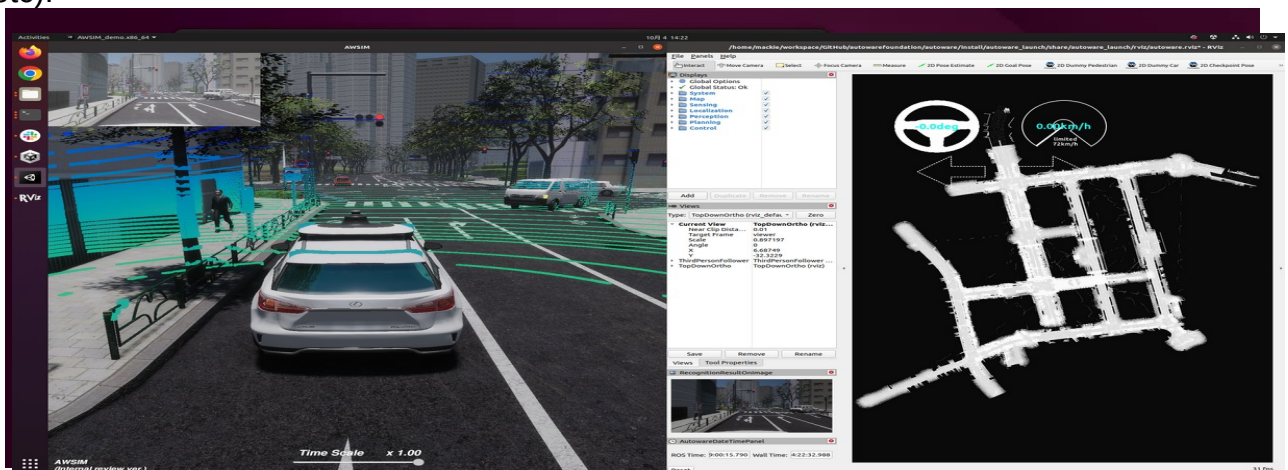
3.The Autoware will automatically set its pose estimation as presented below.

If ego position not initializing properly then you can set ego position using **2D pose estimate** button and drag arrow on map.

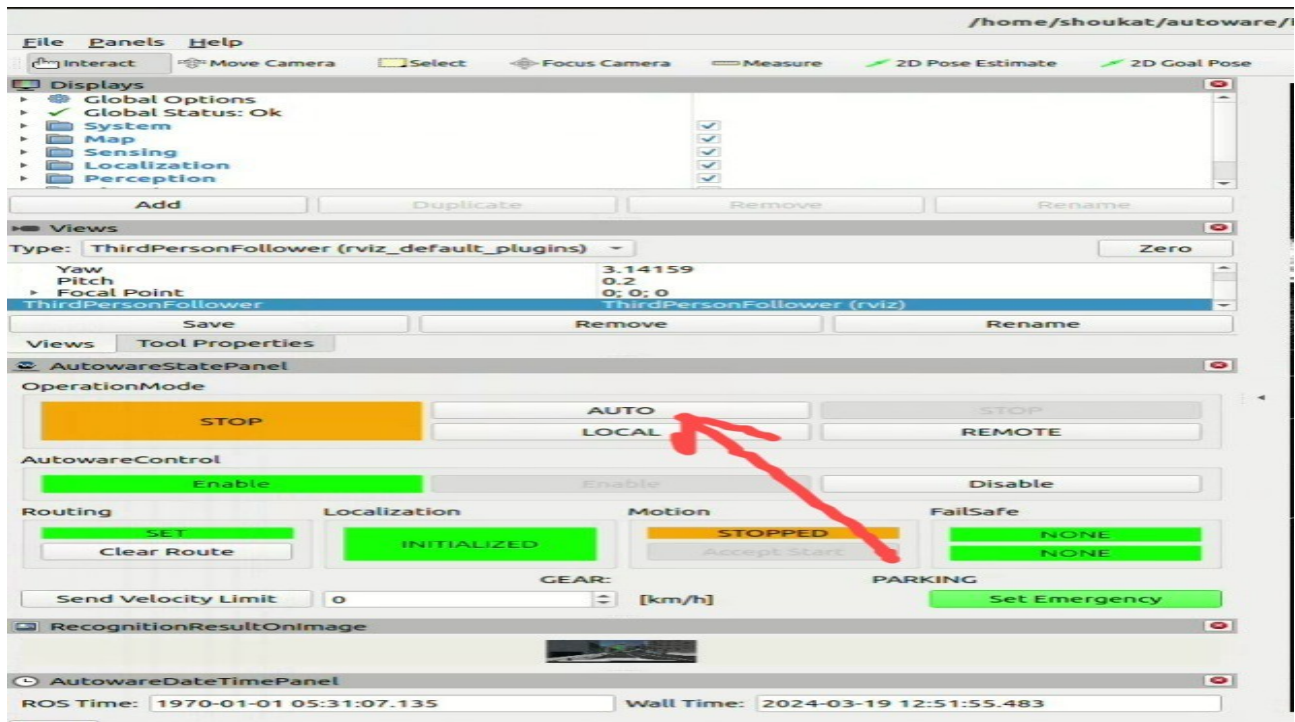4. Open a new terminal change AWSIM dir and start detection node using below cmd.

```
$ cd AWSIM_WS
$ ./nodeStartDetection.sh
```
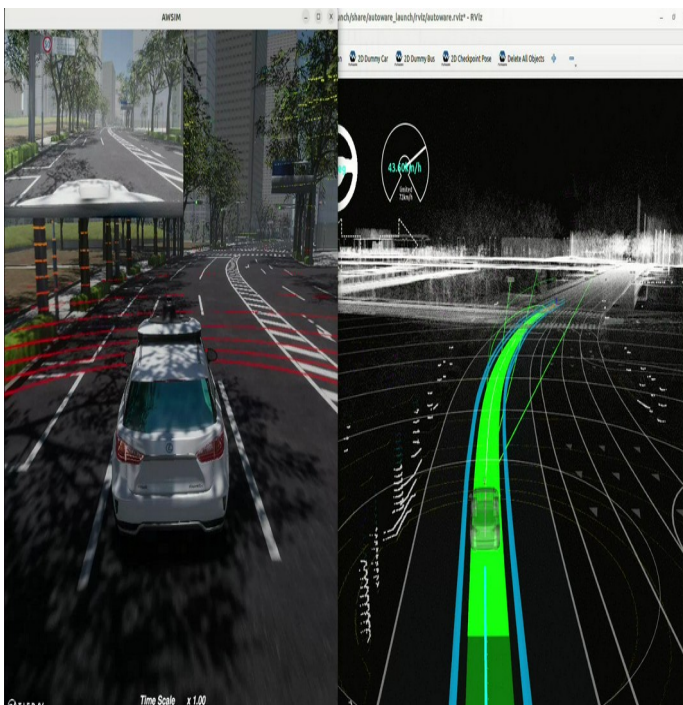
5.Once DetectionNode started then start scenario by clicking on button AUTO on AutowareStatePanel and scenario get started

Let complete the scenario as shown below images

6.Once execution of seen get complete then stop detection node on terminal by closing the terminal or by keyboard interrupt **pressing CTR + C**

```
shoukat@Shoukat:~/AWSIM_WS$ ./nodeStartDetection.sh
Detection node Started...
/home/shoukat/AWSIM_WS/Reports/19_03_2024_12_51_51
CurrDir: /home/shoukat/AWSIM_WS
CTraceback (most recent call last):
  File "/home/shoukat/AWSIM_WS/Node/Node_perception_validation_ws/node_perception_validation_ws/install/node_perception_validation/lib/node_perception_validation/de
    sys.exit(load_entry_point('node-perception-validation==0.0.0', 'console_scripts', 'detection_node')())
  File "/home/shoukat/AWSIM_WS/Node/Node_perception_validation_ws/node_perception_validation_ws/install/node_perception_validation/lib/python3.10/site-packages/node
, line 138, in main
    rclpy.spin(detetction_subscriber)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/__init__.py", line 222, in spin
    executor.spin_once()
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/executors.py", line 739, in spin_once
    self._spin_once_impl(timeout_sec)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/executors.py", line 728, in _spin_once_impl
    handler, entity, node = self.wait_for_ready_callbacks(timeout_sec=timeout_sec)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/executors.py", line 711, in wait_for_ready_callbacks
    return next(self._cb_iter)
  File "/opt/ros/humble/local/lib/python3.10/dist-packages/rclpy/executors.py", line 608, in _wait_for_ready_callbacks
    wait_set.wait(timeout_nsec)
KeyboardInterrupt
[ros2run]: Interrupt
shoukat@Shoukat:~/AWSIM_WS$
```

7.To generate reports run the report generation script using cmd $ ./reportGenerateDetection.sh

```
shoukat@Shoukat: ~/AWSIM_WS
shoukat@Shoukat:~/AWSIM_WS$ ./reportGenerateDetection.sh
```

```
shoukat@Shoukat: ~/AWSIM_WS          X          Detection Validation Result          X

location :  /home/shoukat/AWSIM_WS
Calculating Detection Params.
Entering in TimeStampMatching function
Exiting from TimeStampMatching function
Entering in TimeStampMatching function
Exiting from TimeStampMatching function
Entering in ObjectDetectionRate function
```

7.Find the generated reports in dir path as shown below
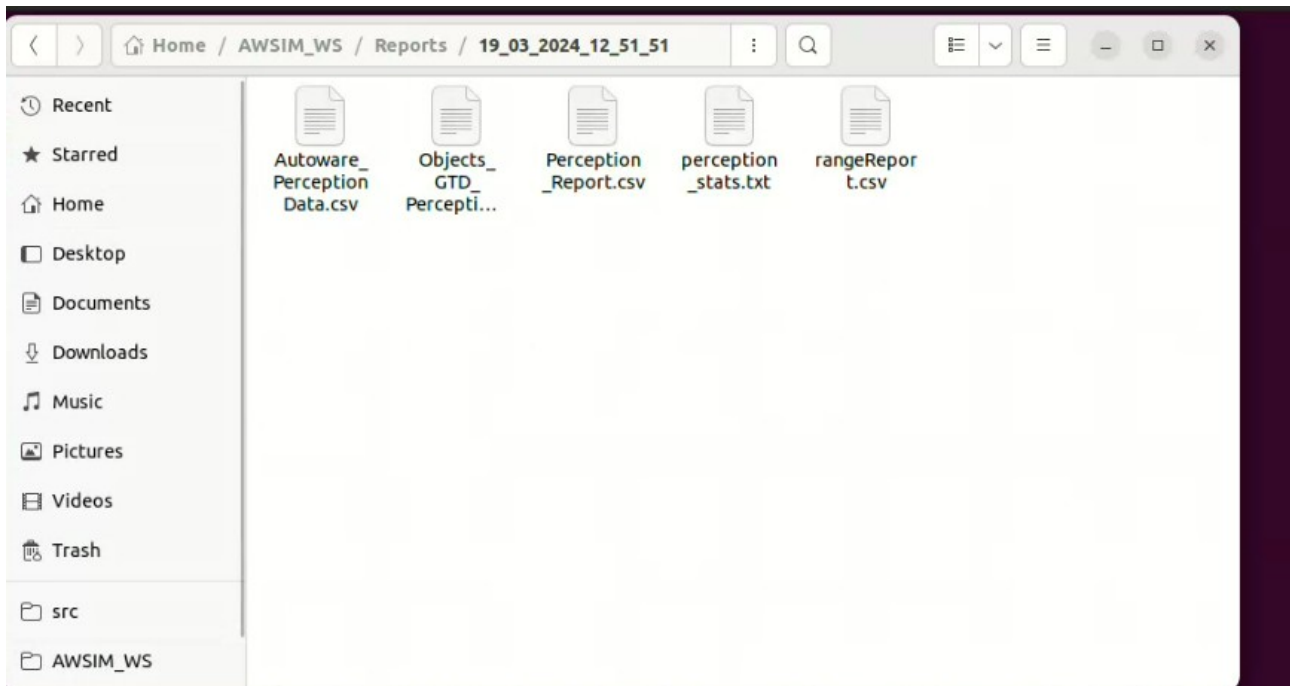
8. From above steps DetectionNode pipeline completed and for other pipelines follow same steps as earlier performed.