

2.Handheld Lidar Mapping

This function requires starting the program in the slam_gmapping function package. The source code is located in the [yahboomcar_ws] folder.

Here we use the matching virtual machine to explain how to start the program. If you want to put it on your own RDK motherboard, put yahboomcar_ws in the root directory and compile it.

2.1 Start lidar

Input following command in the terminal.

```
ros2 launch ydlidar_ros2_driver ydlidar_launch.py
```

2.2 Release static odom conversion

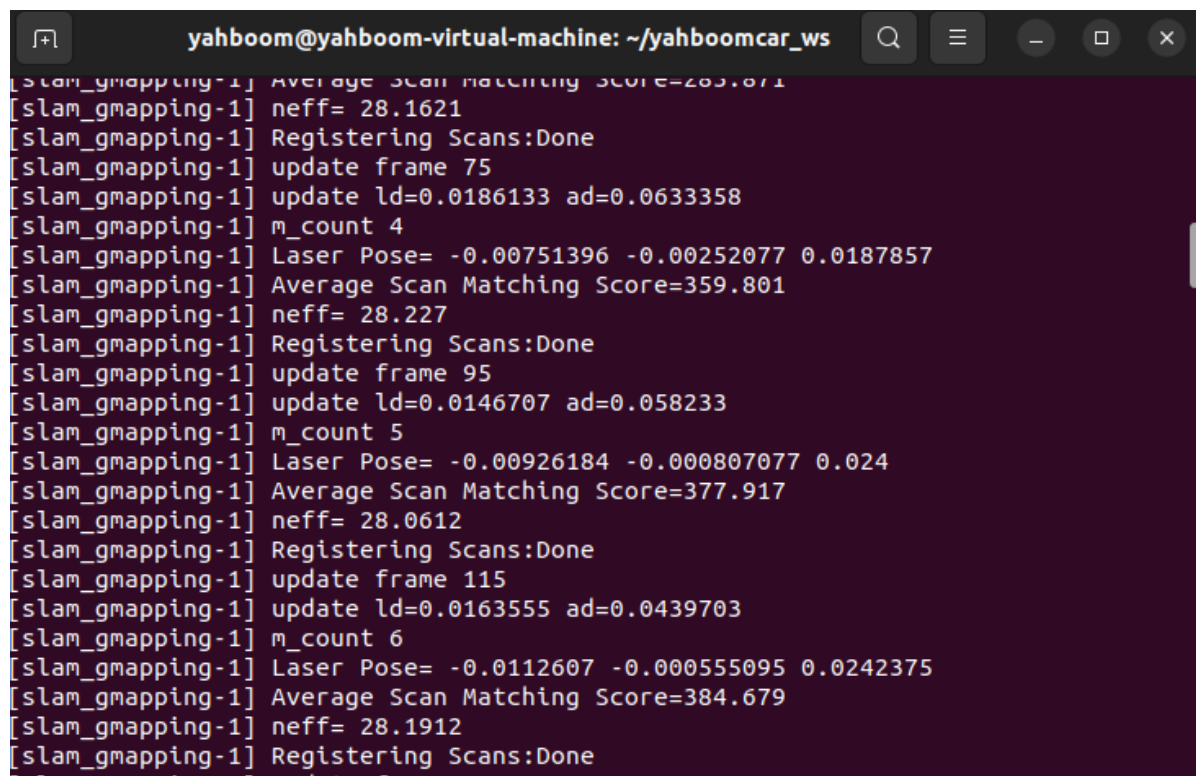
Input following command in the terminal.

```
ros2 launch rf2o_laser_odometry rf2o_laser_odometry.launch.py
```

2.3 Start gmapping mapping

Input following command in the terminal.

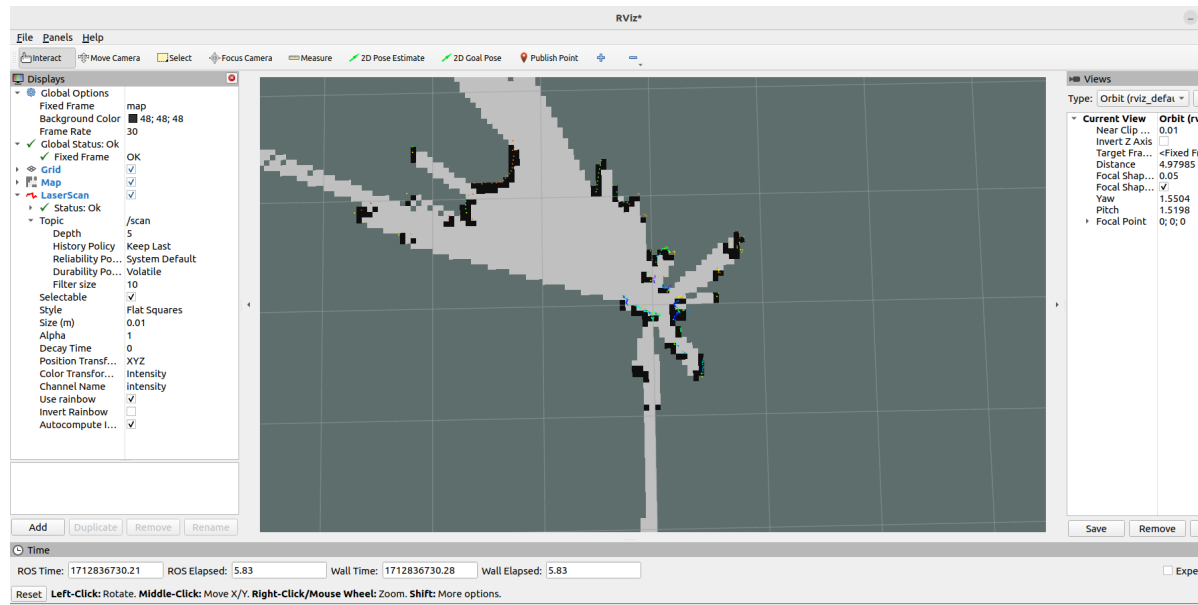
```
ros2 launch slam_gmapping slam_gmapping.launch.py
```

A terminal window titled 'yahboom@yahboom-virtual-machine: ~/yahboomcar_ws' displays the output of the 'ros2 launch slam_gmapping slam_gmapping.launch.py' command. The output shows a series of log messages from the 'slam_gmapping-1' process, including 'Average Scan Matching Score', 'neff', 'Registering Scans:Done', 'update frame', 'update ld', 'ad', 'm_count', and 'Laser Pose' values. The scores are 285.871, 359.801, 377.917, and 384.679. The 'neff' values are 28.1621, 28.227, 28.0612, and 28.1912. The 'Laser Pose' values are -0.00751396 -0.00252077 0.0187857, -0.00926184 -0.000807077 0.024, and -0.0112607 -0.000555095 0.0242375. The terminal window has a dark background and standard window controls at the top.

```
yahboom@yahboom-virtual-machine: ~/yahboomcar_ws
[slam_gmapping-1] Average Scan Matching Score=285.871
[slam_gmapping-1] neff= 28.1621
[slam_gmapping-1] Registering Scans:Done
[slam_gmapping-1] update frame 75
[slam_gmapping-1] update ld=0.0186133 ad=0.0633358
[slam_gmapping-1] m_count 4
[slam_gmapping-1] Laser Pose= -0.00751396 -0.00252077 0.0187857
[slam_gmapping-1] Average Scan Matching Score=359.801
[slam_gmapping-1] neff= 28.227
[slam_gmapping-1] Registering Scans:Done
[slam_gmapping-1] update frame 95
[slam_gmapping-1] update ld=0.0146707 ad=0.058233
[slam_gmapping-1] m_count 5
[slam_gmapping-1] Laser Pose= -0.00926184 -0.000807077 0.024
[slam_gmapping-1] Average Scan Matching Score=377.917
[slam_gmapping-1] neff= 28.0612
[slam_gmapping-1] Registering Scans:Done
[slam_gmapping-1] update frame 115
[slam_gmapping-1] update ld=0.0163555 ad=0.0439703
[slam_gmapping-1] m_count 6
[slam_gmapping-1] Laser Pose= -0.0112607 -0.000555095 0.0242375
[slam_gmapping-1] Average Scan Matching Score=384.679
[slam_gmapping-1] neff= 28.1912
[slam_gmapping-1] Registering Scans:Done
[slam_gmapping-1] update frame 135
```

rviz displays as follows.

If there is no map display, click file --> Open config --> sunrise --> yahboomcar_ws --> src --> slam_gmapping --> rviz --> map_view.rviz in the upper left corner to open the rviz map display configuration.



2.4 View TF tree

Input following command in the terminal.

```
ros2 run tf2_tools view_frames
```

```
yanboom@yanboom-virtual-machine:~/yahboomcar_ws$ ros2 run tf2_tools view_frames
[INFO] [1712836863.067737751] [view_frames]: Listening to tf data for 5.0 second
S...
[INFO] [1712836868.112822280] [view_frames]: Generating graph in frames.pdf file
...
[INFO] [1712836868.118691540] [view_frames]: Result:tf2_msgs.srv.FrameGraph_Response(frame_yaml="odom: \n parent: 'map'\n broadcaster: 'default_authority'\n rate: 20.197\n most_recent_transform: 1712836868.110051\n oldest_transform: 1712836863.059761\n buffer_length: 5.050\nlaser_frame: \n parent: 'base_link'\n broadcaster: 'default_authority'\n rate: 10000.000\n most_recent_transform: 0.000000\n oldest_transform: 0.000000\n buffer_length: 0.000\nbase_link: \n parent: 'base_footprint'\n broadcaster: 'default_authority'\n rate: 10000.000\n most_recent_transform: 0.000000\n oldest_transform: 0.000000\n buffer_length: 0.000\nbase_footprint: \n parent: 'odom'\n broadcaster: 'default_authority'\n rate: 10.237\n most_recent_transform: 1712836867.890208\n oldest_transform: 1712836863.005946\n buffer_length: 4.884\n")
yanboom@yanboom-virtual-machine:~/yahboomcar_ws$
```

The system will generate a frames.pdf file in the directory where the command terminal is started. This is the generated TF tree.

view_frames Result

Recorded at time: 1712836868.1316395

