rplidar_python tutorial

Package Summary

A ROS node for rplidar written in python, which (you may need to change port name, if you wanna custom you setting).

Maintainer status: maintained
Maintainer: Zhihao Xu <xu001314@163.com>
Download:https://github.com/DinnerHowe/rplidar_python.git
Author: Zhihao Xu
License: TODO

Overview

the rplidar_python package provides a solution for RPlidar sensor usage in ros. This
packge also allow robot launches a 360 degree scanning map through gmapping module
without twisting.

Hardware Requirements

to use rplidar_python, you should get a robot that provides odometry, like turtlebot. Also,
you need a RPlidar sensor. Here we use RPLIDAR 360 laser scanner development kit.
we use RPlidar to replace kinect sensor and we mount it in the position of kinect , thus
kinect tf frame is useful for RPlidar sensor as well.

Launch Example

to make a map by RPlidar, you should launch rplidar_gmapping_demo.launch.

```
roslaunch rplidar_python rplidar_gmapping_demo.launch
```

Or you may wanna see laser frame only by typing following command

```
roslaunch rplidar_python rplidar_demo.launch
```

Nodes

rplidar_scan_ver3.py

driver for RPlidar. Automatically starts sensor and convert data stream into sensor_msgs/LaserScan type. sensor publish topic every frame, one frame contain 360
laser data.

Published Topic

/scan(sensor_msgs/LaserScan)

output Laser scans to create the map from

Parameters

range_min (flaot default 0.15)
   the min range that laser can scan

range_max (float default 6.0)
   the min range that laser can scan

frame_id (string default 'laser')
   rplidar frame

angle_max(float default pi)
   the max angle that laser can reach

angle_min(float default -pi)
   the min angle that laser can reach

angle_increment(float default -0.017453292519943295)
   angular distance between measurements

scan_time (float)
   time between scans

ranges (float[])
   range data

RPlidar c++ tutorial

Please check at here