Report of LAB3

By Younglar Han and Echo Li

The code is attached to the report as a python script file.

1. Get the data from LiDAR

The data from LiDAR is published as a topic called "**/scan**". To learn more about the topic, we can use command as follow:

```
rostopic echo /scan
```

2. Operate the data from LiDAR

To do this, we need to create a node, which subscribes the "/scan" topic and publishes a "/revised_scan" topic.

```
 #! /usr/bin/env python

import rospy
from sensor_msgs.msg import LaserScan
import sensor_msgs.msg

pub = rospy.Publisher('/revised_scan', LaserScan, queue_size = 10)
scann = LaserScan()

def callback(msg):
    current_time = rospy.Time.now()
    scann.header.stamp = current_time
    scann.header.frame_id = 'laser'
    scann.angle_min = 0.0
    scann.angle_max = 6.28318458203
    scann.angle_increment = 0.0174532923847
    scann.time_increment = 2.98899994959e-05
    scann.scan_time = msg.scan_time
    scann.range_min = 0.119999997318
    scann.range_max = 3.5
    scann.ranges = msg.ranges[180]
    scann.intensities = msg.intensities
    pub.publish(scann)

def listener():
    rospy.init_node('revised_scan', anonymous=True)
    sub = rospy.Subscriber('/scan', LaserScan, callback)
    rospy.spin()

if __name__ == '__main__':
    listener()
```

And all we need to do is to modify the scan data in the callback function.

3. Get the real minimum and maximum ranges of LiDAR

We need to collect all of the scan data during a period of time, and find the minimum and maximum of the dataset.

We can get the minimum and maximum in **msg.ranges** when the callback function is called. And use two global variables to store the real maximum(or minimum) in the maximums(or minimums).

In our code, we realize the idea as follow:(just the callback function)

```
real_range_max_golbal = 0.0
real_range_min_global = 10.0

def callback(msg):
    global real_range_max_golbal
    global real_range_min_global

    # detect the min and max range #
    real_range_max = max(msg.ranges)
    real_range_min = 10.0
    for i in range(len(msg.ranges)):
        if msg.ranges[i] < real_range_min and msg.ranges[i] != 0.0:
            real_range_min = msg.ranges[i]
    if real_range_max > real_range_max_golbal:
        real_range_max_golbal = real_range_max
    if real_range_min < real_range_min_global:
        real_range_min_global = real_range_min
    print('real range max =', real_range_max_golbal, '  real range min =',
real_range_min_global)
```

After code execution, we got that **real minimum of range is 0.091m and real maximum range is 4.198m**

4. Get the accuracy and precision of LiDAR

In this part, we will select the first element of every **msg.ranges** as the sample points, which reflects the distance between the center of the LiDAR rotation and the obstacle right in front of the LiDAR. By computing the range of this distance dataset, we can get the precision of LiDAR. By comparing the mean of the dataset and the ground truth, we can get the accuracy of LiDAR.

In our code, we realize the idea as follow:(just the callback function):

```
sample_point_max = 0.0
sample_point_min = 10.0
sample_point_sum = 0.0
sample_point_mean = 0.0
sample_point_num = 0

def callback(msg):
    global sample_point_max
    global sample_point_min
    global sample_point_sum
    global sample_point_mean
    global sample_point_num
```

```
13
14      # compute the precision and accuracy #
15      if msg.ranges[0] > sample_point_max and msg.ranges[0] != 0.0:
16          sample_point_max = msg.ranges[0]
17      if msg.ranges[0] < sample_point_min and msg.ranges[0] != 0.0:
18          sample_point_min = msg.ranges[0]
19      if msg.ranges[0] != 0.0:
20          sample_point_sum += msg.ranges[0]
21          sample_point_num += 1
22          sample_point_mean = sample_point_sum / sample_point_num
23
24      print('sample point max =', sample_point_max, '  sample point min =',
    sample_point_min)
25      print('sample point mean =', sample_point_mean)
26      print('precision =', sample_point_max - sample_point_min)
```

After executing the code 3 time at difference position, we got this table(distance in m)

max	min	mean	ground truth	precision	accuracy(abs)
0.2910	0.2900	0.2902	0.2861	0.001	0.0041
0.5200	0.5180	0.5192	0.5100	0.002	0.0092
1.6580	1.6250	1.6389	1.6520	0.033	0.0131