

智能机器人概论作业二报告

2100013104 尹骄洋

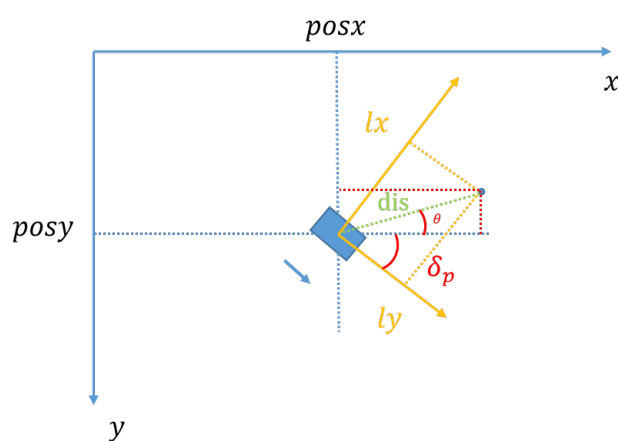
October 21st 2023

1 实验目的

- 通过利用激光及定位数据，实现栅格地图的计算。

2 实验原理

2.1 障碍物世界坐标计算原理



已知激光点相对小车角度 δ_i 与距离 dis ，可得激光束小车坐标系位置 $[lx, ly]$ 。已知小车位置 $[x_p, y_p]$ 与姿态 δ_p 。计算可得 $\theta = \delta_i - \delta_p$ 。随后即可根据小车坐标得到激光点（障碍物）的世界坐标 $[x_g, y_g]$

2.2 简易投票法

先将地图上各点设置为 0，每计算得出一个激光点的全局坐标之后转化为地图坐标，然后相应点的值增加（我这里每次加 10），直到 255 为止。颜色越深是障碍物的可能性就越高。最后即可实现地图的可视化。

3 实验数据

- 激光扫描数据: *URG_X_20130903_195003.lms*
- 定位数据: 小作业 1 生成的小车行驶轨迹

4 实验代码

ProcessorMulti_Algorithm_Mapping_PrivFunc.cpp 中 `bool DECOFUNC` 补全部分如下:

```
1
2 // ===== 请完成以下程序段! =====
3 // 遍历所有激光点, 计算它们在全局坐标系中的位置
4
5 const double pi = 3.1415926535897934384626;
6 double posx = inputdata_0.front()->x * 10.0;
7 double posy = inputdata_0.front()->y * 10.0;
8 double pos_ori = inputdata_0.front()->orientation;
9
10 for(int i=0;i<inputdata_1.front()->datasize;i++){
11
12     // 计算得到单个激光点的距离(m)和角度(rad)
13     double dis = inputdata_1.front()->data[i] / 10.0;
14     if (dis==0)
15         continue; // get rid of invalid points
16
17     double angle = pi * i / 360.0;
18     double lx = dis * cos(angle);
19     double ly = dis * sin(angle);
```

```

20
21 // 激光点在全局坐标系中的位置 单位m
22 double gx, gy;
23 LP2GP(lx, ly, posx, posy, pos_ori, &gx, &gy);
24
25 // 激光点在栅格地图中的位置 单位 pixel
26 int mapx = gx+100;
27 int mapy = gy+500;
28
29 // 更新地图
30 if (mapx >= 0 && mapx < params->mapWidth
31 && mapy >= 0 && mapy < params->mapHeight)
32 {
33     if (vars->map[mapy][mapx] < 245)
34         vars->map[mapy][mapx] += 10;
35     else
36         vars->map[mapy][mapx] = 255;
37 }
38 }
39 // ===== 请完成以上程序段！ =====

```

transform.cpp 补全如下:

```

1 void LP2GP(double lx, double ly, double posx,
2 double posy, double ori, double * gx, double * gy) {
3     //此处需完成
4     //输入: lx, ly 为激光坐标系下激光点坐标;
5     //      posx, posy, ori 为机器人当前位姿;
6     //输出: gx, gy 即函数输出的变换后的全局坐标
7     double pi = 3.1415926535897934384626;
8     double theta = pi - (atan(lx/ly) - ori);
9     double dis = sqrt(lx*lx + ly*ly);
10    *gx = posx + dis * cos(theta);
11    *gy = posy - dis * sin(theta);
12 }

```

5 地图可视化

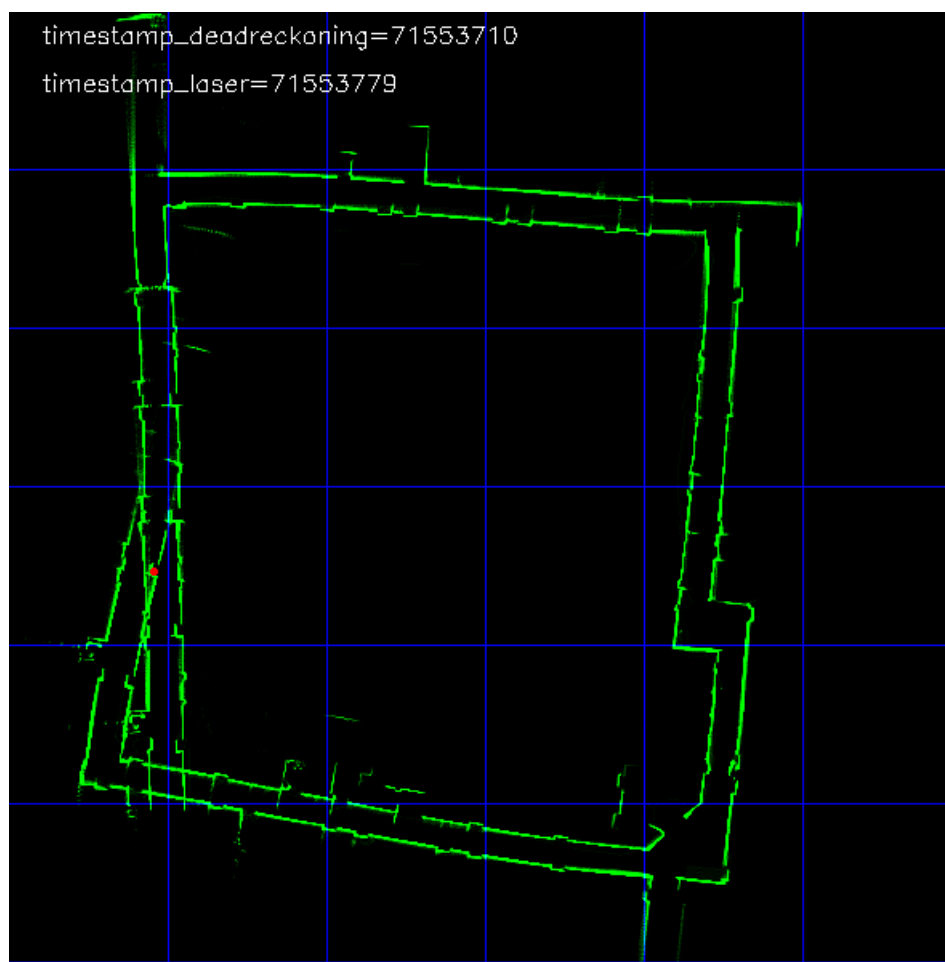


Figure 1: 实验二结果