

使用说明

文件说明

文件说明

src文件夹的内容

racecar文件夹的内容

launch文件夹的内容

racecar/src下的内容

复现说明

技术手册

速度快但建图好的关键

循迹相关参数的调节：

导航相关参数的调节

最关键的代码是auto_nav.py

解耦调试

经验总结

使用说明

文件说明

文件说明

实验源码文件夹存放的是与技术报告有关的代码，为区域赛时所写，国赛时并未更新

代码文件夹存放了三份代码

这里的速度均指第一圈的速度，第二圈导航的速度为2075，时间大约为11s

x1850为低速版保底代码，第一圈（包含停车的3s）时间大约为30s

x1925为高速版本，第一圈时间为24s

planB为备用方案，当第二圈导航无法正常使用时，使用这个方案。这个方案两圈都是跑雷达循迹

src文件夹的内容

名称	修改日期	类型	大小
encoder_driver	2023/12/20 4:17	文件夹	
imu_launch	2023/12/20 4:17	文件夹	
imu_tool	2023/12/20 4:17	文件夹	
lidar_tracking	2023/12/20 4:17	文件夹	雷达循迹模块
lidar_undistortion_2d	2023/12/20 4:17	文件夹	雷达去畸变模块，不使用
ls01x	2023/12/20 4:17	文件夹	
my_slam_gmapping	2023/12/20 4:17	文件夹	重写的gmapping 不使用
racecar	2023/12/20 4:17	文件夹	关键代码在个文件夹下
racecar_driver	2023/12/20 4:17	文件夹	
rf2o_laser_odometry	2023/12/20 4:17	文件夹	
serial	2023/12/20 4:17	文件夹	
serial_imu	2023/12/20 4:17	文件夹	
CMakeLists.txt	2023/12/5 19:45	文本文档	1 KB

my_slam_gmapping重写的gmapping，只是当时测试的残留，在实际代码中并未使用到，不用关注，如果你要研究

链接: [我手写了个SLAM算法! - 知乎\(zhihu.com\)](#)

[给gmapping加激光雷达运动畸变去除模块 - 知乎\(zhihu.com\)](#)

lidar_undistortion_2d**雷达去畸变模块，测试残留，在实际代码中并未使用到

链接: [LiuLimingCode/lidar_undistortion_2d\(github.com\)](#)

racecar文件夹的内容

名称	修改日期	类型	大小	
cfg	2023/12/20 4:17	文件夹		
launch	2023/12/20 4:17	文件夹		launch文件--关键
map	2023/12/20 4:17	文件夹		map存储
param	2023/12/20 4:17	文件夹		导航规划器参数
rviz	2023/12/20 4:17	文件夹		
src	2023/12/20 4:17	文件夹		存放重要代码的地方
udev	2023/12/20 4:17	文件夹		
CMakeLists.txt	2023/10/27 20:01	文本文档	8 KB	
package.xml	2022/10/9 0:07	XML 源文件	4 KB	
racecar_rviz.sh	2022/10/9 0:07	sh_auto_file	1 KB	
README.md	2022/10/9 0:07	Markdown File	3 KB	
save_map.sh	2023/11/15 19:59	sh_auto_file	1 KB	地图保存脚本，请根据实际路径修改

launch文件夹与src文件夹里的内容最关键

launch文件夹的内容

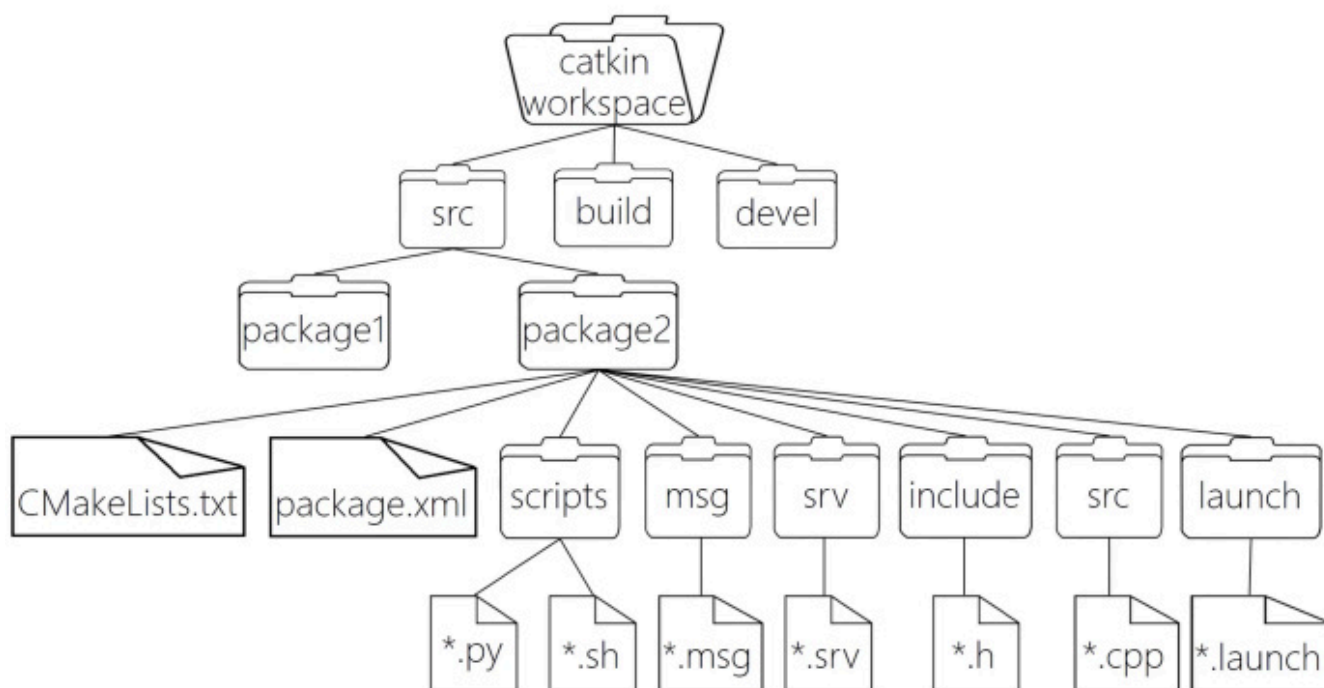
includes	2023/12/20 4:17	文件夹		
amcl_nav.launch	2023/11/4 21:58	LAUNCH 文件	3 KB	
amcl_nav_copy.launch	2023/5/17 21:28	LAUNCH 文件	3 KB	
amcl_nav_dl.launch	2023/10/28 20:30	LAUNCH 文件	3 KB	
amcl_nav_no_control.launch	2023/5/17 21:26	LAUNCH 文件	3 KB	
amcl_nav_teb.launch	2023/12/5 19:32	LAUNCH 文件	3 KB	使用teb的导航
amcl_nav_teb_light.launch	2023/12/4 11:20	LAUNCH 文件	3 KB	
car.launch	2022/10/9 0:08	LAUNCH 文件	1 KB	
lidar_tracking.launch	2023/12/5 19:31	LAUNCH 文件	1 KB	开启雷达循迹的launch，里面有雷达循迹的重要参数
ls01b_lidar.launch	2023/11/1 16:52	LAUNCH 文件	1 KB	
ls01g_lidar.launch	2022/10/9 0:08	LAUNCH 文件	1 KB	
map_adv.launch	2023/11/12 18:05	LAUNCH 文件	1 KB	
map_save.launch	2023/10/25 11:24	LAUNCH 文件	1 KB	地图保存，里面内容请根据实际路径修改
move_base1.launch	2023/11/15 19:12	LAUNCH 文件	2 KB	
Run_car.launch	2023/11/1 21:41	LAUNCH 文件	1 KB	启动小车的传感器
Run_gmapping.launch	2023/11/30 17:23	LAUNCH 文件	3 KB	启动建图
rviz.launch	2023/11/1 20:58	LAUNCH 文件	1 KB	
teleop_joy.launch	2022/10/9 0:08	LAUNCH 文件	1 KB	

racecar/src下的内容

名称	修改日期	类型	大小	
__pycache__	2023/12/20 4:17	文件夹		
lib	2023/12/20 4:17	文件夹		
other	2023/12/20 4:17	文件夹		
parking_py	2023/12/20 4:17	文件夹		
siftparking	2023/12/20 4:17	文件夹		
siftred	2023/12/20 4:17	文件夹		
2023-11-21-15-49-49.bag	2023/11/21 15:50	BAG 文件	56,578 KB	
auto_nav.py	2023/12/5 19:29	Python 源文件	11 KB	一键启动的脚本
auto_nav_test.py	2023/11/19 15:04	Python 源文件	11 KB	
auto_nav2.py	2023/12/4 12:24	Python 源文件	8 KB	
car_controller.cpp	2022/10/9 0:10	C++ 源文件	6 KB	
car_controller.hpp	2022/10/9 0:10	DevCpp.hpp	2 KB	
car_controller_new.cpp	2023/11/29 16:48	C++ 源文件	18 KB	控制器相关的代码
ESC_calibration.py	2022/10/9 0:10	Python 源文件	3 KB	
getpoints.py	2023/5/19 3:49	Python 源文件	2 KB	
goal.csv	2023/10/27 13:52	Microsoft Excel ...	10 KB	
goal_loop.py	2023/12/5 19:44	Python 源文件	6 KB	
mapBlur.py	2023/11/15 20:36	Python 源文件	2 KB	地图滤波，实际并未使用
mapBlur_oncar.py	2023/11/3 17:13	Python 源文件	9 KB	
nav_point.csv	2023/12/5 19:43	Microsoft Excel ...	3 KB	
PID.h	2022/10/9 0:10	C Header 源文件	1 KB	
racecar_joy.py	2022/10/9 0:10	Python 源文件	4 KB	
racecar_teleop.py	2023/11/7 16:43	Python 源文件	6 KB	
stop.py	2023/11/20 21:06	Python 源文件	1 KB	停车脚本，很关键，车在高速运动时人无法抓住，只能通过启动这个来停车
zsh_history	2023/11/3 17:07	文件	93 KB	

复现说明

- 1.最基础的准备工作：按照官方配置好小车
- 2.导航用的算法是TEB算法，具体内容情况技术手册
请根据手册配置好teb模块
- 3.将任意一份代码复制到主目录下，编译
- 下图是ros的文件系统层级



4.编译完成后，source一下工作空间

5.修改下面的路径为自己的真实路径

```
145
146 if __name__ == "__main__":
147     rospy.init_node('main')
148     rviz=Run_launch(["/home/scnu-car/vanguardcar/src/racecar/launch/rviz.launch"])
149     gmapping=Run_launch(["/home/scnu-car/vanguardcar/src/racecar/launch/Run_gmapping.launch"])
150     amcl=Run_launch(["/home/scnu-car/vanguardcar/src/racecar/launch/amcl_nav_teb.launch"])
151     map_save=Run_launch(["/home/scnu-car/vanguardcar/src/racecar/launch/map_save.launch"])
152     map_adv=Run_launch(["/home/scnu-car/vanguardcar/src/racecar/launch/map_adv.launch"])
153     # amcl.start()
154     # rospy.sleep(10)
```

6.打开1个终端，输入 `roslaunch racecar Run_car.launch` 启动小车的传感器

7.在src/racecar/src下打开一个终端，输入 `python3 stop.py`，不要回车，这个是停车的脚本，当小车失控乱跑的时候才回车使用，这里只是做好准备

8.确保小车电调处于关闭状态，在src/racecar/src下再打开一个终端，输入 `python3 auto_nav.py` 回车，即可运行。等待所有东西加载完毕后，打开电调，小车就能跑了

技术手册

国赛版与区域赛版主要区别在雷达循迹算法的优化上，增强了其抗光性能，进而提升了鲁棒性

速度快但建图好的关键

提高ekf融合发布频率

param下的ekf_params.yaml, 适当提高frequency

```
1 frequency: 100
2
3 sensor_timeout: 0.01
4 two_d_mode: true
5 transform_time_offset: 0.0001
6 transform_timeout: 0.025
7 print_diagnostics: false
8 debug: false
```

循迹相关参数的调节:

```
lidar_tracking.launch x
src > racecar > launch > lidar_tracking.launch > ...
1 <?xml version="1.0"?>
2 <launch>
3   <node pkg="lidar_tracking" name="lidar_tracking_nnew" type="lidar_tracking_nnew">
4     <param name="topic" value="laser_control"/>
5     <!-- <param name="topic" value="car/cmd_vel"/> -->
6     <param name="speed" value="1925"/>
7     <param name="kp" value="0.93"/> <!-- 0.8 -->
8     <param name="ki" value="0.0"/>
9     <param name="kd" value="1.5"/>
10    <param name="max_right_dis" value="1.5"/>
11
12    <param name="rate1" value="0.8"/>
13    <param name="rate2" value="2.2"/>
14    <param name="rate3" value="0.75"/>
15  </node>
16
17
18 </launch>
19
```

① rate1~rate3 是从近到远 3 对锥桶的误差计算权重, 循迹在 1925 以下速度的基本不需要再改动了, 速度快了才考虑适当加大 rate2和 rate3。

② max_left_dis 不改动, max_right_dis 代表雷达循迹能扫描到的 右边的锥桶的最大 x 方向距离, 目前 max_right_dis 的值为 1.5, 除非现场确实出现很大的互相影响, 否则不再减小 max_right_dis 了, 如果发现他找不到靠右边的桶可以适当加大一些。

③ **最重要的就是 PID 的 kp 和 kd 参数**，这是最可能要修改的两个参数了，当发现小车转不过弯适当增大 kp 值，因为参数做了归一化，所以小小的 kp 就会有大大的影响，特别是小车能跑只是贴外圈而已，增大 kp 值的时候可以**每次增大 0.05 进行尝试**，相反如果是太贴内圈肯定降低 kp 会好，入弯反应慢或者直道摆动严重对采点影响大，还是建议玄学调调，最后一次调试的时候发现增大 kd 好像对减小抖动帮助不大，反而降低 kd 之后直道稳一些。

④ 有一个参数，就是 middle_angle，目前 1925/1850 版本使用的是 84（每辆车的舵机中值可能有差别），正常情况下这是不该改的，但如果为了让小车在速度快的時候提前拐大弯，因为两个大弯都是左转，所以适当增大 middle_angle 是可以的，虽然不合理，但我们试过了确实有奇效。

导航相关参数的调节

在 amcl_nav_teb.launch 中

```
<node pkg="racecar" type="car_controller_new" respawn="false" name="car_controller">

  <param name="Vcmd" value="1.5" />

  <param name="baseSpeed" value="200"/>
  <param name="baseAngle" value="90.0"/>
  <param name="Angle_gain_p" value="-5.5"/>
  <param name="Angle_gain_d" value="-3.0"/>

  <param name="Lfw" value="1.1"/>           前瞻
  <param name="vp_max_base" value="575"/>   速度
  <param name="vp_min" value="575"/>
  <param name="goalRadius" value="0.5"/>     距离目标点半径多少就视为到达

</node>
```

最关键的代码是 auto_nav.py

细读代码与手册

解耦调试

如果只要调试雷达循迹，只需要将 topic 改为 car/cmd_vel，即注释第四行，解注释第五行

修改后：

```
lidar_tracking.launch
src > racecar > launch > lidar_tracking.launch > launch > node
1  <?xml version="1.0"?>
2  <launch>
3      <node pkg="lidar_tracking" name="lidar_tracking_nnew" type="lidar_tracking_nnew">
4          <!-- <param name="topic" value="laser_control"/> -->
5          <param name="topic" value="car/cmd_vel"/>
6          <param name="speed" value="1925"/>
7          <param name="kp" value="0.93"/> <!-- 0.8 -->
8          <param name="ki" value="0.0"/>
9          <param name="kd" value="1.5"/>
10         <param name="max_right_dis" value="1.5"/>
11
12         <param name="rate1" value="0.8"/>
13         <param name="rate2" value="2.2"/>
14         <param name="rate3" value="0.75"/>
15     </node>
16 </launch>
17
18
19
```

开启小车传感器

```
1 | roslaunch racecar Run_car.launch
```

开启雷达循迹

```
1 | roslaunch racecar lidar_tracking.launch
```

记得准备好 stop.py 随时停车

如果要耦合调试，记得修改回来

经验总结

多调试！

最后，祝取得好成绩！