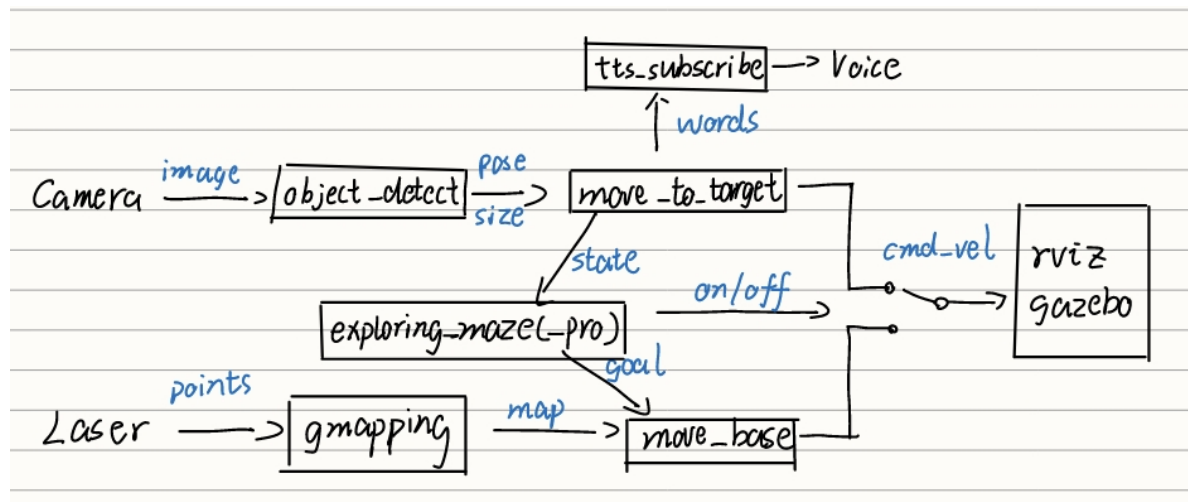


机器人操作系统“迷宫寻宝”作业报告

程序中各模块功能实现



1. 首先需要使用Laser对环境进行感知和建图，通过gmapping算法实现地图的构建以及导航操作
2. 在机器人进行随机或者离散的导航探索过程中，Camera不断的搜索拍摄到的图像信息寻找是否有宝藏
3. 以上导航和视觉跟踪都需要发布线速度和角速度目标来控制机器人运动，此处还涉及导航模块和视觉模块的交互控制
4. 所有的状态信息和内容均可以通过窗口打印和语音输出的形式告知使用者

探索地图功能的简单改进

在基础的地图探索中，通过在地图范围内随机生成目标位置进行随机探索，这样的方法具有较大的不确定性，搜索宝藏的时间难以保证。结合激光雷达可以进行大范围的地图扫描，可以通过在地图中均匀地插入一些目标点，让机器人逐一探索，从而提高搜索效率。exploring_maze_pro改进部分如下：

```
# 初始位置
start_location = Pose(Point(0, 0, 0), Quaternion(0.000, 0.000, 0.709016873598, 0.705191515089))

locations = []

locations.append(Pose(Point(-0.002, 7.663, 0.000), Quaternion(0.000, 0.000, 0.709016873598, 0.705191515089)))
#####请补充更多位置(开始)#####
locations.append(Pose(Point(3., 7.663, 0.000), Quaternion(0.000, 0.000, 0.709016873598, 0.705191515089)))
locations.append(Pose(Point(3., 0.001, 0.000), Quaternion(0.000, 0.000, 0.709016873598, 0.705191515089)))
locations.append(Pose(Point(7., 0.001, 0.000), Quaternion(0.000, 0.000, 0.709016873598, 0.705191515089)))
locations.append(Pose(Point(7., 7.663, 0.000), Quaternion(0.000, 0.000, 0.709016873598, 0.705191515089)))
```

首先大致均匀地给出搜索目标序列，依次存入队列尾部。

```

self.goal = MoveBaseGoal()
self.goal.target_pose.pose = start_location # 初始化
self.goal.target_pose.header.frame_id = 'map'
self.goal.target_pose.header.stamp = rospy.Time.now()

if self.exploring_cmd is STATUS_EXPLORING: # 随机探索模式
    temp_location = locations.pop() # 取出队列末尾元素
    self.goal.target_pose.pose = temp_location # 赋值到目标值
    temp_location.position.x += random.randint(-1,1) # 在目标附近随机离散
    temp_location.position.y += random.randint(-1,1)
    if temp_location.position.x >= 9: # 超出地图边界做相应处理
        temp_location.position.x = 8.5
    elif temp_location.position.x <= 0:
        temp_location.position.x = 0.5
    locations.insert(0,temp_location) # 送到队列前面,形成下一个循环
elif self.exploring_cmd is STATUS_CLOSE_TARGET: # 视觉接管模式
    rospy.sleep(0.1)
elif self.exploring_cmd is STATUS_GO_HOME: # 回到初始点模式
    self.goal.target_pose.pose.position.x = 0
    self.goal.target_pose.pose.position.y = 0

```

随后在循环中取出一个目标点作为目标值，同时在此目标点附近随机生成另一个点，进行边界处理并且插入到队列首部，作为下一组要探索的位置。这个做法能让机器人先对整个地图环境有一个大致的了解，随后再进行一些更加细致的搜索和感知，提高搜索效率。

遇到的问题

在调试程序过程中遇到了语音程序不能运行的问题，经过排查发现是cmake里面相关库的位置没有找到，因此修改语音库动态库路径如下

