

ROS2 KCF 跟随功能使用与讲解

1.功能简介

ROS2 当中的 KCF 跟随功能，主要是通过读取鼠标左键的键值来确认是否框选了需跟随的目标物体，如检测到框选成功则会通过 RGB 相机实时记录跟踪结果并将跟踪结果转化为 ROS 图像话题发布出去，同时根据目标的深度图像信息来对小车和目标的角度及距离进行计算，根据这个计算结果发布 ROS 速度话题来实现小车 KCF 控制，具体效果表现为：KCF 跟随功能启动后，会弹出一个显示小车 RGB 相机图像的窗口，在图像中使用鼠标左键框选好需要跟随的目标物体后小车会开始跟随该物体，使小车与目标物体始终保持一定距离，同时保持目标物体始终处于图像的中心。

2.使用方法

使用 SSH 登录后打开一个终端，并运行 KCF 跟随功能指令

```
ros2 launch wheeltec_robot_kcf wheeltec_robot_kcf.launch.py
```

终端输出如下：

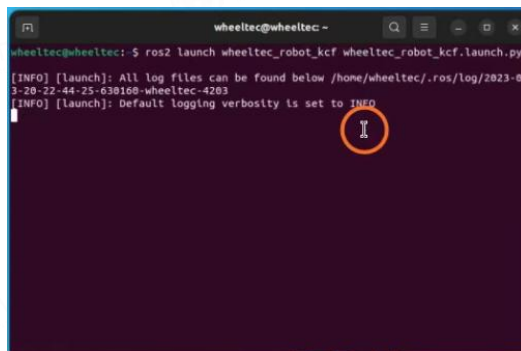


图 1 kcf 功能运行效果

稍等片刻后会弹出一个图像窗口，用鼠标左键在图像中框选需要跟随的目标物体

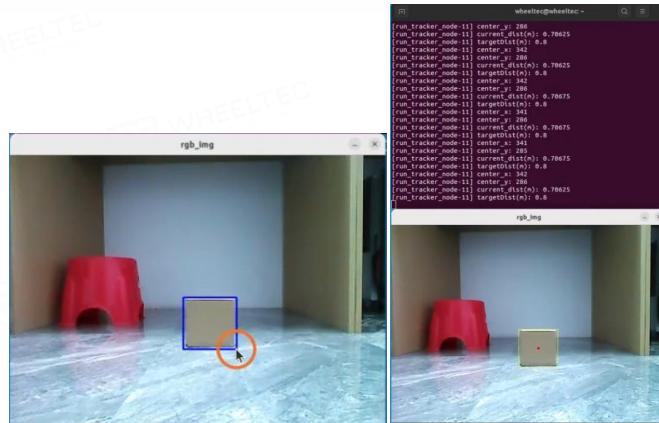


图 2 框选目标物体操作

框选完成后方框会变成黄色，代表着成功开始跟随，此时移动目标物体，发现小车会跟随物体移动，并始终与目标物体保持一定的距离与角度。

3.注意事项

部分主控使用 SSH 远程登录后运行 KCF 跟随功能会出现卡顿的现象，可以使用 VNC 远程登录到小车桌面后在小车端运行 KCF 跟随功能，卡顿问题会好转。

4.功能讲解

KCF 跟随源码存放在工作空间下的 wheeltec_robot_kcf 文件夹，通过 run_tracker.cpp 文件我们可以看到 KCF 跟随的工作原理，图 4-2 将 KCF 跟随的工作原理较直观展现出来。

源码通过创建 run_tracker_node 节点订阅 RGB 和深度图像话题，同时将两个图像话题都转化为 opencv 格式方便后续的处理，通过 opencv 显示 RGB 图像窗口、同时读取鼠标左键键值，获取图像中框选物体的坐标信息，当读取到鼠标释放操作时（代表框选操作结束），会将对目标物体的跟踪结果实时发布，同时根据框选物体也就是目标物体的深度信息来实时计算小车与目标物体的相对距离与角度，根据计算的距离与角度信息发布 ROS 速度话题来控制小车移动以确保小车与目标物体始终保持一定的距离和角度，这样就实现了 KCF 跟随效果。

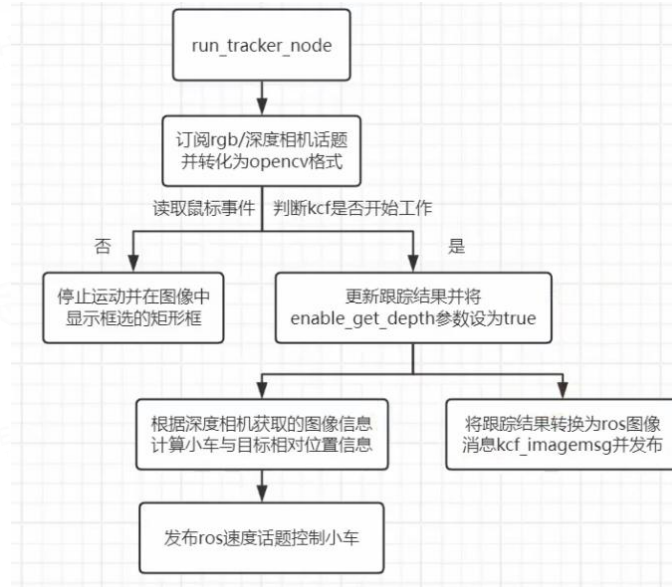


图 3 功能节点示意图

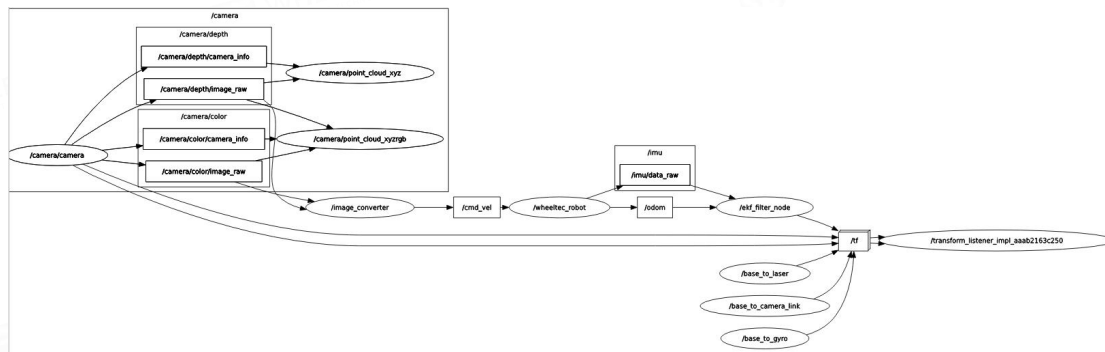


图 4 kcf 跟随功能对应 rqt_graph