

2、颜色跟随

2、颜色跟随

- 2.1、实验说明
- 2.2、实验目标
- 2.3、实验操作
- 2.4、实验效果
- 2.5、实验总结

2.1、实验说明

本实验是属于拓展类实验，需要搭配其他外部设备使用，这里用到了小车底盘和ROS扩展板都不属于K210开发板套件内容，所以本实验效果仅供参考，如果没有对应的设备是不能直接使用本例程代码。

ROS扩展板需要提前烧录固件：ROS-CAR.hex

由于本次使用的电机电压为8.4V，所以ROS扩展板的电池不可以插入12.6V电池，需要插入8.4V电池。

小车电机连接线如下图所示：

电机Motor 1连接左前轮，电机Motor 2连接左后轮，电机Motor 3连接右前轮，电机Motor 4连接右后轮。



K210开发板与ROS扩展板连接的线序如下图所示：

白色连接GND，黄色连接VCC，黑色连接SCL，红色连接SDA。

这里需注意一下，图示中的标识为I2C线序标识，但是K210使用的是串口通讯，由于烧录的ROS-CAR.hex文件已经把这个接口修改为串口信号，所以实际上ROS扩展板上的接口对应关系是：SCL实际上是TX，SDA实际上RX。



2.2、实验目标

本节课主要学习K210开发板搭配小车底盘做视觉巡线的功能。

本次实验的参考代码路径为：06-export\follow_color.py

2.3、实验操作

1. ROS扩展板烧录固件：ROS-CAR.hex
2. 底板电机连接ROS扩展板，按照M1连接左前电机，M2连接左后电机，M3连接右前电机，M4连接右后电机。
3. 请将06-export\library目录下的小车驱动库和PID控制库提前下载到内存卡的根目录上。
4. 打开CanMV IDE将follow_color.py代码下载进入K210开发板。
5. 将K210开发板通过4PIN排线连接到ROS扩展板上。

6. 将小车放入白色背景中，掰动K210开发板支架到合适的角度，打开小车的开关。
7. 等待系统初始化完成后，LCD显示摄像头画面，并且屏幕中间有一个白色的方框，请移动小车，将要识别的颜色填满白色方框，等待白色方框变绿则开始采集颜色，采集完成绿框消失，开始运行程序。

2.4、实验效果

等待系统初始化完成后，LCD显示摄像头画面，并且屏幕中间有一个白色的方框，请将要识别的颜色放到白色方框内，等待白色方框变绿则开始采集颜色，采集完成绿框消失，开始运行程序。

小车会跟着上一步绿色方框识别到的颜色移动。当颜色远离小车时，小车向前运动，当颜色靠近小车时，小车向后运动，当颜色向左/向右时，小车也跟着向左/向右运动，让识别的颜色保持在屏幕的中间位置。

如果中途发现小车经常无法跟随物体，请在白色背景下放置纯色物体来给小车跟随，如果是小车反应过慢或者过快的问题，可调节适当调整PID_x和PID_y的值。

```
follow_color.py  ▾ | × |
1  import sensor, image, time, lcd
2
3  from modules import ybserial
4  from robot_Lib import Robot
5  from simplePID import PID
6
7  PIDx = (30, 1, 3)
8  PIDy = (5, 0, 2)
9  SCALE = 1000.0
10
11  ▾ PID_x = PID(
12      160,
13      PIDx[0] / 1.0 / (SCALE),
14      PIDx[1] / 1.0 / (SCALE),
15      PIDx[2] / 1.0 / (SCALE))
16
17  ▾ PID_y = PID(
18      120,
19      PIDy[0] / 1.0 / (SCALE),
20      PIDy[1] / 1.0 / (SCALE),
21      PIDy[2] / 1.0 / (SCALE))
22
```

2.5、实验总结

小车颜色跟随主要靠的是识别颜色后获取颜色在图像中的位置，然后根据位置的中心点判断与屏幕中心点的偏移量，结合PID算法，从而计算出小车是应该前进后退，或者左转右转等状态，让跟随的颜色保持在屏幕中间位置。