

9、云台追踪人脸

- 9、云台追踪人脸
 - 9.1、实验说明
 - 9.2、实验目标
 - 9.3、实验操作
 - 9.4、实验效果
 - 9.5、实验总结

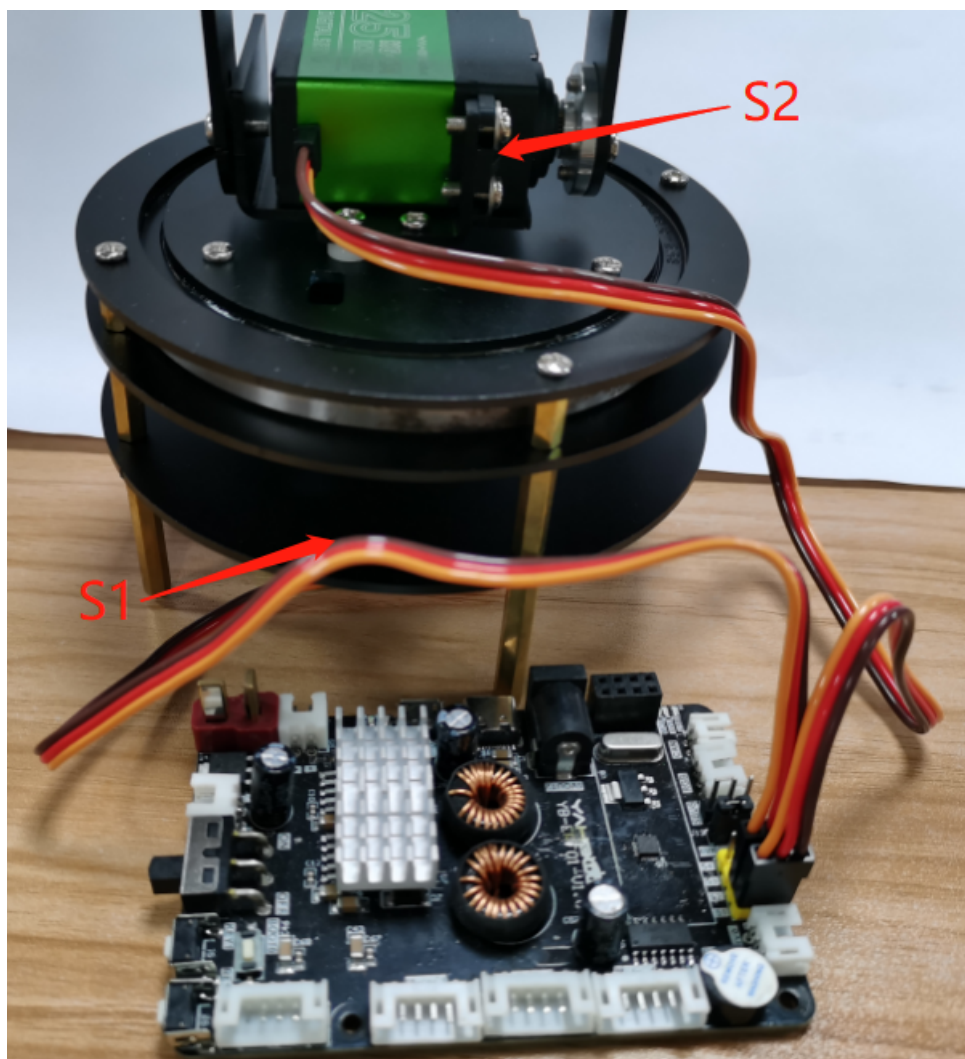
9.1、实验说明

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

云台舵机连接到ROS扩展板上对应的接口为：

S1连接X轴舵机，S2连接Y轴舵机。黄色为信号，红色为VCC，黑色为GND。



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。



9.2、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\tracking_face.py

9.3、实验操作

1. ROS扩展板烧录固件：ROS-CAR.hex
2. 将RGB灯条插入ROS扩展板的RGB灯接口。
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打开tracking_face.py代码并下载到K210开发板上。
5. 将K210开发板通过4PIN排线连接到ROS扩展板上。
6. 将云台放在白色或者黑色背景中，然后打开ROS扩展板的电源。
7. 人脸进入K210开发板摄像头的采集范围，K210开发板会框出人脸，并且云台会追踪人脸。

9.4、实验效果

等待系统初始化完成后，K210开发板摄像头会实时检测是否画面中有人脸，如果画面中有人脸，修改云台舵机角度尽量让人脸保持在屏幕画面中间，摄像头追踪人脸转动。

如果追踪反应太快或者太慢，可以适当修改程序中的PID参数。

```
PIDx = (50, 0, 15)
PIDy = (50, 0, 10)
SCALE = 1000.0

✓ PID_x = PID(
    160,
    PIDx[0] / 1.0 / (SCALE),
    PIDx[1] / 1.0 / (SCALE),
    PIDx[2] / 1.0 / (SCALE))

✓ PID_y = PID(
    120,
    PIDy[0] / 1.0 / (SCALE),
    PIDy[1] / 1.0 / (SCALE),
    PIDy[2] / 1.0 / (SCALE))
```

9.5、实验总结

云台追踪人脸玩法是基于人脸检测功能开发，将K210开发板摄像头检测到人脸的位置坐标，通过PID算法计算出云台需要运动的位置，从而使云台能够追踪摄像头前的人脸。由于画面帧率与识别限制，人脸活动不能太快速，否则云台可能会跟不上反应。