

一、需要添加的代码

```
#include <boost/thread.hpp>    //ros 的线程库（可选择使用）
//ros 中线程的使用: https://www.ncnynl.com/archives/201702/1319.html

int main(int argc, char **argv)
{
    ros::init(argc,argv,"ColorMove");//初始化 ROS 节点
    ros::NodeHandle n;
    cmd_pub = n.advertise<geometry_msgs::Twist>("cmd_vel", 5);//定义速度发布者

    //开一个线程，一直读摄像头数据
    boost::thread analyseColor_thread(&analyseColor); //自己编写代码识别颜色

    //开一个线程，一直发速度，有颜色就发对应前后左右移动小车，没颜色就不发
    boost::thread sendCmd_thread(&sendCmd);//&sendCmd 发送速度的方法

    analyseColor_thread.join();//启动识别颜色线程
    sendCmd_thread.join();//启动发送速度线程
    return 0;
}

void sendCmd() {
    ros::Rate loop_rate(10);//发布消息频率，1 秒 10 次
    while (1) {
        if(redN < 80) {        //redN 图像程序输出的值
            twist.linear.x = 0;//线速度
            twist.linear.y = 0;
            twist.linear.z = 0;
            twist.angular.x = 0;
            twist.angular.y = 0;
            twist.angular.z = 0.4;//角速度
            ROS_WARN("*****cmd");//调试使用，打印输出（可以看程序执行到哪里）
            cmd_pub.publish(twist); //发布消息
        }
        loop_rate.sleep(); //根据频率进行睡眠
    }
}

void analyseColor() {
    //识别图像， 改变 redN（图像程序输出的值）的值
}
```

二、启动小车方法

- (1) 打开一个终端，先启动小车的驱动程序 `roslaunch dashgo_driver driver_imu.launch`
- (2) 再打开一个终端，启动上述识别颜色程序