十二届光电追逐组程序说明

我们组使用的单片机是**K60FX**型号，使用的程序库是**山外K60参考代码V5.3**

这篇说明的主要内容：

一、单片机实现功能

二、整体程序结构

三、图像处理程序讲解

四、总结

一、单片机实现功能

1.输入：

（1）采集输入**总钻风**摄像头灰度图像信息

（2）采集输入**编码器**反馈的速度信息

（3）采集输入**鸳鸯超声波**模块采集的距离信息

（4）采集输入**NRF2401**模块接收的通信信息

（5）采集输入**按键、拨码开关**的电平信号

（6）采集输入**MPU6050**陀螺仪的信息

2.输出：

（1）输出一路**PWM波**控制舵机转向

（2）输出两路**PWM波**控制H桥电路工作

（3）输出**LCD**显示信息

（4）输出**NRF2401**模块发送的信息

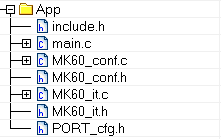
（5）输出控制**鸳鸯超声波**工作的电平信号

（6）输出控制**蜂鸣器**工作的电平信号

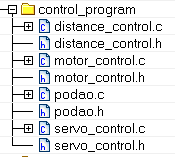
二、整体程序结构

使用过**山外K60参考代码**的同学应该都知道，山外库内自己编写的程序都存放在**APP**文件夹内，其他文件夹内为山外库自带的程序。下面讲一下我们组程序APP文件夹下的内容。

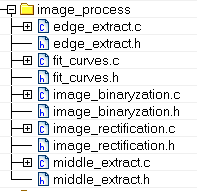
1.下图中，APP文件夹下的程序是山外库自带的



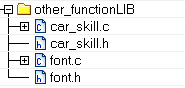
2.app\control\_program文件夹内为车模方向控制、速度控制、双车距离控制的程序



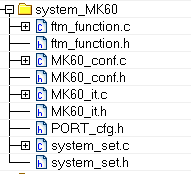
3.app\image\_process文件夹内为边沿提取、中线提取、图像二值化等，图像处理相关的程序



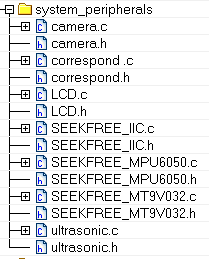
4.app\other\_functionLIB文件夹内为双车控制程序



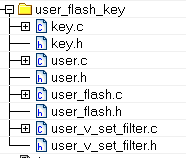
5.app\system\_MK60文件夹内为FTM功能配置、引脚配置、中断配置等，底层功能配置



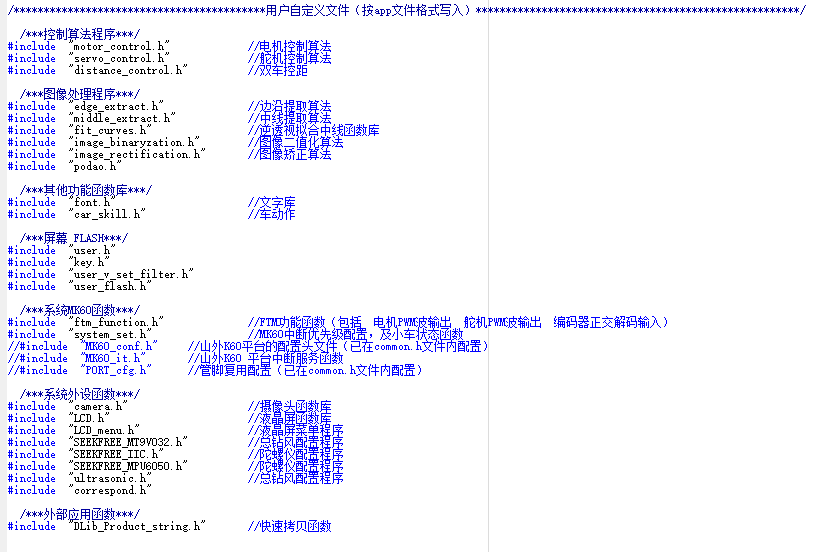
6.app\system\_peripherals文件夹内为摄像头采集、陀螺仪采集、超声波采集、LCD显示等，模块功能的程序



7.app\user\_flash\_key文件夹内为按键、flash存储、屏幕显示功能程序



具体每个.c文件的功能，可以参考include.h程序中的注释



三、图像处理程序讲解

图像处理的思路可以参考**示例技术报告**中**第四章**的内容

1.对原始图像进行桶形失真校正和像素抽取，实际处理的图像分辨率为60（高）\*160（宽）。

使用image\_rectification.c文件中的void barrel\_rectification()函数。（图像桶形失真、透视失真，可以百度了解）

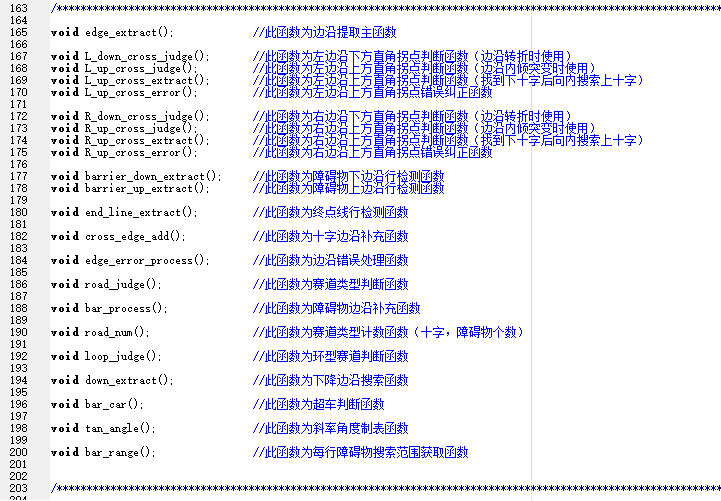
2.对图像进行二值化处理。

使用image\_binaryzation.c文件中的void img\_binary()函数。（原理为otsu法又称大津法，可以百度了解）

3.对二值化后的图像进行边沿提取。

使用edge\_extract.c文件中的void edge\_extract()函数。

edge\_extract.c文件中其他函数都用于赛道的判断，功能可以参考注释。



简单说明程序内容

void edge\_extract()

{

对变量赋初值；

针对某些特定情况，设置一些标志位状态；

For()对图像每一行进行边沿提取，从最底行开始搜索；

{

针对某些特定情况，设置一些标志位状态；

下面搜索左边沿；

If(是否搜索左边沿)

{

分类搜索：

1. 丢边时；
2. 边沿内倾时；
3. 边沿外倾时；
4. 处于十字时；

搜索过程中，判断十字；

搜索过程中，判断弯道和是否停止搜索；

}

下面搜索右边沿；

If()

{

内容和左边沿对称；

}

对边沿搜索出错的处理；

对一些特殊赛道的识别；

}

对一些特殊赛道的识别和处理；（十字、环道、障碍物、弯道等）

}

4.拟合图像中线。

使用middle\_extract.c文件中的void midline\_extract()函数。



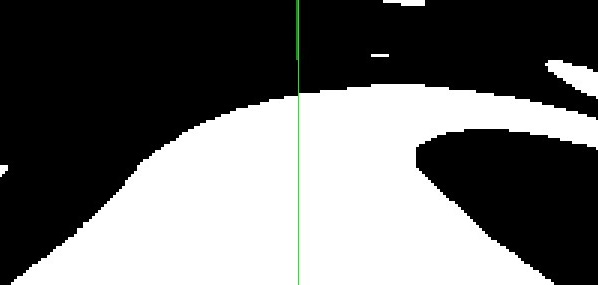
逆透视变换和反逆透视变换在边沿提取和中线拟合中大量使用，可以百度了解，也可以参考示例技术报告。

简单说明思路：

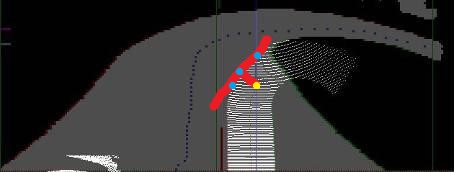
左右边沿都有的情况，左右相加除以二。

只有单边有的情况：

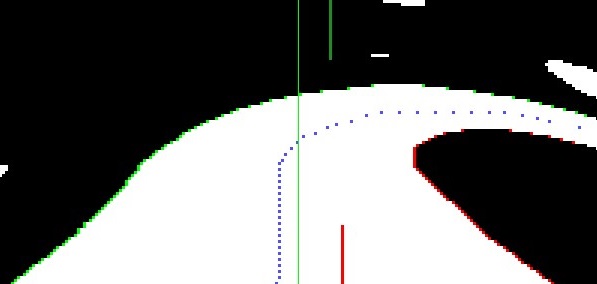
(1)对三个边沿点进行逆透视变换



(2)三点最小二乘法拟合出斜率，再从中间点向垂直方向平移1/2赛道宽度获得中线点



(3)通过反逆透视获得图像的中线点



5.获得中线后，得到某个范围内中线和图像中心的偏差值，用于控制车模方向。

四、总结

1.我们的程序是根据我们的情况写的，直接用一定会有很多问题。建议自己架构程序，有问题进行参考，理解后搬过去。

2.图像处理中图像的桶形失真，逆透视等原理比较麻烦，使用消耗时间比较长。请自行考虑使用。

3.二值化算发中，使用otsu法效果不一定比定值好，请具体情况具体分析。建议多学习摄像头图像采集相关寄存器的配置。

4.鼓励创新，多尝试新的控制方法。我们组的控制方法并不一定好用。

5.当开始使用编码器闭环控速后，不考虑速度的情况下，变速运行（主要是弯道减速）的稳定性（也就是比赛成功率）会比匀速高很多，但是依赖赛道识别和赛道分类。