



Q792499178

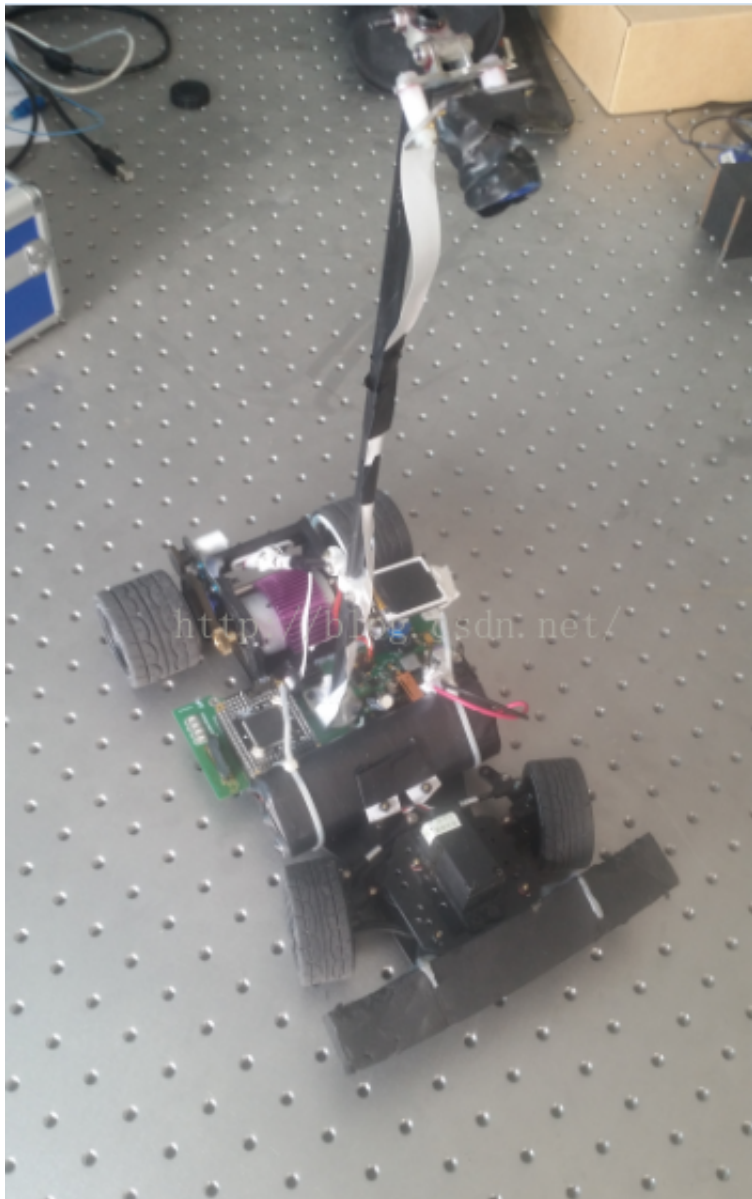
关注

浅谈全国大学生智能车竞赛-摄像头组图像处理及控制算法

原创 2016年11月08日 23:05:38 阅读 14276

由于是第一次写技术博客，先做一下自我介绍。本人本科专业是光电信息科学与工程(工)，就读于长春理工大学光电工程学院，研究生为华中科技大学光学工程专业。本科期间参加参加过各类比赛，其中印象最深的还是智能车竞赛。参加了十届和十一届的智能车竞赛，当然结果是都没有出校赛这让我很尴尬。做了两届的摄像头，第一年跑的速度是1.8m/s第二年跑到了3m/s以上。虽然最后有一些可惜但是我希望把我自己的一些感想和遇到的问题给大家介绍一下，分享一下我的经验，希望能够对学弟学妹有一些帮助。

步入主题，智能车竞赛摄像头组主要包括几个部分，我就以采集图像；赛道模式识别；控制算法；PID算法；机械结构调整等几个方面为大家介绍我的整体思路。



首先我先说一下我第二年用的是B车具体车型自己可以百度一下吧，在这里我简单介绍一下，它用的是一个大电机，速度个人认为是其他车无法比拟的，然后舵机用的是SD-5舵机，是一个数字舵机它的响应频率很快，可以满足我用很高的频率去控制转角，这样也对于控制方面提供了方便。

接下来谈谈如何调整机械部分，首先摄像头的位置，个人把摄像头放在了整个车的重心位置，也就是整个车的中间。舵机采用卧式结构，推荐采用卧式或立式，舵机与摄像头之间放置的电池，为什么将电池放在这是因为为了让整个车的重心靠近中间，摄像头我直接开到了电路板上，将电路板固定在底盘上，整个主控上面包含了所有的模块，节省了空间降低了重心。整车拼起来后最高的位置是舵机和轮子，个人喷了黑色喷漆感觉很帅(这个是看个人了)。前轮调整的时候很多人很迷茫在这我给大家说一下，前轮内八，让最小的转弯半径越大越好，可以得到很好地转弯效果，又因为我

的电池是靠近舵机的所以增加了转弯时候的侧滑力量，内八的程度是多少呢就是当打到最大角度的时候轮子稍微给点离心力就可以所有的面都着地，这样就是最好的状态，还有就是舵机的拉杆，要V字型，为什么呢，就是为了当摆角最大的时候舵机给的推力是最大的，当摆角小的时候舵机反应不是特别灵敏(吉林大学车辆工程的一位车友告诉的方法)，这样调整完前轮后就是后轮的调整，调整到什么状态呢，当电机给最大占空比的时候只要按住车瞬间电机就能停住，这样就可以让以后编码器测出来的数据不会有空转的情况(自己慢慢测试就好没有什么很好地方法)，安装编码器的时候编码器齿轮与后车轮的齿轮啮合要怎么检测呢，就是当你转动编码器齿轮时候看不出空回，如果有空回就要继续调整直道看不出空回为止。到此为止只有一个事情要解决了，那就是B车底盘的问题，这个东西的话大家很多人是不知道的，就是买车模的时候它会给两个小块通常是黑色的(我当时以为这个东西是鸡肋，后来才知道我太天真了。)，它是用来垫前轮的底盘的(具体名词不知道想了解可以百度)，然后后轮如果拆下来可以看见轮轴与底盘连接那个位置有个类似椭圆的一个件把它倒过来装上你会发现，咦？？我的底盘低了好多。对就是这样你的底盘调整的更低更有赛车的样子了。至此所有的机械结构基本调整完毕。特别提醒，前后底盘的连接一定要在底盘的上部连接否则你会很不开心，会遇到各种问题，尽量做到的就是用最少的料让前后的底盘连接的最稳定。在此不多做介绍了。

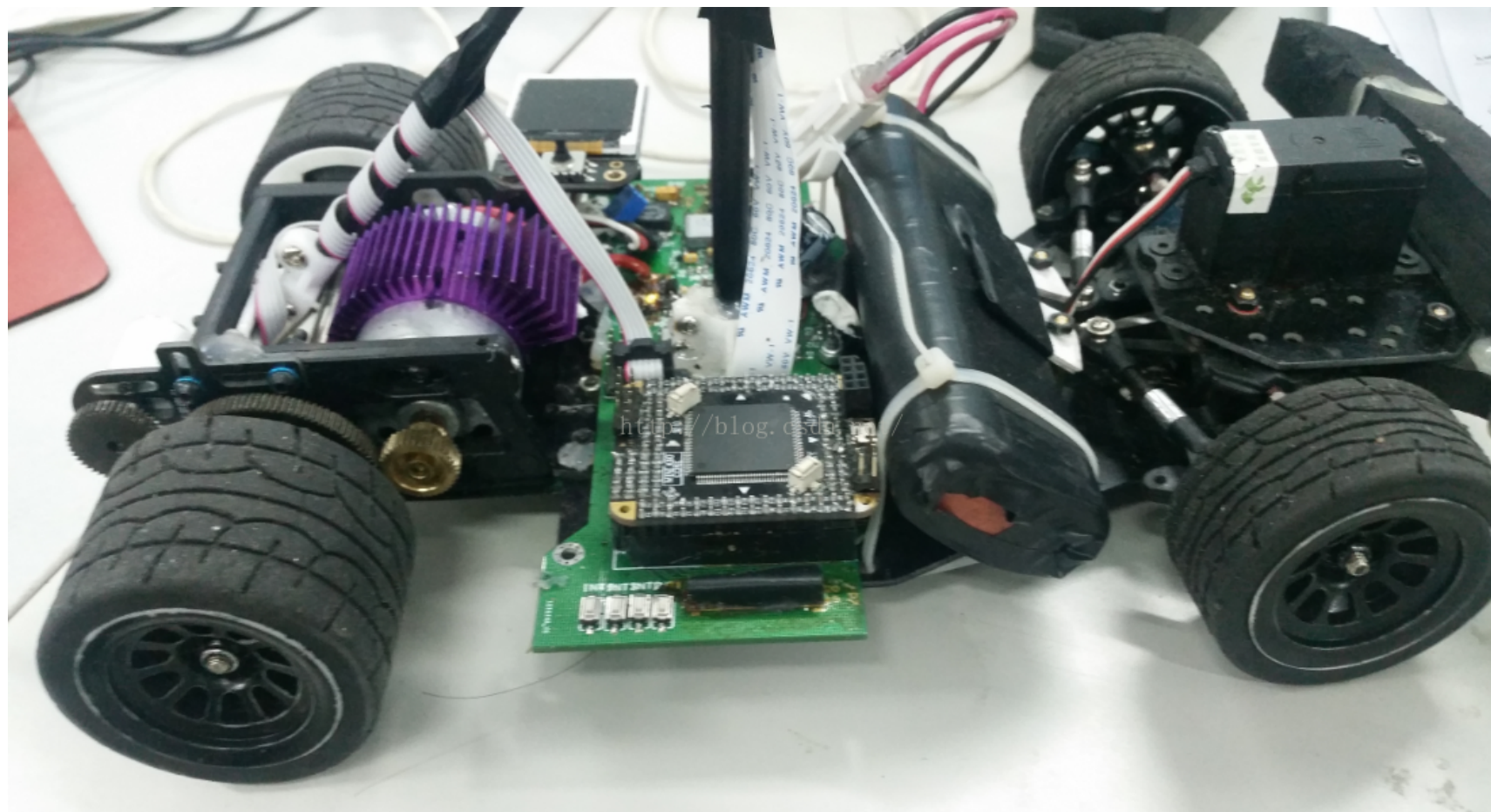
以下这种固定方法是不可取的。



舵机拉杆效果图



整车的结构效果图

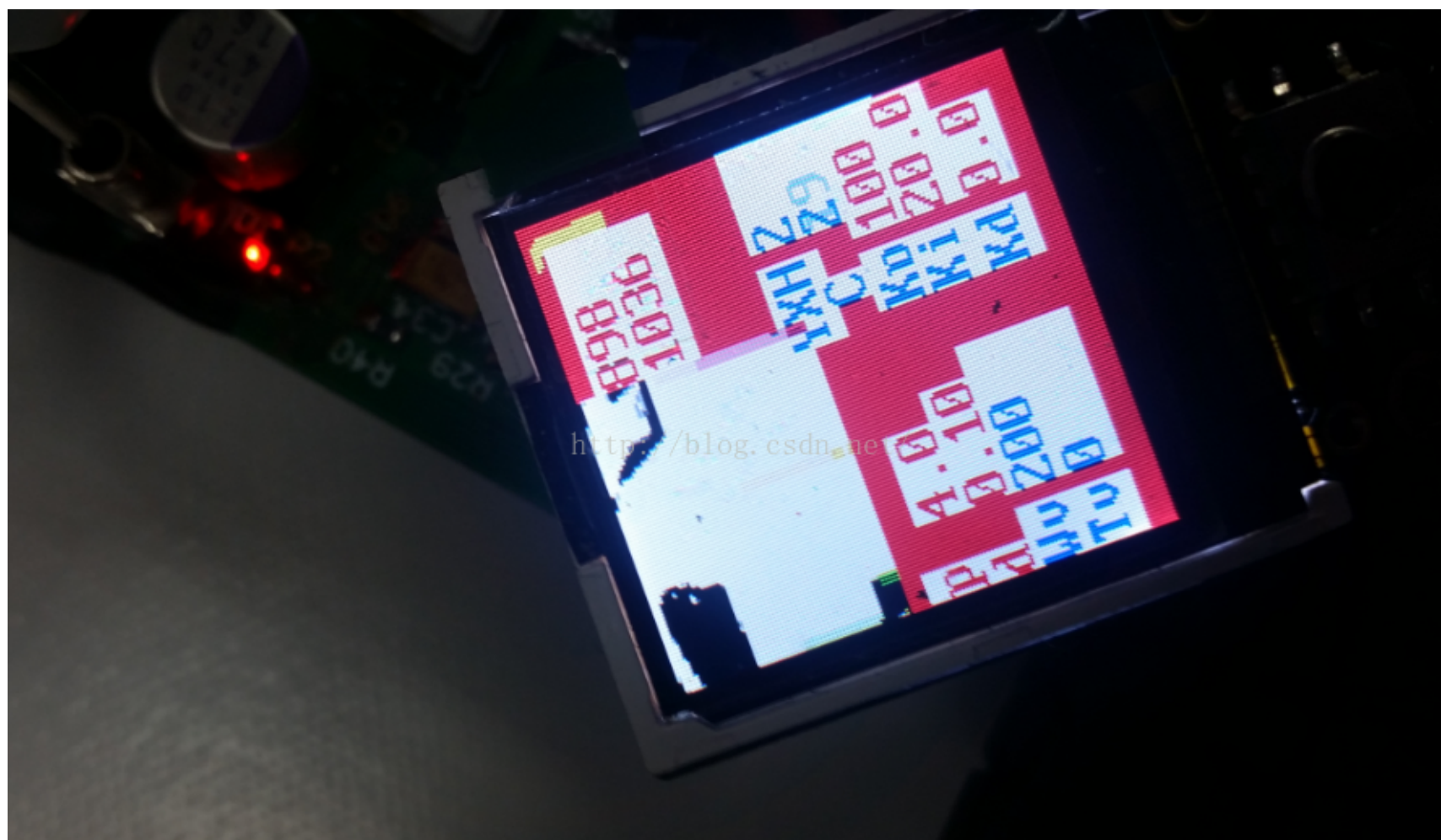


接下来就应该说说软件部分了，第一个就是摄像头图像采集问题，个人用过OV7620，蓝宙捕食者3代，山外鹰眼硬件二值化(不要问我为什么做了两次用了三种摄像头)。这个图像采集的问题呢，我觉得很值得一提，我首先介绍一下三种摄像头吧，7620这款摄像头是最经典的一种摄像头，30帧每秒COMS传感器，个人认为用这个摄像头做到3m/s的大神真的是太牛逼了，这个摄像头如果想要用首先你必须有很强的图像处理能力，因为它采集速度慢，图像质量很渣，个人用它跑到的最大速度是2.2m/s。再就是蓝宙捕食者3代，这个不吹不黑，个人感觉是最渣最渣的一款摄像头，会出现各种各样的问题，而且你不知道该怎么去做，不推荐使用。接下来就是重头戏山外鹰眼摄像头，我采用的是60*80像素，采集速度是148帧每秒，采集速度都快比线性CCD快了。它的噪声如何呢，个人测试过没有滤波算法跑到了3m/s的速度，不解释。这个摄像头首先个人觉得需要一个比较大的广角镜头，其次就是前瞻的远处1.2M就足够了，当然也可以用1.4m个人喜好问题和算法问题。这个摄像头属于动态的阈值，阈值可以通过一个参数进行修改，这个自己多看看山外鹰眼摄像头的介绍就知道了。然后还有一个比较头疼的问题就是不能用山外直接给的摄像头采集程序否则达不到148帧/s这

部分只能靠自己去研究时序然后改正了，思路呢就是在初始化的时候直接初始化所有的中断源，然后用DMA一只采集可以实现理论上的图像处理与图像采集的并行执行。这部分是很重要的当你的采集速度达到148帧/s后车速达到3m/s你每个图像冲出去的距离也不过2.03cm，这样相对于7620就有了一个明显的优势，你可以提高你的极限速度，如果7620跑3m/s你可以跑到更高。

赛道模式识别，这部分是很多人最头疼的部分，我也走过很多的弯路，在处理十字的时候自己看着图像看了一个月的时间最后才解决的，还有就是弯道如何识别，什么入弯什么出弯都是什么情况，本人介绍的方法都是按照我以上介绍的配置才适用的，我问过很多司机师傅问他们怎么开车，他们给的回答都很模糊，那么我们是怎么看的呢，我的理解就是我们看的实际上就可以理解为直线和弯道的分量，当前的图像中直线的分量是多少，只要我能够提取出来这个分量我就能得到很好地一个模糊的数值去控制当前的电机转速(个人认为赛道类型就是用来控制电机转速的)，那我是怎么做到的识别出来这个分量的呢？我应用了三种方式，第一个就是采用《两点算法求智能车赛道曲率》-蒋旭 吴涛 论文中的计算曲率算法计算了赛道前部和后部的曲率，更信任前部分的曲率，然后加上有效行数据，有效行数据与赛道类型建立一个三次函数的映射关系求出其所占的权重，这样三个算法算出来的数值所占的权重不同最终就能得到一个当前赛道直线分量的权重，得到这个权重后就能通过模糊的方式控制电机了。个人采用的是三次函数映射，在过弯的时候能够达到很好地效果，不会出现不连续的现象，这样就能有一个很好的系统响应。具体算法需要自己去写了，

这是屏幕显示的部分，当时在十字问题卡主的时候这个时候提取出来了直线的分量。



在此我贴一部分代码

//找到A B C三个点 对应于 《两点算法求智能车赛道曲率》-蒋旭 吴涛 论文中的 图1

```
Down_A_hang=left_line[0][0];//A点所在行数
```

```
Down_A_lie=left_line[1][0];//A点所在列数
```

```
Down_C_hang=left_line[0][(60-youxiaohang)/2];//C点所在行数
```

Down_C_lie=left_line[1][(60-youxiaohang)/2];//C点所在列数

L_AB=Down_C_hang-Down_A_hang;//求出 L_AB

```
L_BC=abs(Down_C_lie-Down_A_lie);//求出 L_BC
```

$$R_{qulv} = L_{AB} * L_{AB} / (2.0 * L_{BC}) + L_{BC} / 2.0; // \text{计算出当前赛道的曲率半径}$$

////

```
Down_A1_hang=left_line[0][1];//A点所在行数
```

```
Down_A1_lie=left_line[1][1];//A点所在列数
```

Down_C1_hang=left_line[0][((60-youxiaohang)/2)/2+1];//C点所在行数

Down_C1_lie=left_line[1][((60-youxiaohang)/2)/2+1];//C点所在列数

L_AB=Down_C_hang-Down_A_hang;//求出 L_AB

L_BC=abs(Down_C_lie-Down_A_lie);//求出 L_BC

R_qulv=L_AB*L_AB/(2.0*L_BC)+L_BC/2.0+R_qulv;//计算出当前赛道的曲率半径

////////

Down_A2_hang=left_line[0][2];//A点所在行数

Down_A2_lie=left_line[1][2];//A点所在列数

Down_C2_hang=left_line[0][((60-youxiaohang)/2)/2+2];//C点所在行数

Down_C2_lie=left_line[1][((60-youxiaohang)/2)/2+2];//C点所在列数

L_AB=Down_C_hang-Down_A_hang;//求出 L_AB

L_BC=abs(Down_C_lie-Down_A_lie);//求出 L_BC

R_qulv=L_AB*L_AB/(2.0*L_BC)+L_BC/2.0+R_qulv;//计算出当前赛道的曲率半径

R_qulv=R_qulv/3.0;//三个点取平均

这部分是我代码中的一部分，当然需要你们自己去做后面的工作，如果有兴趣欢迎联系qq：792499178.

应用此算法后在实际实践中将车子的速度从2.2达到了2.6m/s这时候又出现了一个新的问题就是舵机出现了抖动(抖动情况可以去看QQ空间上面写的那个号网址

http://mobile.qzone.qq.com/l?g=1336&appid=311&subtype=0&blog_photo=0&ciphertext=28AE2E611D3234CC4EED2BE09951BC60&uw=792499178&g_f=2000000393
)以上是赛道类型识别模糊算法。

接下来就是说说控制算法了，控制算法的话我跟其他人应该没什么太大的区别舵机PD，电机PID。

首先讲讲PID吧，PID的话就是P-比例，I-积分，D-微分，三个部分组成，这个算法的提出是在很久

以前了，具体是谁提出的自行百度吧。在这里首先说说我电机闭环采用的是增量式PID算法，代码

呢在这里给大家贴出来

```
s8 dspeed;

u32 speed_set;

s16 speed_set_base;

s16 Speed_Error=0;

s16 LastSpeed_Error=0;

s16 PrevSpeed_Error=0;

s32 SpeedOut=500000; //电机PWM

s32 LastSpeedOut=500000;

s32 SpeedOut1=0;

extern u8 Start_line;

extern u8 roadflag;

s32 car_speed=0;

extern u8 dispay_on;

void car_speed_control(s16 car_speed_set,u32 Moto_P,u32 Moto_I,u32 Moto_D)

{

    car_speed=LPTMR0_CNR;

    lptmr_counter_clean();

    s16 Speed_ErrorMax=50;
```

```
s16 Speed_ErrorMin=-50;

Speed_Error=car_speed_set-car_speed;

if (Speed_Error>Speed_ErrorMax)Speed_Error=Speed_ErrorMax;

else if (Speed_Error<Speed_ErrorMin)Speed_Error=Speed_ErrorMin;

        SpeedOut=LastSpeedOut+        Moto_P*(Speed_Error-LastSpeed_Error)+
Moto_I*Speed_Error+Moto_D*(Speed_Error+PrevSpeed_Error-2*LastSpeed_Error);

    if(SpeedOut>dianji_PWM_MAX)

{

    SpeedOut=dianji_PWM_MAX;

}

else if(SpeedOut<dianji_PWM_MIN)

{

    SpeedOut=dianji_PWM_MIN;

}


if (SpeedOut>0)

{SpeedOut1=SpeedOut;

    FTM_PWM_Duty(FTM0,CH2,0);

    FTM_PWM_Duty(FTM0,CH0,SpeedOut1);

}

else
```

```
{

    SpeedOut1=-SpeedOut;

    FTM_PWM_Duty(FTM0,CH2,SpeedOut1);

    FTM_PWM_Duty(FTM0,CH0,0);

}

PrevSpeed_Error = LastSpeed_Error;

LastSpeedOut=SpeedOut;

LastSpeed_Error=Speed_Error;

}
```

这就是我用的增量式PID算法，为什么应用增量式PID算法呢，其实我也不是很明白就是感觉它好吧（个人是光电工程专业的我属于不务正业的那伙所以还请专业人士勿喷），当然也不是完全的没有根据，大家可以百度一下位置式PID和增量式PID两种算法的区别，很容易比较出来哪个有优势。在接下来就是在增量式PID的基础上加了一个棒棒算法，这个实际上是我没有调好的，个人能力问题对PID算法没有理解到很好所以加了棒棒算法后整个车加速的速度很快但是声音特别大(但是个人听起来感觉很爽，如果调好了声音是很小的)，棒棒算法是什么呢就是当你编码器返回的速度与你预期速度差距很大的时候直接满占空比输出，这样可以让你的车很快的达到目标速度(这就是对应于之前我那个机械结构的设计，在后轮调整中有提到)，测试后可以在参数一般的情况下2m的距离内加速到10ms，250个脉冲的速度(采用的512线的迷你编码器)，这个根据车模不同有不同的数值，这里的数据仅供参考。接下来就介绍一下舵机PD的参数设定了，这个参数难倒了智能车的大部分人在这里我就透露我的一些经验吧，首先我采用的是P三次函数算法，自变量是有效行，因变量是P的值，当然更建议的是自己设定P值，首先找到一个效果比较好的函数映射曲线后在用matlab进行拟合取出来相应的数值后用数组的形式导入到程序中，在根据调试的情况具体情况具体分析后在微调数组中

的P值，然后D值的选取呢，当你的车速度没有超过2.6m/s之前个人认为没有必要用这个参数，为什么这么说呢，因为我在2.2m/s的时候完全不会出现抖动的情况，当速度提升后在直线时候舵机抖动厉害，这个时候就需要D的值了，这个D值作用是什么呢？它的作用就是放大偏差，有人会说放大偏差干什么，个人的理解是变相的一种闭环，当速度大的时候惯性比较大拐弯不是很容易，当你需要拐的角度越大需要的力就越大，这样就可以体现出来D值的重要性，通常D值根据拐弯的大小即赛道类型而变化个人采用的是与P值成一定的正比例关系。采用这个方式后车子的速度提升到了3m/s，从2.2m/s到3m/s一共用了2天时间。

下面我留下我的舵机P算法

```
float    Kp    =    (CAMERA_H    -    info->effectiveLine)*(CAMERA_H    -    info->effectiveLine)/gServoPid.coefficient+gServoPid.p;

outPWM   =   (int32_t)(Kp*info->error   +   gServoPid.d*(info->error   -   gServoPid.error1))   +

gCarParam.servoCenter;

gServoPid.error1 = info->error;
```

具体参数自己慢慢调。有兴趣的可以联系上面的qq,或联系BaiYH1994@163.com.

个人认为智能车摄像头组大体上我涉及的就只有这些了，还有一些细节问题我没有提过，这些就需要自己去创新了，目标只有一个，但是过程我们可以有上百上千种，以上是我的方法，如果有错误的地方还请各位大神指正，也希望能够帮助新手了解一下智能车的一些知识。

白银浩

🔔 举报



Q792499178

访问量 4万+ 原创 26

[博主更多文章>](#)

浅谈智能车摄像头组斜入十字处理方法

BaiYH1994 3767次阅读 2017-03-17 17:28:49

飞思卡尔之摄像头图像处理——初始化

nanhaiyuhai 326次阅读 2018-03-09 13:15:27

智能车摄像头组 障碍处理方法

BaiYH1994 2805次阅读 2016-12-10 22:24:14

恩智浦杯（飞思卡尔）全国大学生智能车竞赛摄像头简单的图像失真矫正技术原理与实现（透视变换）

m0_37454852 3481次阅读 2017-09-10 15:20:50

飞思卡尔智能车之摄像头使用篇

haiking1 5598次阅读 2015-08-30 11:07:18

智能小车制作过程全纪录： 六、软件平台--- 摄像头云台控制

mwlwlm 1833次阅读 2016-12-30 10:49:22

智能小车二十 《摄像头和路由器装上小车》

koolfret 334次阅读 2017-08-26 16:11:59

更多相关文章