

◆◆◆◆◆ 有关摄像头组的材料 ◆◆◆◆◆

图像采集与路径识别

要实现一个完整的基于摄像头的智能小车，第一步要做的就是将摄像头输出的模拟信号通过 DSP 的 A/D 转换采集到 DSP 中，然后对采集到的原始图像数据进行处理，以获取赛道中央的黑线在图像坐标系中的位置。接着，就要利用处理得到的图像信息对智能小车进行控制。

所以，可以说对于基于摄像头的智能小车，图像采集是至关重要的。只有做好了图像采集，才谈得上“智能”，否则小车只能是个“瞎子”

摄像头。

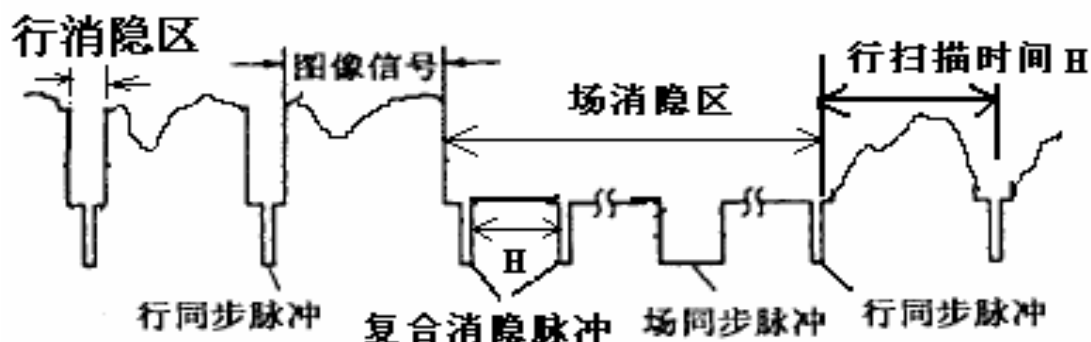
要做图像采集，那么首先要选择好摄像头。

摄像头分黑白和彩色两种，由于赛道是“白底黑线”，所以为达到寻线目的，只需提取画面的灰度信息，而不必提取其色彩信息，所以我们选择采用黑白摄像头。较使用同等分辨率的彩色摄像头而言，这样可减少单片机采样的负担。

此外，摄像头根据成像原理，分为 CCD 和 CMOS 两种，CCD 成像效果好，CMOS 更省电，可以根据自己队的侧重点进行选择。

摄像头的工作原理是：按一定的分辨率以隔行扫描的方式采集图像上的点，当扫描到某点时，就通过图像传感芯片将该点处图像的灰度转换成与灰度一一对应的电压值，然后将此电压值通过视频信号端输出。

摄像头连续地扫描图像上的一行，则输出就是一段连续的电压信号，该电压信号的高低起伏反映了该行图像的灰度变化。当扫描完一行，视频信号端就输出一个低于最低视频信号电压的电平（如 0.3V），并保持一段时间。这样相当于，紧接着每行图像信号之后会有一个电压“凹槽”，此“凹槽”叫做行同步脉冲，它是扫描换行的标志。然后，跳过一行后（因为摄像头是隔行扫描的），开始扫描新的一行，如此下去，直到扫描完该场的视频信号，接着又会出现一段场消隐区。该区中有若干个复合消隐脉冲，其中有个远宽于（即持续时间长于）其它的消隐脉冲，称为场同步脉冲，它是扫描换场的标志



摄像头每秒扫描 25 幅图像，每幅又分奇、偶两场，先奇场后偶场，故每秒扫描 50 场图

像。奇场时只扫描图像中的奇数行，偶场时则只扫描偶数行。

由于奇场和偶场在空间上相差非常小，所以可以忽略奇偶场差别。

也就是说，每秒有 50 副图像产生，每 20ms 就有一副完整的图像产生。（所以基于 CCD 的小车控制周期即为 20ms）

摄像头指标

摄像头有两个重要的指标：有效像素和分辨率。分辨率实际上就是每场行同步脉冲数，这是因为行同步脉冲数越多，则对每场图像扫描的行数也越多。事实上，分辨率反映的是摄像头的纵向分辨能力。有效像素常写成两数相乘的形式，如“320x240”，其中前一个数值表示单行视频信号的精细程度，即行分辨能力；后一个数值为分辨率，因而有效像素=行分辨能力×分辨率。

上届摄像头分析

我们用的是以 1/3 Omni Vision CMOS 为传感芯片的摄像头，它扫描的每场中有 320 行信号，其中第 23 行到 310 行是视频信号，第 311 行到下一场的第 22 行是场消隐信号。在视频信号区，每行信号持续的时间相同，约为 62us；每行的行同步脉冲持续时间也相同，约为 4.7us。而在场消隐区，每行持续的时间会有所变化，每行对应的消隐脉冲持续的时间，尽管其中大多数为 3.5us，但也有变化。在场消隐区中，第 320 行的消隐脉冲持续的时间远长于其他消隐脉冲的时间，此脉冲即为场同步脉冲。

参数分析

因为 8013 DSP 的 AD 转换时间在 32M 主频的情况下最短为 2.125us。如果选用一个分辨率为 320 线的摄像头，则单行视频信号持续的时间约为 $20\text{ms}/320=62.5\mu\text{s}$ ，AD 对单行视频信号采样的点数将不超过 $[62.5/2.125]=29$ 个。若使用分辨率为 640 线的摄像头，则单行视频信号持续的时间约为 $20\text{ms}/640=31\mu\text{s}$ ，AD 对单行视频信号采样的点数将不超过 $[31/2.125]=14$ 个。这就是说，分辨率越高，单行视频信号持续的时间就越短，AD 对单行视频信号所能采样的点数就越少。

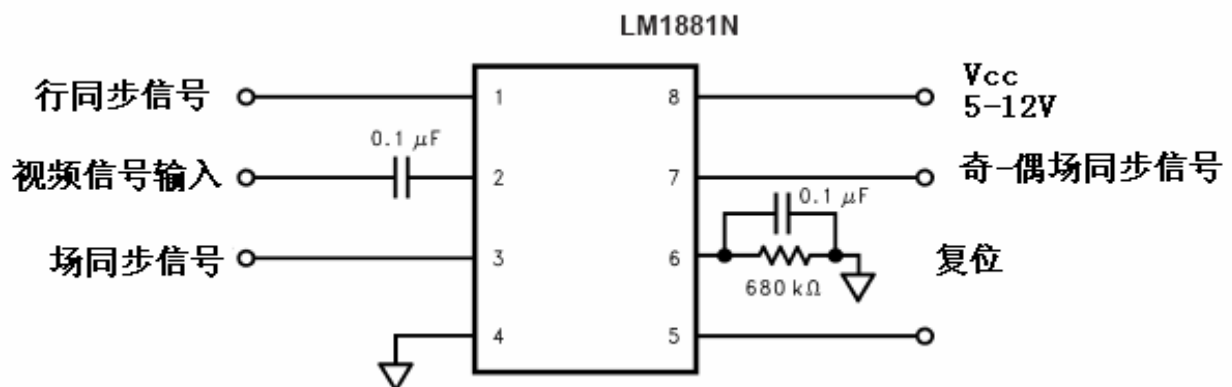
所以我们应该选择线数尽量低的摄像头

实际情况下，8013 每行采集到 32 点与理论相符。

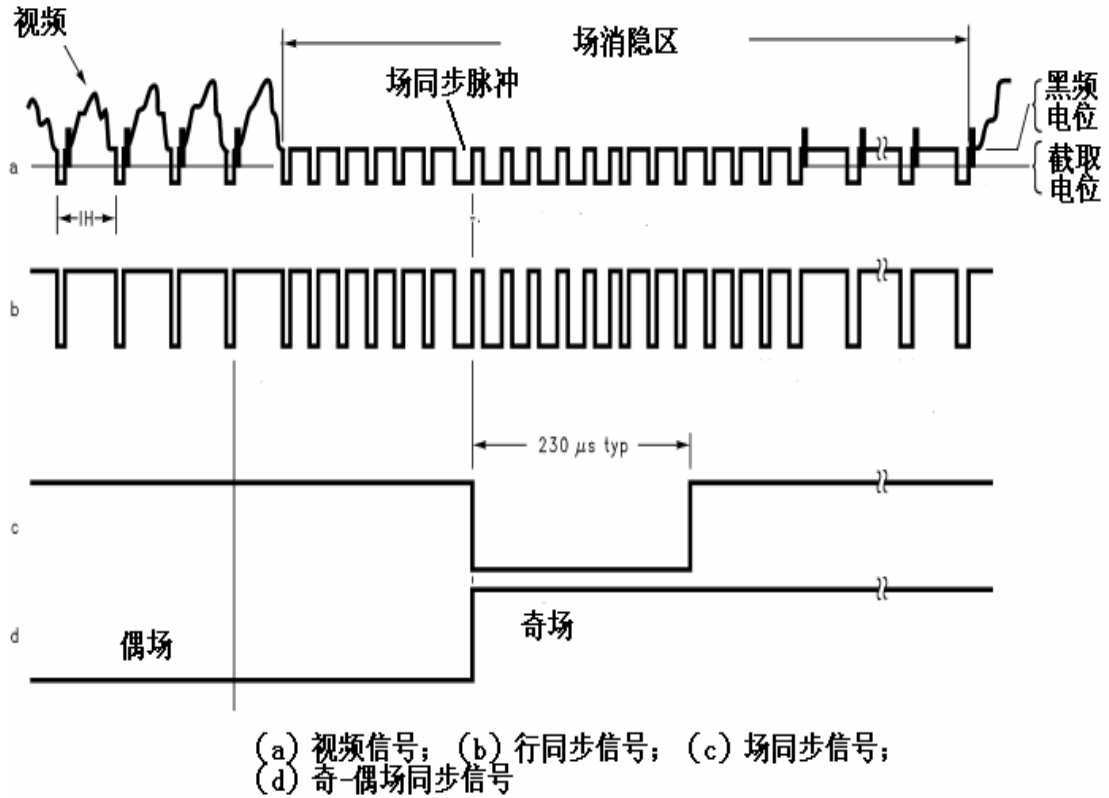
LM1881

要使用模拟摄像头，那么必然要用到 LM1881 视频同步信号分离芯片（简称 LM1881）。

1881 可从摄像头信号中提取信号的时序信息，如行同步脉冲、场同步脉冲和奇、偶场信息等，并将它们转换成 TTL 电平直接输给单片机 I/O 口作控制信号之用。



其中，引脚 2 为视频信号输入端，引脚 1 为行同步信号输出端（如图 4.7 中的 b）。引脚 3 为场同步信号输出端，当摄像头信号的场同步脉冲到来时，该端将变为低电平，一般维持 230 μ s，然后重新变回高电平（如图 4.7 中的 c）。引脚 7 为奇-偶场同步信号输出端，当摄像头信号处于奇场时，该端为高电平，当处于偶场时，为低电平。事实上，我们不仅可以用场同步信号作为换场的标志，也可以用奇-偶场间的交替作为换场的标志。

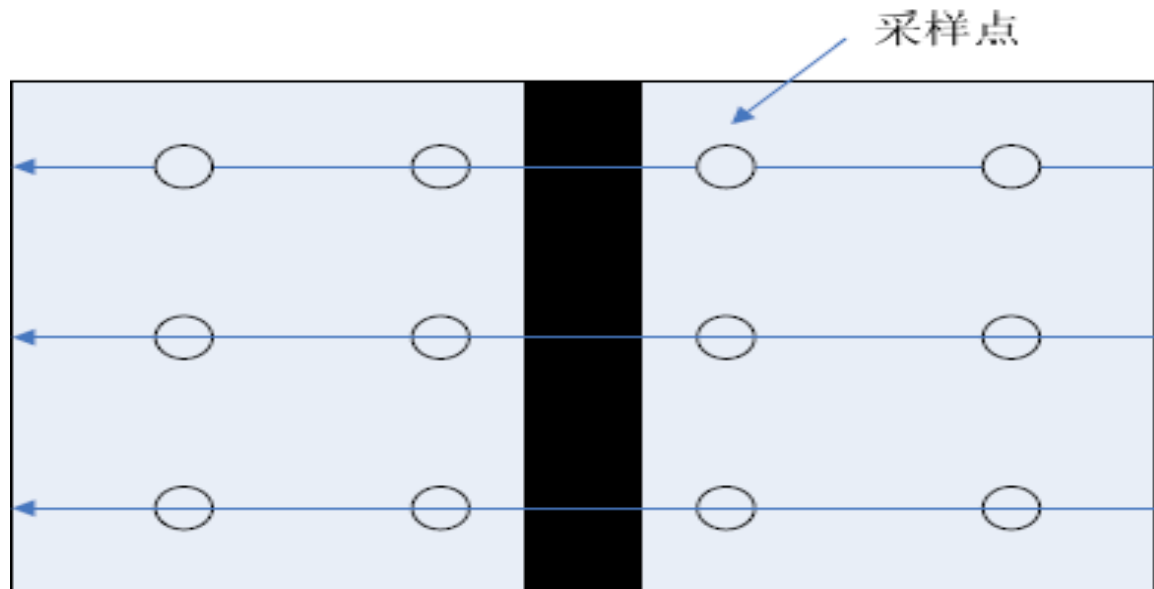


数据提取

摄像头采用的是隔行扫描的方式，为方便设计，我们忽略奇场和偶场在扫描位置上的细微差别，认为奇、偶场的扫描位置相同。由实测结果可知，所用摄像头每场信号的第 23 行至第 310 行为视频信号，即每场有 288 行视频信号，这已远远超出了系统所需的精度要求。实际我们没必要对这 288 行中的每一行视频信号都进行采样，如此会增大 DSP 存储和数据处理的负担，甚至会超出 DSP 的处理能力。再者，这样做是没必要的，事实上，小车的定位系统在纵向上只要有 10~20 个像素的分辨能力就足够了。因此，我们只需对这 288 行视频信号中的某些行进行采样就行了。假设每场采样 40 行图像数据，我们可以均匀地对 288 行视频信号进行采样，例如采样其中的第 7 行、第 14 行、第 21 行、……、第 273 行、第 280 行，即采样该场信号的第 29 行、第 36 行、第 43 行、……、第 295 行、第 302 行（每场开始的前 22 行视频为场消隐信号）

理论上摄像头的视距可以达到 1m 到 2m，这与摄像头的安装位置与安装角度有关。但是如果视距过远，那么远端的图像将变得很小（近大远小），指引线在拍摄画面上的宽度将会小于单行中相邻两采样点间的间隔。若指引线恰好位于相邻两采

样点之间，就可能会漏检测到该条指引线，为了保证指引线在任何位置都不会被漏检，指引线距摄像头不能太远，要依据每行采集的点数调整视距，以便不会漏检，所以由此可见，每行所能采集到的点数对于路径识别是很重要的。我们每行采集了 32 个点，视距大概在 70-80cm 时，黑线可以采集到 3 个点，还算比较理想。



路径规划

其实的得到赛道的图像后，我们的任务仅仅才完成了一部分。要想让小车能很好的跑下来，还有很多工作需要做，其中很重要的一点就是图像处理。

得到图像后，我们就要分析图像得到黑线的位置，怎么去分析图像，分析后怎么根据当前的路径进行路径规划，然后进行合理的控制，都是很重要的。

那么下面我们就简单看一下，如何进行黑线提取和路径规划。

黑线提取

目标指引线（即赛道上的黑线）宽度相对整个赛道较窄，因此只要提取目标指引线的某些特征点，就能反映出指引线的形状。

1 二值化算法

二值化算法的思路是：设定一个阈值 $valve$ （例如 45），对于视频信号矩阵中每一行，从左至右比较各像素值和阈值的大小。若像素值大于等于阈值，则判定该像素对应的是白色赛道；反之，则判定对应的是目标指引线。记下第一次和最后一次出现像素值小于阈值时的像素点的列号，算出两者的平均值，以此作为该行上目标指引线的位置。

Tips: 该算法的思想简单，具体实现时还可以一旦检测到左边缘后就退出该行扫描，这样上面的流程图将变得更加简洁。但是这种提取算法鲁棒性较差，当拍摄图像中只有目标指引线一条黑线时，还能准确提取出目标指引线，但当光强有大幅度的变化，或图像中出现其它黑色图像的干扰时，该算法提取的位置就有可能与目标指引线的实际位置偏离较大。

2 直接边缘检测算法

思路是：设定一个阈值（例如 15），对于视频信号矩阵中每一行，从左至右求出相邻两像素值的差值（左减右）。若差值大于等于阈值，则判定下一个的像素点对应的是目标指引线的左边缘，以此像点作为该列的特征点，记录下此像素点的列号，作为该行上目标指引线的位置。当然，可能出现差值始终小于阈值的情况，此时一种方法是令该行上目标指引线位置为 0，通过进一步滤波或拟合来修正；另一种方法是让该行上目标指引线位置和通过上一场视频数据求得的位置一样

Tips:

该算法较二值化方法而言，抗环境光强变化干扰的能力更强，同时还能削弱或消除垂直交叉黑色指引线的干扰。因为该算法在视频信号矩阵中是由左至右来寻找目标指引线的左边缘的，所以当黑色图像出现在目标指引线左方时，该算法无法排除干扰，而当其出现在右方时，则可以排除干扰。

3 跟踪边缘检测算法

这种算法直接边缘检测算法一样，也是寻找出目标指引线的左边缘，仍然用左边缘的位置代表目标指引线的位置。但跟踪边缘检测从视频信号矩阵每行中寻找左边缘的方法与上一小节介绍的不同。因为目标指引线是连续曲线，所以相邻两行的左边缘点比较靠近。跟踪边缘检测正是利用了这一特性，对直接边缘检测进行了简化。其思路是：若已寻找到某行的左边缘，则下一次就在上一个左边缘附近进行搜寻

Tips: 在首行边缘检测正确的前提下，该算法具有较强的抗干扰性，能更有效地消除垂直交叉黑色指引线的干扰，以及指引线外黑色图像的影响，始终跟踪目标指引线。另外，较之前两种算法，跟踪边缘检测算法的时间复杂度更低，因此效率更高。但值得注意的是第一行的左边缘位置对整个目标指引线的搜寻影响很大，一旦它的位置和实际导引线偏差较大，就会产生一连串的错误，这是不可容忍的。

路径规划

赛道元素

一般的赛道，分析下，有哪些典型的元素？

应该说包括下面几种元素：

1. 长直道
2. 蛇形弯（小 S 弯）
3. 直角弯
4. U 型弯
5. 大 S 弯
6. 圆形回旋
7. 十字交叉
8. 起始线

那么我们下面就分析一下在各种路面上应该采取 什么何种控制策略。

直道

毫无疑问直道应该加速。但是应该何时加速何时减速，都是应该考虑的。加速太晚，直道的速度的就加不起来，减速太晚了，过弯时就减不下速，容易冲出报道。最好就是在刚出弯时就开始加速，然后保持高速跑直道的大部分路程。然后利用摄像头视野远的优势，在检测出有弯道出现的迹象时就开始减速。这样就能比较理想的跑完直道。

直角弯、U 型弯

这个也很明显，过弯要减速，在过直角弯时应该把小车的速度减到能够安全过弯

的速度，平稳过弯，当然这是基于安全考虑，你也可以大胆尝试玩一下漂移或者甩尾。不过一般这都是要付出代价的。还有一点就是前面提到的，出弯之后及时加速把速度提起来。U型弯还有一点要注意的就是：要排除临近赛道的影响。

小 S 弯, 大 S 弯

要想跑的快，那么就要好好处理小 S 弯和大 S 弯，小 S 弯要能准确的判断出来，减小舵机摆角，滤除 s 形，直接当直道来跑，这样才跑的快。大 S 弯，不能当成直道来跑，那样是很容易跑出去的，大 S 弯的时候，要求车子转向要灵敏，及时对弯道做出反应，舵机响应不能滞后。

圆形回旋与十字交叉

赛道里非常不好跑的一个路况就是它了。大家都知道 ccd 有看的远的优点，那么同时它就有个很明显的缺点：有视场死角，太近的东西反而看不到了。所以在以 CCD 为采集元件的情况下，在过圆形回旋时会发生非常严重的“丢线”

还好它有一个特点就是圆形回旋都是和十字交叉线一起出现的。我们可以先准确的判断出十字交叉，同时检测出此时丢线严重，那么我们就知道小车进入十字回旋了，可以采取补救措施来补救丢线带来的影响，如以保持较大转角旋转，直到再次检测出十字回旋为止。

起始线与十字交叉

从第三届开始，增加对起始线的识别，要求小车在跑完 2 圈后能够自动停下了。这就要求小车每次经过起始线时都能准确的识别出起始线来，并且要求不能够和十字交叉相混淆，否则圈数的计算就会全乱了，轻则该停的时候不停，重则在十字交叉处停下来了，不能完成比赛。对比一下可以发现两者的差别。

注意

用 CCD 来识别路径还有一个很重要的问题不得不提。那就是环境参数的改变对图像采集的影响，以及赛道周围环境对图像采集的干扰。

CCD 是光学器件，那么光学器件共有的一个缺点就是容易受环境光线的影响，在不同的光照强度下采集到的图像是有较大的差距的。对于我们这个黑白赛道的识别就会造成不小的影响。所以要想跑的好，那么就得好考虑下如何尽量减少这方面的影响。

还有一点就是，CCD 的确看的远，视野看的宽，但是伴随而来的就是容易受赛道两旁其他东西的影响，赛道旁边的物品，人，远处的墙壁，以及临近的赛道等都可能被摄像头当成“赛道”来处理了，非常容易造成小车冲出赛道。所以必须想办法克服上述的缺点。用好这把双刃剑。

数字摄像头

技术总是在不断的进步，第三届已经有些队伍采用了数字摄像头来采集图像，相较之模拟摄像头，数字摄像头采集更快，每行采集的点数也越多，由于输出直接是数字量，可以直接用 I/O 口读进数字量，所以读取速度大大增加，另外由于不用 ADC 转换了，所以大大节省了系统的开销，使系统能够将更多的精力放在处理图像上和控制小车姿态上。本届可以有所尝试