**车牌粗定位**

基于目标检测的方法，使用Haar特征分类器：Haar-like特征+积分图方法+Adaboost算法+级联，将车牌与非车牌区分开，从而实现车牌粗定位。

**算法综述**：

Haar特征分类器算法，使用 Haar-like 特征提取图像的车牌特征，采用积分图快速计算特征数值，依据加权投票的方式挑选出弱分类器并将其构造成强分类器，有效提高分类器的检测速度。

为什么使用此算法？

以前我们使用的是**边缘检测**来定位车牌，然后对目标区域进行筛选，主要是使用一些特殊的知识或信息，直接检测单个图像。例如车辆的颜色，长宽比，边框，等形态学特征，来筛选车牌。但是此算法识别率不高，只能识别较为清晰的图片，对不同环境和不同角度的车牌识别准确率较差，所以我们采用基于统计的车牌检测方法：Adaboost算法

**算法优势：**

这种算法对车牌字符区域进行定位，具有识别准确率高、识别速度快和鲁棒性好等三方面性能良好。Haar特征分类器算法对车牌图像质量要求不敏感，即使图像中车牌处于倾斜、多个并存、夜光或车牌字符出现污渍、模糊等情况，本算法均能快速准确地定位出车牌区域。

**算法原理：**

分类：对已知物体类别总数的识别方式，我们称为分类，并且训练的数据是有标签的，比如已经明确指定了是车牌还是非车牌，这是一种监督学习。

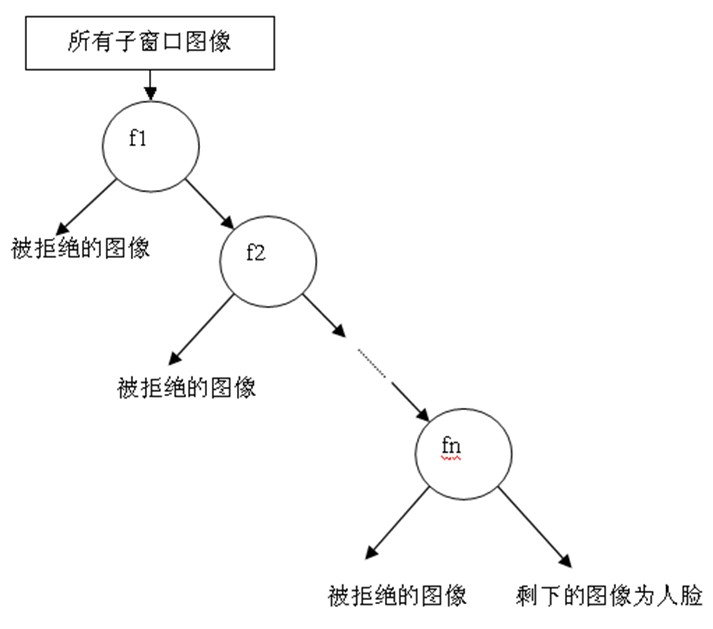
分类器：指对车牌和非车牌进行分类的算法。

Haar分类器算法要点：

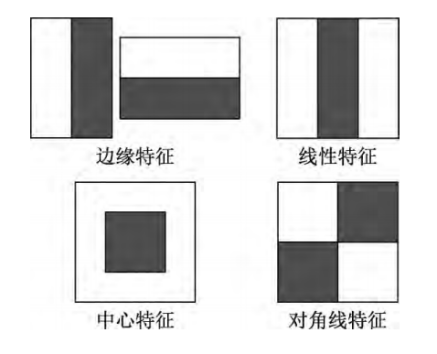
1. 使用Haar-like特征做检测。
2. 使用积分图对Haar-like特征求值进行加速。
3. 使用AdaBoost算法训练区分车牌和非车牌的强分类器。
4. 使用筛选式级联把强分类器级联到一起，提高准确率。

**Haar-like特征：**

原理：假设在车牌检测时我们需要有这么一个子窗口在待检测的图片窗口中不断的移位滑动，子窗口每到一个位置，就会计算出该区域的特征，然后用我们训练好的级联分类器对该特征进行筛选，一旦该特征通过了所有强分类器的筛选，则判定该区域为车牌。



对于灰度图像而言，Haar-like 矩形模板由两个或多个全等的黑白矩形相邻组合而成，而 Haar-like 特征值是白色矩形的灰度值的和减去黑色矩形的灰度值的和就是车牌特征图。如果把矩形放到非车牌区域，那么计算出的特征值和车牌的特征图是不一样的，而且越不一样越好，这些矩形的目的就是把车牌特征化，以区分车牌非车牌。



Haar-like特征图

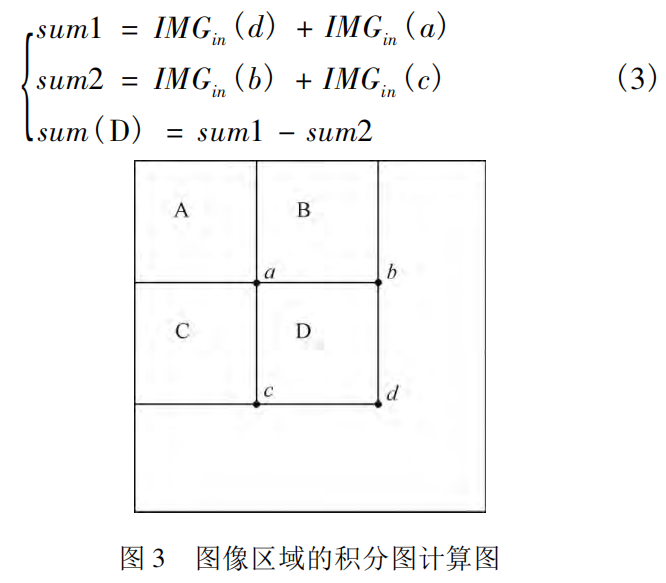
**积分图：**

类 Haar 特征表示框图进行特征值计算，即白色矩形的像素和减去黑色矩形的像素和，通过改变特征矩形的大小和位置，可以在图像中穷举出大量的类 Haar 特征，导致样本图像中的矩形特征数量巨大。如: 在 24 × 24 像素大小的样本图像中，类 Haar 的边缘特征、线性特征、中心和对角线特征对应的矩形特征数量可以达到 16 万个。而积分图就是只遍历一次图像就可以求出图像中所有区域像素和的快速算法，大大的提高了图像特征值计算的效率。因此为了实现类 Haar 特征值实时快速计算，采用了积分图计算方法。其计算原理为，针对图像像素点 ，其对应积分图中的像素为该点位置左上角所有像素之和。

即:



为图像在 (i i,j j) 处的像素值; 为图像在 (i,j) 处的积分图( 该位置左上角所有的像素的之和) 。如图3所示，对于一副图像中的 A、B、C、D 区域，假设每个区域的右下角像素点分别为 a 、b 、c 、d 像素点，则区域 D 的像素值和为:



图像中任一子区域的像素累加和都可以通过式( 3) 简单运算求得。而类 Haar 特征值是两个矩阵像素和的差，所以特征值计算，只与此特征矩形端点的积分图有关，无论此特征矩形尺度如何改变，只要遍历图像一次，就可以求得图像中所有类 Haar特征值。

**AdaBoost(自适应提升)算法：**

Adaboost 算法是分别训练多个弱分类器，再把这些弱分类器组合起来构造强分类器，最后用这个强分类器对样本集进行分类。Adaboost 算法是一个迭代的过程，它每一次迭代对样本是否正确分类进行判定，并用判定结果调整每个样本在训练过程中所占的权值。权值调整的原则是正确样本权重高，错误样本权重调低或直接去掉。每次迭代后对调整后的样本集分别训练分类器，再把各个分类器进行组合得到最终的级联结构即是最终我们要的级联强分类器。Adaboost 算法其实是弱分类的加权调整组合结果。

弱分类器：弱学习算法：准确性会比随机猜测算法稍好，错误率低于 0.5 的学习算法。

强分类器：强学习算法是一种准确率比较高且能在一定时间范围内完成的算法。

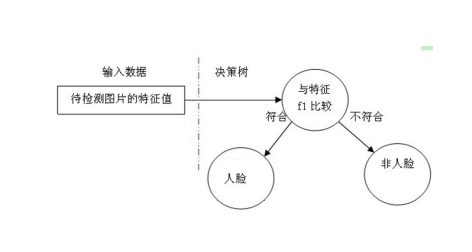
只要有足够的数据，弱学习算法就能通过集成的方式生成任意高精度的强学习算法。

**生成最优弱分类器：**

矩形特征数量和特征值被确定之后，我们需要对每一个特征 f 训练出一个弱分类器，定义一个弱分类器 ，找出 f 的最优解，使得这个弱分类器对所有的训练样本的分类误差最低。



其中 x 是待检子窗口，f (x) 是计算特征值的函数，θ 是通过训练得到的特征值的阈值，p 表示不等式的方向，取值为+1 或-1。



最重要的就是如何决定每个结点判断的输出，要比较输入图片的特征值和弱分类器中特征，一定需要一个阈值，当输入图片的特征值大于该阈值时才判定其为车牌。训练最优弱分类器的过程实际上就是在寻找合适的分类器阈值，使该分类器对所有样本的判读误差最低。

具体操作过程如下：

1）对于每个特征 f，计算所有训练样本的特征值，并将其排序。

扫描一遍排好序的特征值，对排好序的表中的每个元素，计算下面四个值：

全部人脸样本的权重的和t1；

全部非人脸样本的权重的和t0；

在此元素之前的人脸样本的权重的和s1；

在此元素之前的非人脸样本的权重的和s0；

2）最终求得每个元素的分类误差 θ = min((s1 + (t0 - s0)), (s0 + (t1 - s1)))

在表中寻找θ 值最小的元素，则该元素作为最优阈值。有了该阈值，我们的第一个最优弱分类器就诞生了。

**生成强分类器：**

生成强分类器需要T轮迭代，具体如下：

1. 给定训练样本集S，共N个样本，其中X和Y分别对应于正样本和负样本；T为训练的最大循环次数；

2. 初始化样本权重为1/N ，即为训练样本的初始概率分布；

3. 第一次迭代训练N个样本，得到第一个最优弱分类器；

4. 提高上一轮中被误判的样本的权重；

5. 将新的样本和上次本分错的样本放在一起进行新一轮的训练。

6. 循环执行4-5步骤，T轮后得到T个最优弱分类器。

7.组合T个最优弱分类器得到强分类器，组合方式如下：



相当于让所有弱分类器投票，再对投票结果按照弱分类器的错误率加权求和，将投票加权求和的结果与平均投票结果比较得出最终的结果。

**生成级联分类器：**

通过AdaBoost算法训练出来的强分类器，在现实的车牌识别中，只靠一个强分类器还是难以保证检测的正确率，这个时候需要训练出多个强分类器将他们强强联手，最终形成正确率很高的级联强分类器，也就是我们最终的目标Haar分类器。

级联强分类器的策略是，将若干个强分类器由简单到复杂排列，希望经过训练使每个强分类器都有较高检测率，而误识率可以放低，比如几乎99%的人脸可以通过，但50%的非人脸也可以通过，这样如果有20个强分类器级联，那么他们的总识别率为0.99^20约等于98%，错误接受率也仅为0.5^20约等于0.0001%。

**这样的效果就可以满足现实的需要了，但是如何使每个强分类器都具有较高检测率呢，为什么单个的强分类器不可以同时具有较高检测率和较高误识率呢？**

设K是一个级联检测器的层数，D是该级联分类器的检测率，F是该级联分类器的误识率，di是第i层强分类器的检测率，fi是第i层强分类器的误识率。如果要训练一个级联分类器达到给定的F值和D值，只需要训练出每层的d值和f值，这样：

d^K = D, f^K = F

级联分类器的要点就是如何训练每层强分类器的d值和f值达到指定要求。AdaBoost训练出来的强分类器一般具有较小的误识率，但检测率并不很高，一般情况下，高检测率会导致高误识率，这是强分类阈值的划分导致的，要提高强分类器的检测率既要降低阈值，又要降低强分类器的误识率就要提高阈值，这是个矛盾的事情。据参考论文的实验结果，增加分类器个数可以在提高强分类器检测率的同时降低误识率，所以级联分类器在训练时要考虑如下平衡，一是弱分类器的个数和计算时间的平衡，二是强分类器检测率和误识率之间的平衡。

具体训练方法如下,我用伪码的形式给出：

1）设定每层最小要达到的检测率d，最大误识率f，最终级联分类器的误识率Ft；

2）P=车牌训练样本，N=非车牌训练样本，D0=1.0，F0=1.0；

3）i=0；

4）for : Fi>Ft

++i;

ni=0;Fi=Fi-1;

for : Fi>f\*Fi-1

++ni;

利用AdaBoost算法在P和N上训练具有ni个弱分类器的强分类器;

衡量当前级联分类器的检测率Di和误识率Fi;

for : di<d\*Di-1;

降低第i层的强分类器阈值;

衡量当前级联分类器的检测率Di和误识率Fi;

N = Φ;

利用当前的级联分类器检测非车牌图像，将误识的图像放入N

**目标检测：**

检测体系以现实中的一副大图作为输入，然后对图片进行多尺度的检测，级联分类器的检测框架就是一个多尺度缩放+滑动窗口遍历搜索的框架。

**功能实现：**

粗定位模块分为**样本训练**和**识别测试**两个主要部分。

**样本训练：**

1. 样本的创建(正样本和负样本)

正样本：只包含检测目标的图片即车牌

1.所有图片尺寸必须相同，例如36\*9

2.进行灰度处理

负样本：不包含检测目标的任何图片

1. 不要求样本尺寸，但样本量要大于正样本；且负样本不能重复，增大负样本差异性。
2. 进行灰度处理

2、训练分类器(生成.xml)

对Haar 级联分类器进行训练。利用OpenCV提供的中文车牌样本库，选取4700个正样本12000个负样本，调用 opencv\_ haartraining. exe和opencv\_createsamples.exe命令工具进行样本训练，该工具封装了 Haar 特征提取以及 Adaboost 分类器训练过程，训练后的Haar分类特征可记录在对应的 xml 文档中。

**识别测试：**

使用opencv中的cascadeclassifier函数进行多尺度检测即可。