

**程序开发文档**

题 目 车辆标牌识别

学院（部） 信息工程学院

专 业 计算机科学与技术

班 级 2018240201

成 员 朱子康(2018902564)

唐卓然(2018901135)

赵孟泰(2018904182)

**目录**

**一、概述1**

**二、开发环境1**

**2.1 开发环境1**

**2.2 环境配置1**

**三、程序功能及实现方法1**

**3.1 多特征的车牌定位及提取方法1**

**3.1.1基于HSV色彩空间的车牌定位1**

**3.1.2基于边缘检测的再次定位2**

**3.2基于投影法的字符分隔3**

**3.2.1大津法进行二值化3**

**3.2.2倾斜矫正4**

**3.2.3边缘处理4**

**3.2.4基于投影法的字符分割4**

**3.3 基于KNN分类器的字符识别5**

**四、实验结果与分析6**

**五、总结与心得7**

1. **概述**

针对停车场出入口场景的车牌自动识别，我们完整地完成了车牌识别的三个流程：定位、分割、识别，实现了在特定场景下较为可靠的车牌识别系统。我们综合运用了HSV色彩空间、边缘检测，实现了对车牌的定位。通过竖直投影，倒序分割等方法，实现了对测试图片的字符分割。利用KNN（K最近邻，K-NearestNeighbor）完成了对字符的识别。经测试，使用上述方法实现的车牌识别系统在前述场景下的车牌自动识别效果较好，有一定的实用价值。

## 开发环境

#### 开发环境

Windows10 操作系统，集成开发环境为Microsoft Visual Studio 2017，OpenCV版本为4.4 x64。

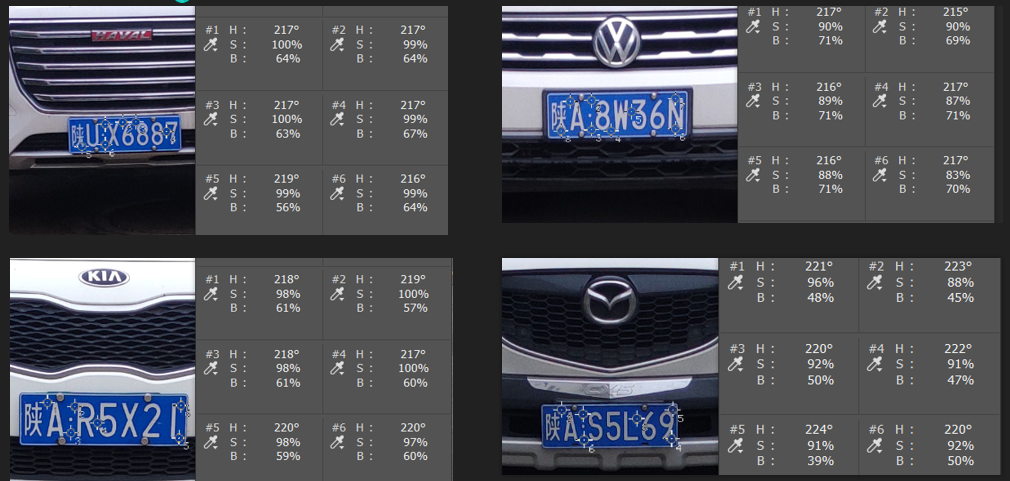
#### 2.2环境配置

打开项目属性，将opencv目录下的build\include和build\include\opencv2添加到VC++目录下的包含目录，build\x64\vc15\lib添加到VC++目录下的库目录，将opencv\_world440d.lib添加到链接器下的输入页中的附加依赖项中。将调试选项改为Debug x64，点击本地Windows调试器即可编译运行。

1. **程序功能及实现方法**

#### 多特征的车牌定位及提取方法

###### 3.1.1基于HSV色彩空间的车牌定位

该方法适用于对非蓝色车辆进行车牌定位。HSV颜色空间有三个分量，分别是 H（色调）、S（饱和度）、V（亮度）。通过分析由特定摄像头拍摄得到的样例，得到的蓝色车牌的HSV值的范围为色调：210~225°，饱和度：0.85~1，亮度：0.5~1，如图3.1所示。这些参数可较好地应用于选定的摄像头，但是不能保证用于其他任意摄像头的效果。

**图3.1 蓝色车牌对应HSV范围分析**

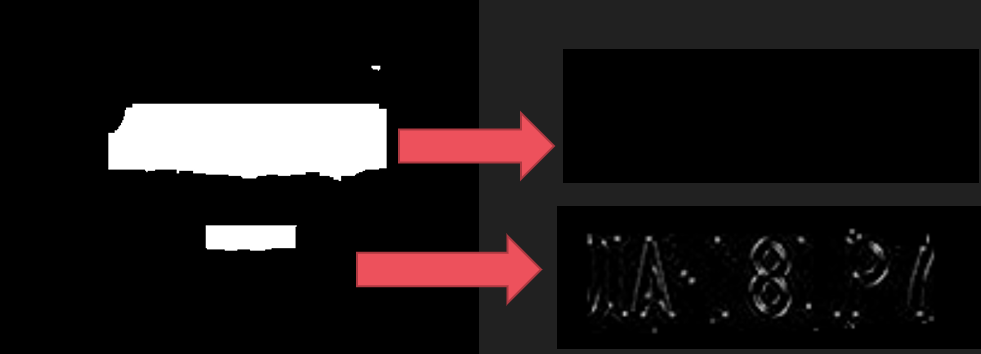
由于仅使用上述HSV参数进行掩膜获取蓝色车牌的方案在遇到蓝色车辆时会错误地将其他区域（如蓝色的引擎盖）当作车牌，如图3.2所示，故需要进行改进。



**图3.2 HSV方案失效示例**

**3.1.2基于边缘检测的再次定位**

将使用不同白色区域进行分别掩膜后的图像二值化，然后再对二值化图像进行边缘检测，由于车牌区域白色字符的存在，对应于车牌区域的掩膜后二值化图将检测到较多边缘，而其他区域检测到的边缘远远少于车牌区域，如图3.3所示。利用这种特征，进行二次定位，即可较为精准地定位到车牌区域。



**图3.3 利用边缘特征进行再定位**

#### 基于投影法的字符分割

* + 1. **大津法进行二值化**

大津法（OTSU）是一种确定图像二值化分割阈值的算法，该方法又称作最大类间方差法，因为按照大津法求得的阈值进行图像二值化分割后，前景与背景图像的类间方差最大。它被认为是图像分割中阈值选取的最佳算法，计算简单，不受图像亮度和对比度的影响，它是按图像的灰度特性，将图像分成背景和前景两部分。因方差是灰度分布均匀性的一种度量,背景和前景之间的类间方差越大,说明构成图像的两部分的差别越大,当部分前景错分为背景或部分背景错分为前景都会导致两部分差别变小。因此,使类间方差最大的分割意味着错分概率最小，适用于大部分需要求图像全局阈值的场合。优点：计算简单快速，不受图像亮度和对比度的影响。缺点：对图像噪声敏感；只能针对单一目标分割；当目标和背景大小比例悬殊、类间方差函数可能呈现双峰或者多峰，这个时候效果不好。



**图3.4 大津阈值二值化效果图**

**3.2.2 倾斜矫正**

利用Radon变换找到处于边界的白色直线边缘，找到最长的一条并计算直线的斜率，反推出图像旋转过的角度，并将图像按此角度反向旋转实现对倾斜图像的矫正，如图3.5所示。



**图3.5 倾斜矫正**

**3.2.3 边缘处理**

二值化后的图像，可以看到有铆钉和边缘，会导致垂直投影分割字符的方法失败，所以需要去除边缘。①去除上下两边的边缘：以上边缘为例，从上到下（从第0行到图像正中央一行）统计每行像素值由255到0（由白到黑）的跳变次数，找到跃变次数不足7次（文字区域跃变次数一定大于等于7次，因为有7个字符）的最下方一行，以此为边界，边界之上部分全部置为黑色。下边缘同理②去除左右两边的边缘：统计位于左边几列的白色区域宽度，若过窄则认为是白色边缘而非字符，右边缘同理。



**图3.6 去除边缘效果图**

**3.2.4基于投影法的字符分割**

对去除边缘后的图像进行投影法分割字符：先设置一个数组统计每一列的白色像素个数，之后从右往左遍历图像，若遍历到某一列的白色像素个数值不为0，标记列号，表示从这一列开始进入字符区；然后继续遍历到某一列白色像素个数值为0，再标记列号，表示字符区结束了，并以这两个列号和上一步边缘处理的两个行号为范围把图像分割出来存到图像数组里。如果遇到“川”字车牌，投影法会把一个字分割成三个字，所以是从右往左遍历图像，并且判断如果当前分割的图像数目已经达到车牌字符总数减1，就直接把剩余的图像分割出来。



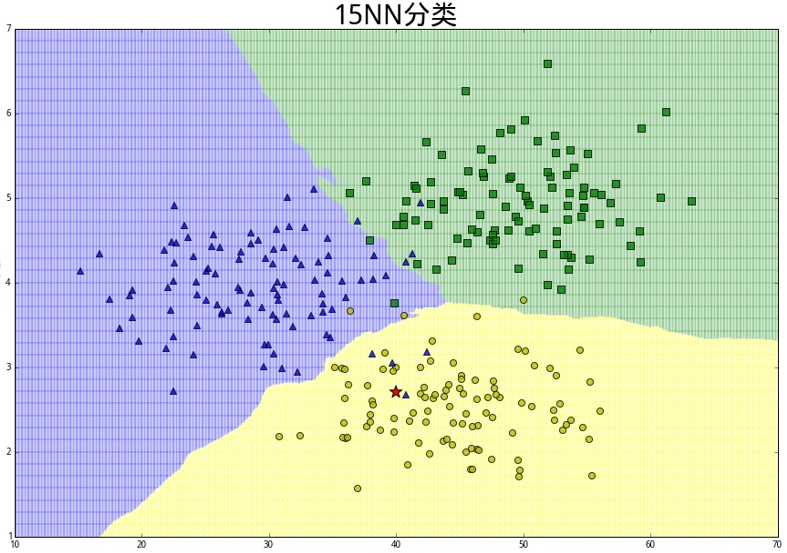
**图3.7 川字车牌字符分割效果图**



**图3.8 其他车牌字符分割效果图**

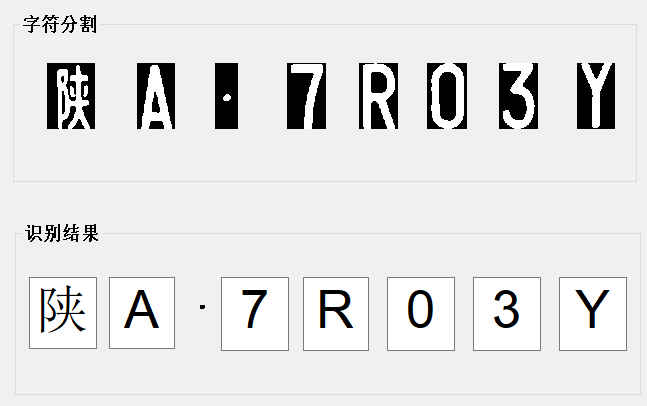
#### 基于KNN分类器的字符识别

由于传统的数字图像方法进行字符识别时准确率难以达到要求，我们采用了机器学习中最简单的分类算法之一K近邻算法进行字符识别。KNN算法的核心思想是，如果一个样本在特征空间中的K个最相邻的样本中的大多数属于某一个类别，则该样本也属于这个类别。如图3.8所示，当给出一定数量的三种样本点进行训练后，KNN将空间分为三个部分，依据待测试点在特征空间中所处的区域将其进行分类。



**图3.9 KNN示例**

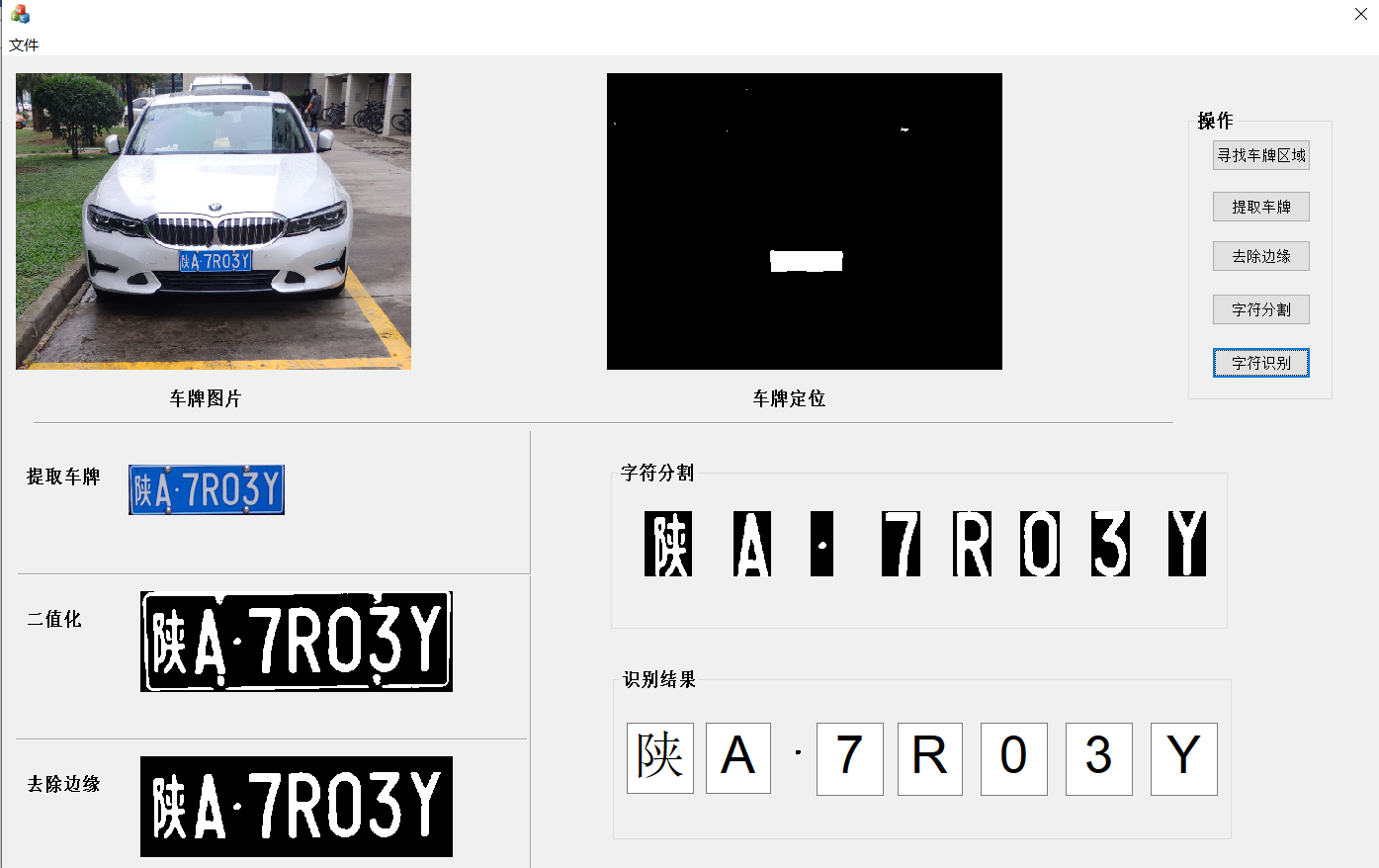
我们针对每种字符收集了100到400张图片作为训练集，最终字符识别效果达到了预期要求，见图3.9所示。



**图3.10 KNN字符识别效果**

## 实验结果与分析

对拍摄的25张图片进行识别，仅有一张图片中的“陕”字被识别为“桂”，其余均能正常识别，下为其中两张的识别结果，见图4.1和4.2，其中图4.2模拟了会对车牌定位造成困难的蓝色车辆以及会对字符分割造成困难的“川”字车牌。



**图4.1 最终程序识别示例**



**图4.2 川字车牌和蓝车识别成功。**

## 总结与心得

通过此次大作业，我们对整个车牌识别的流程方法有了大致的认识。最开始时我们没有认识到设定特定应用场景的重要性，而企图做出一个全场景下可用的车牌识别系统，但很快我们发现全场景适用的车牌识别系统的实现难度之大不是我们现阶段的能力所能解决的，因此我们转向了停车场出入口车牌自动识别这一较小的特定应用场景。

在车牌定位的实现过程中，最开始使用的RGB色彩空间，之后改用了HSV色彩空间，我们发现HSV色彩空间确实在表示上很有优势，十分适合应用于各种绘图软件的色盘。由于蓝色车辆对基于色彩提取车牌可会造成干扰，我们又使用车牌的文字边缘特征进行了二次定位。在这个过程中我们也发现，使用机器学习及神经网络也能完成车牌的定位（图像标注，相关的软件有Labelme），在复杂的全场景可应用的商用车牌识别系统中，也许使用更多的是这些方法而非传统的数字图像处理方法。

使用投影法进行字符分割时，由于车牌的金属边框和铆钉等的干扰导致分割并不顺利，但是如果腐蚀消除噪音又会同时对字符区域造成影响，经过一番搜索后发现了一种简单而有效的方案，统计每一行的白到黑跃变次数，由于字符区7个字符跃变次数一定大于7，而白色边缘没有那么多跃变次数，由此可将无用的白色边缘及铆钉去掉。之后再进行垂直投影便可获得比较好的分割后的单个字符图像。

在进行字符识别时起初尝试了传统的数字图像处理方法（不变矩特征、模板匹配等）发现效果并不理想，故采取了机器学习中一种最简单的KNN算法。在实际使用过程中发现各种各样的机器学习、神经网络算法如同炼丹一般，理论充满了不明确性实际应用起来也很是不明确，就像新闻报道中的在道路标志牌上动手脚导致特斯拉自动驾驶误判出事故描述的一样，一些微小的变化可能使整个模型完全失效（神经网络对抗攻击等）。

另外发现现在存在所谓“端到端”的车牌识别方案，即不需进行字符分割这一步，而是直接由整块车牌得出车牌号。

本来想尝试一些功能，如对不同角度拍摄拍摄的车牌通过错切、投影等方法以实现自动矫正，对蓝色车牌、黄色车牌的识别，以及在不同光照条件下的自适应算法等。但是这些主要是全应用场景下必须关注的问题，而在我们选定的较小应用场景下可以认为不必考虑。同时我们发现，漫无边际的增加功能反而会使得工作方向不明晰。

这次大作业让我们认识到了数字图像处理这门课程是一门实践性很强、很适合投入应用的课程，增加了我们使用OpenCV进行编程的经验，锻炼了我们团队合作能力及信息搜索能力，还使得我们学到了一些入门级的机器学习、神经网络方法。本次大作业，由朱子康完成车牌定位部分，唐卓然完成字符分割部分，但在实际操作中遇到了许多细节问题，这些细节问题由朱子康和唐卓然共同完成，赵孟泰和朱子康完成KNN字符识别部分，mfc界面由唐卓然完成。总之，这次大作业使我们受益匪浅。