

华南师范大学

本 科 毕 业 论 文

论文题目：基于 OpenCV 的车牌自动识别系统的研究

指导老师：潘家辉

学生姓名：邓园园

学 号：20152005036

院 系：软件学院

专 业：软件工程

毕业时间：2018 年 7 月

2018 年 6 月 制

基于 OpenCV 的车牌自动识别系统的研究

中文摘要

随着经济与社会的不断发展，机动车数量的增加使各种智能交通系统应运而生，车牌自动识别系统是智能交通系统的基础。本文对基于 OpenCV 的车牌自动识别系统展开研究，主要围绕车牌图像预处理、车牌检测、字符分割和字符识别四部分进行阐述。

车牌图像预处理包括图像增强和去噪。得到预处理的图像后，采用一种基于颜色的车牌定位方法、基于 Sobel 的边缘检测的车牌定位方法、基于文字区域的车牌定位方法相结合的算法进行车牌区域粗定位，获得若干个候选车牌区域，将其进行后续处理操作以到达进一步筛选车牌的目的。再将筛选后的候选车牌集合输入到训练好的 SVM 模型中进行精确定位，得到车牌图像块。采用基于字符轮廓提取的分割方法进行字符分割，得到单一的字符块，用于输入字符识别模块中。采用基于 BP 神经网络的车牌字符识别方法进行字符识别，将识别后的结果进行组合得到车牌号码。

四组不同场景的实验表明，系统能够适应多种交通场景，在单一正常场景下的车牌识别正确率能达到 90% 以上，在综合复杂场景下的车牌识别率约为 85%。

关键词：车牌检测，字符分割，字符识别，支持向量机，神经网络

Abstract

With the continuous development of economy and society, the increasement of the number of motor vehicles has led to the emergence of various intelligent transportation systems. The automatic license plate recognition system is the basis of the intelligent transportation system. This paper studies the automatic recognition system of license plate based on OpenCV, mainly on the four aspects of license plate image preprocessing, license plate detection, character segmentation and character recognition.

License plate image preprocessing includes image enhancement and denoising. After obtaining the pre-processed image, a color-based license plate location method, a Sobel-based edge detection based license plate location method, and a text region-based license plate location method are combined to calculate the license plate area coarsely, and several candidate license plate regions are obtained. And carry out subsequent processing operations to achieve the purpose of further screening the license plate. The selected candidate license plate set is input into the trained SVM model for precise positioning to obtain a license plate image block. The character segmentation method based on character contour extraction is used to perform character segmentation, and a single character block is obtained for inputting the character recognition module. The license plate character recognition method based on BP neural network is used for character recognition, and the recognized results are combined to obtain license plate numbers.

Experiments in four different sets of scenarios show that the system can adapt to a variety of traffic scenarios, the correct rate of license plate recognition in a single normal scene can reach more than 90%, and the license plate recognition rate in a complex scene is about 85%.

Key word: License plate detection, character segmentation, character recognition, SVM, neural network

目录

基于 OpenCV 的车牌自动识别系统的研究	1
中文摘要	1
Abstract	11
1. 绪论.....	1
1.1 选题背景及意义.....	1
1.2 国内外研究现状.....	1
1.3 课题研究内容.....	2
1.4 论文结构安排.....	4
1.5 本章小结.....	4
2. 车牌图像预处理.....	5
2.1 图像增强.....	5
2.2 图像去噪.....	6
2.3 本章小结.....	7
3. 车牌检测.....	8
3.1 车牌区域粗定位.....	8
3.1.1 基于颜色的车牌定位方法.....	9
3.1.2 基于 Sobel 的边缘检测的车牌定位方法.....	13
3.1.3 基于文字区域的车牌定位方法.....	15
3.2 车牌区域粗定位的后续处理.....	17
3.2.1 二值化.....	17
3.2.2 闭操作.....	17
3.2.3 轮廓跟踪.....	18
3.2.4 外接矩形尺寸验证.....	18
3.2.5 倾斜偏斜校正.....	20
3.3 车牌区域精确定位.....	22
3.3.1 SVM 理论.....	23
3.3.2 SVM 训练.....	24
3.3.3 车牌分类.....	25
3.4 本章小结.....	26
4. 字符分割.....	27
4.1 常用的字符分割方法.....	27
4.2 基于字符轮廓提取的分割方法.....	27
4.3 本章小结.....	30
5. 字符识别.....	31
5.1 常用的字符识别方法.....	31
5.2 基于神经网络的字符识别方法.....	31
5.2.1 神经网络概述.....	31
5.2.2 BP 神经网络在车牌字符识别的应用.....	32
5.3 本章小结.....	34

6. 车牌自动识别系统在 Qt 上的设计与实现.....	35
6.1 OpenCV 简介	35
6.2 Qt 图形化界面的设计与实现	36
6.3 实验	37
6.3.1 评估指标.....	37
6.3.2 实验方案.....	37
6.3.3 实验结果与分析.....	37
6.4 本章小结.....	41
7. 总结与展望.....	42
参考文献	44

1. 绪论

1.1 选题背景及意义

随着经济与社会的发展，加快了城市化进程的发展，人们的生活水平和质量得到提高，越来越多的人选择自驾的方式出行，导致机动车数量的爆增，在给人们生活带来便利的同时也给社会治安带来了巨大的压力，并导致城市拥堵、交通事故等社会问题，甚至很多城市采取排号限购的方式来限制人们购买车辆。为了缓解交通压力，各种智能交通系统应运而生，运用先进的科学技术来管理交通运输，智能交通系统是一个综合性系统，包含了机通管理、车联网、公共交通、应急指挥、货物运输、电子政务、公共服务、电子收费等子系统。而机动车是智能交通系统的主体，车牌号码是机动车的唯一标识，因此获取机动车的车牌号码是智能交通系统运行的基础。

自动车牌检测识别技术(ALPR)是指在摄像头拍下车辆图片后能够自动检测车牌区域并从中提取车牌信息进行处理的技术^[1]。车辆车牌号码识别技术(LPR)综合了图像处理 and 模式识别等前沿科学技术，通过对车辆图像进行分析，从而得到汽车唯一的车牌号码^[2]。LPR 技术由美国道路交通管理协会在 20 世纪 80 年代初期率先向外界发布，并于 90 年代初被广泛使用并推行至市场，随着 LPR 技术的不断成熟，LPR 被广泛运用于智能交通系统中，提高交通管理的效率^[3]。车牌自动识别系统是智能交通系统的重要组成部分，自动识别车牌号码的过程大大减少了人工处理的工作量，由于人工处理存在效率低、易出错等问题，在一定程度上也提高了车牌号码获取的正确性。车牌自动识别系统被广泛应用于停车场收费管理、交通违章抓拍、城市交通监控、交通流量控制指标测量、ETC 收费管理^[4]、违规犯法车辆追踪等，对交通自动化的实现具有重大意义。

在实际应用场景中由于光照、天气、车牌遮挡倾斜等原因，导致车牌自动识别系统出现车牌定位不准确、车牌识别不正确等问题，对于一些极端环境和干扰时还有很大的上升和探索空间，对车牌自动识别系统的研究仍存在很大的前景。

1.2 国内外研究现状

国外车牌识别系统的研究起步较早，始于上世纪八十年代，至今研究已

有二十多年，并且具有较高的适应性和鲁棒性。英国是第一个研发出自动车牌检测系统的国家，随后美国、加拿大、日本等国家也研发出适用于本国车牌检测识别的系统。目前为止，车牌识别系统已在德国、美国、日本等国家得到广泛应用，而且车牌的有效识别率高达 95%，在一秒左右的时间内能够识别 3 辆机动车^[5]。以色列的 Hi-Tech 公司研发的多种 seeC/arysstem 系统可适用于几个不同国家的车牌识别，但不适用于识别中国车牌，由于国外车牌规范统一，国外车牌识别系统的识别率能达到较高水平。

我国的车牌研究起步较晚，大约在上世纪八十年代末，由于我国车牌形式多样，位数及颜色都不统一，而且国外并无中文字符研究的先例，所以我国对车牌识别的研究具有一定的挑战性。通过许多专家的钻研，我国的车牌研究发展迅速并日渐成熟，目前较为成功的案例应该是中科院研究所的刘智勇^[6]等人开发的 LPR 系统，车牌定位识别准确率高达 99.4%，并运用于现有产品汉王公司的汉王眼中，取得了不错的效果。

目前国内外的车牌识别技术都比较成熟，并在实际生活中得到具体应用，未来必然得到大量的普及使用。但车牌识别技术的发展仍存在挑战，主要是环境变化因素太大影响车牌的识别率，例如行驶车辆的干扰性太大导致不能精确地识别，因此高精度和高适应性的车牌识别技术将成为未来的发展趋势。

1.3 课题研究内容

本文设计了一套基于 OpenCV 的车牌自动识别系统，将汽车牌照从复杂背景中提取并识别出来，针对输入的蓝色和黄色的中文车牌可以输出车牌号码。该车牌自动识别系统主要包括车牌图像预处理、车牌检测、字符分割和字符识别四部分内容，而车牌检测包括车牌区域粗定位、车牌区域后续处理、车牌区域精确定位，系统流程图如图 1-1 所示。

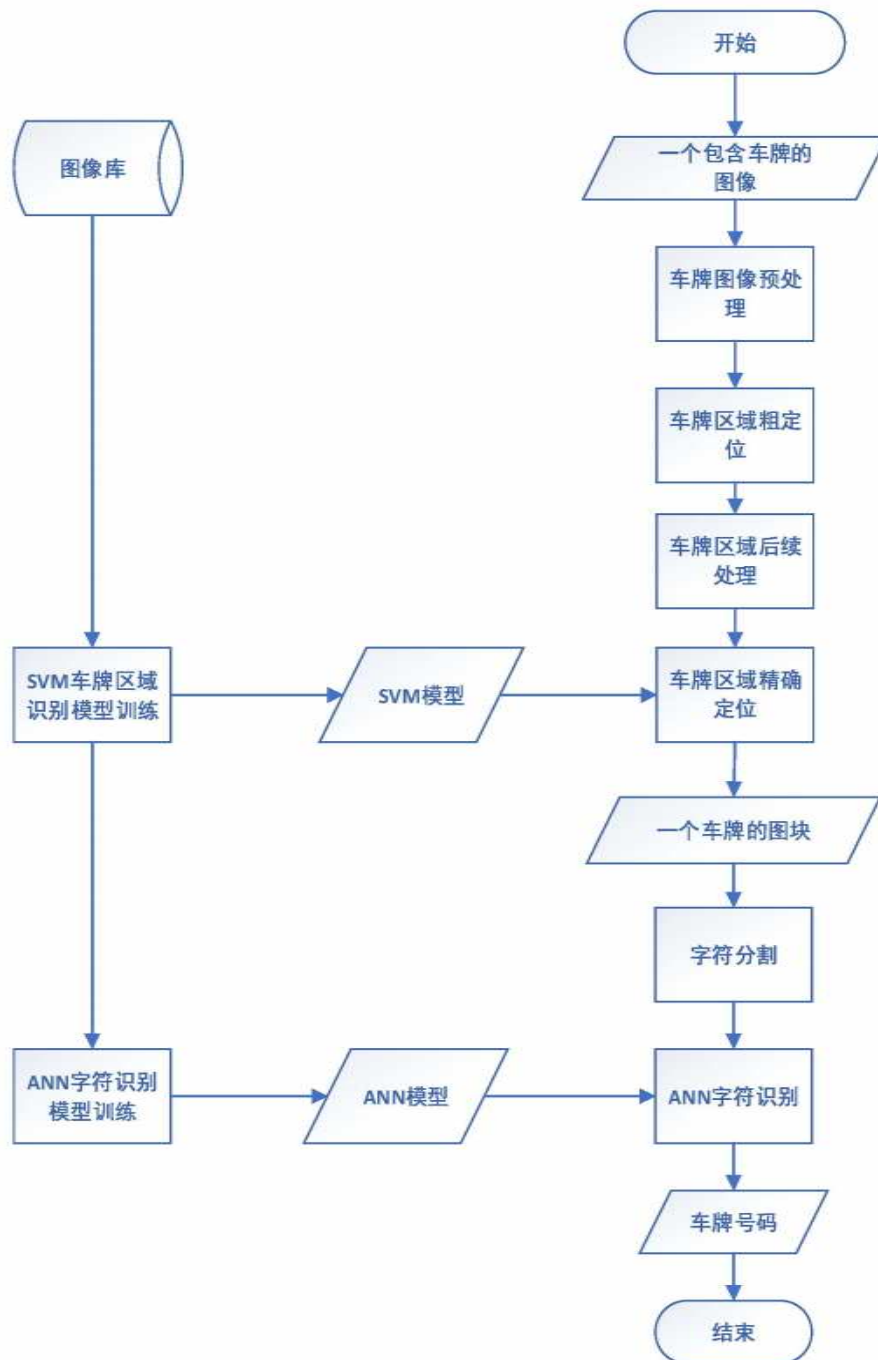


图 1-1 车牌自动识别系统总流程图

(1) 车牌图像预处理：由于采集到的车牌图像受到环境的干扰，影响到车牌图像的质量，会对后期的车牌检测的准确度造成影响。通过对车牌图像进行图像增强、图像去噪等预处理操作，以还原车牌图像最真实的状态。

(2) 车牌检测：主要分为车牌粗定位、车牌粗定位后续处理、车牌精确定位三部分。本文在车牌粗定位时采取一种综合基于颜色的车牌定位方法、基于 Sobel 的边缘检测的车牌定位方法、基于文字区域的车牌定位方法的算法，获得候选车牌区域。将候选车牌进行一系列的二值化、闭操作、轮廓跟踪、外接矩形尺寸验证、倾斜偏斜校正的后续处理操作后，输入到训练好的

支持向量机(SVM)模型中进行分类,得到精确的车牌定位。

(3) 字符分割:采用基于字符轮廓提取的分割方法进行字符分割,通过颜色判断、二值化、去除柳钉和边框、取轮廓、求外接矩形、中文字符特殊处理、截取图像、归一化这一系列步骤,得到单一的字符块,用于输入字符识别模块中。

(4) 字符识别:由于神经网络具有分类准确率高、鲁棒性强的优点,采用基于BP神经网络的车牌字符识别方法进行字符识别,主要包括样本采集、神经网络训练、特征提取和字符识别等步骤,通过字符识别后,将识别后的结果组合在一起得到车牌号码。

1.4 论文结构安排

本文将介绍基于OpenCV的车牌自动识别系统的重要研究算法及其图形化界面的实现,各章内容安排如下:

第一章:绪论。主要介绍选题背景及意义和国内外研究现状,再介绍了本文研究的主要内容。

第二章:车牌图像预处理。主要介绍了图像增强和图像去噪两种预处理方法。

第三章:车牌定位。主要分为车牌区域粗定位、后续处理、车牌区域精确定位三大部分,车牌区域粗定位介绍了基于颜色、边缘检测、文字区域的定位方法的三种算法,车牌区域精确定位介绍了SVM算法。

第四章:字符分割。首先简单介绍常用的字符分割方法,再重点介绍基于字符轮廓提取的分割方法。

第五章:字符识别。首先简单介绍常用的字符识别方法,再重点介绍基于BP神经网络的车牌字符识别方法,包括其工作原理和步骤。

第六章:车牌自动识别系统在Qt上的设计与实现。首先介绍了系统的结构和模块,并对OpenCV进行了简单的介绍,阐述系统在Qt平台下的GUI设计与实现,最后在系统上进行四组不同的实验,并对实验结果进行分析。

第七章:总结与展望。对本文所做的工作进行总结,并说明系统存在的不足之处及改进。

1.5 本章小结

本章主要介绍选题背景及意义和国内外研究现状,然后介绍了本文研究的主要内容,最后列举本文各章节的主要内容。

2. 车牌图像预处理

由于车牌图像通过摄像机等电子设备采集而来，在采集阶段容易受到外界环境的干扰，光照、天气环境、摄像机设备等外界因素都会影响采集到的车牌图像的质量，而且车牌图像在传输过程也有可能引起失真，会对后期车牌检测与识别的精度造成影响。为了减少外界因素对车牌图像质量的影响，对车牌图像进行图像增强、图像去噪等预处理操作，以提高车牌图像的质量，尽可能还原车牌最原始的状态。

2.1 图像增强

在采集和传输过程中，由于某些客观因素导致车牌图像的亮暗对比度不够、画面效果不佳，可以通过图像增强的图像处理方法提高图像中车牌区域与非车牌区域的亮暗对比度，加大亮暗差异，有利于后续对车牌的定位与识别。车牌图像的增强可以减少光线退化引起的图像退化的影响，减少图像的颜色失真，提高图像的质量^[7]。为了增强图像中车牌区域与非车牌区域的亮暗对比度，本文采用非线性动态范围调整的方法进行图像增强，利用动态范围对人类视觉的影响特性，将动态范围进行压缩，扩大重要部分的变化范围，通过抑制非车牌区域信息来增强车牌区域信息。由于人眼接收图像信号到大脑成像过程中存在一个近似对数映射的过程，可采用对数映射来构建非线性动态范围调整的映射关系^[8]。非线性动态范围调整的计算公式如式(1)所示。

$$g(i,j) = c \cdot \log(1 + f(i,j)) \quad i = 1,2,\dots,m; j = 1,2,\dots,n \quad (1)$$

其中： $f(i,j)$ 为原始图像； $g(i,j)$ 为处理后的图像； c 为增益常数，用来调整曲线的弯曲程度。

将一副包含车牌的图像通过对数增强后的对比效果如图 2-1 所示，由图可看出，经过非线性动态范围调整方法进行图像增强后，提高了车牌图像的亮暗对比度，车牌区域与非车牌区域的差异更明显，突出了车牌区域。





图 2-1 图像增强对比效果图

2.2 图像去噪

通过摄像机采集到的车牌图片一般都因受某种外界因素干扰而存在噪声，而这些噪声点大部分都会影响图像的质量，因此要通过合理的方法对图像进行去噪处理。根据噪声对某一像素点的影响将使其灰度和邻点的灰度显著不同的特性，在尽可能保持原图信息的基础上，抑制噪声对图像质量的影响。

均值滤波和中值滤波是图像去噪的常用方法，均值滤波算法步骤简单，计算速度快，但在抑制噪声的同时会使图像变得模糊，中值滤波相比均值滤波算法略微复杂，但在图像清晰度的保持上优于均值滤波。综合以上优缺点的考虑，本文采取中值滤波的方法对图像进行去噪处理。中值滤波能够有效抑制噪声，是基于排序统计理论的一种非线性数字信号处理技术，利用一个含有奇数个点的滑动模板，对模板中的数据点进行排序，由于噪声点突变的特性，所以排序后噪声点将会在序列的两侧，最终选择排在序列中间位置的数据替代处于模板中心元素的像素值，以达到降低噪声的目的^[9]。

设 $f(x, y)$ 为原始图像； $g(x, y)$ 为中值滤波处理后的图像， W 为二维模板，中值滤波函数如式(2)所示：

$$g(x, y) = \text{Med}\{f(x - k, y - l), (k, l) \in W\} \quad (2)$$

假设原始矩阵 f ，变化后的矩阵为 g ，选择 3×3 的模板，边界元素不作处理，具体过程如下：

$$f = \begin{bmatrix} 1 & 2 & 1 & 4 & 3 \\ 1 & 10 & 2 & 3 & 4 \\ 5 & 2 & 6 & 8 & 8 \\ 5 & 5 & 7 & 0 & 8 \\ 5 & 6 & 7 & 8 & 9 \end{bmatrix} \quad g = \begin{bmatrix} 1 & 2 & 1 & 4 & 3 \\ 1 & 2 & 3 & 4 & 4 \\ 5 & 5 & 5 & 7 & 8 \\ 5 & 5 & 6 & 8 & 8 \\ 5 & 6 & 7 & 8 & 9 \end{bmatrix}$$

原始矩阵

中值滤波后矩阵

观察矩阵的变化结果可看出，像素值为 10 和 0 的点与周围像素值相差较大，是一个孤立的噪声点，通过中值滤波处理后，变成了 2 和 8，能够很

好地抑制噪声，并且尽可能保持边缘信息的完整。

将含有车牌的图像利用 OpenCV 的中值滤波函数 `medianBlur()` 处理后的对比效果图如图 2-2 所示。由图中可看出，车牌上的柳钉噪声点经过中值滤波处理后减弱了。



图 2-2 中值滤波对比效果图

2.3 本章小结

本章主要介绍了几种车牌图像预处理的方法，通过非线性动态范围调整方法进行图像增强后，提高了车牌图像的亮暗对比度，使车牌区域与非车牌区域对比更明显；通过中值滤波方法进行图像去噪后，去除了多余的噪声影响。通过图像预处理后，在一定程度上提高了车牌图像的质量，为车牌检测与识别奠定了基础。

3. 车牌检测

车牌检测是指对车牌图像进行车牌粗定位及后续处理与精确定位的过程，首先通过车牌定位方法对车牌区域粗定位，由于得到的车牌区域可能存在倾斜角度过大的情况，针对倾斜角度过大的车牌区域再进行倾斜校正，以得到最优状态的车牌图像块。经过一系列的后续处理操作后，得到候选车牌区域，再将候选车牌区域利用机器学习方法 SVM 模型进行分类，得到最终确定的车牌区域。

车牌检测与流程图如图 3-1 所示。



图 3-1 车牌检测流程图

3.1 车牌区域粗定位

在进行车牌检测时，首先要对车牌区域进行粗定位，目的是从复杂背景图像中定位出几个候选的车牌区域。目前常用的车牌定位方法是基于颜色和基于边缘检测，以及基于文字区域的定位方法^[9]，这三种方法都能在特地场景下对车牌区域进行定位。基于颜色的车牌定位方法能够利用车牌颜色对车牌进行定位，对倾斜的车牌也有较高的定位率，但在车牌颜色与车身颜色相近时就很难将车牌区域定位出来，因为闭操作处理会将车牌区域与车身区域连在一起，难以分辨出车牌区域。基于边缘检测的车牌定位方法可以检测出垂直边缘较多的区域，而车牌具有垂直边缘较多的特性，通过这种方法得到的垂直边缘数量较多的区域极有可能是车牌区域，但面对垂直边缘交错的情况下就无法准确地定位出车牌区域。基于文字区域的车牌定位方法能够定位出含有文字的区域，车牌是含有最多字符的文字区域，故能检测出车牌区域，但在车辆上贴有大量文字标签的情况下，这种文字区域的定位方法则效果不佳。

综合以上三种车牌定位方法的优缺点，本文采取一种基于颜色、基于 Sobel 边缘检测和基于文字区域的车牌定位方法相结合的方法，对车牌图像进行粗定位。首先通过基于颜色的车牌定位方法对车牌图像进行检测，对蓝色和黄色的车牌进行定位，为了避免车牌颜色与车身颜色相近造成的影响，在车牌检测时统计蓝色和黄色在图像中所占比例，若大于四分之一，则认为

车牌颜色特征失效。车牌颜色失效的情况下，再进行基于 Sobel 边缘检测的方法对车牌图像进行垂直边缘的检测，得到候选车牌区域。若上述步骤中未成功定位到车牌区域，说明基于 Sobel 边缘检测的方法失效，车牌图像中存在垂直边缘交错的情况，则进行基于文字区域的车牌定位方法对车牌图像进行检测，得到的候选车牌区域再次进行后续处理操作后输入到 SVM 模型中进行车牌区域精确定位。

车牌区域粗定位的流程图如图 3-2 所示。

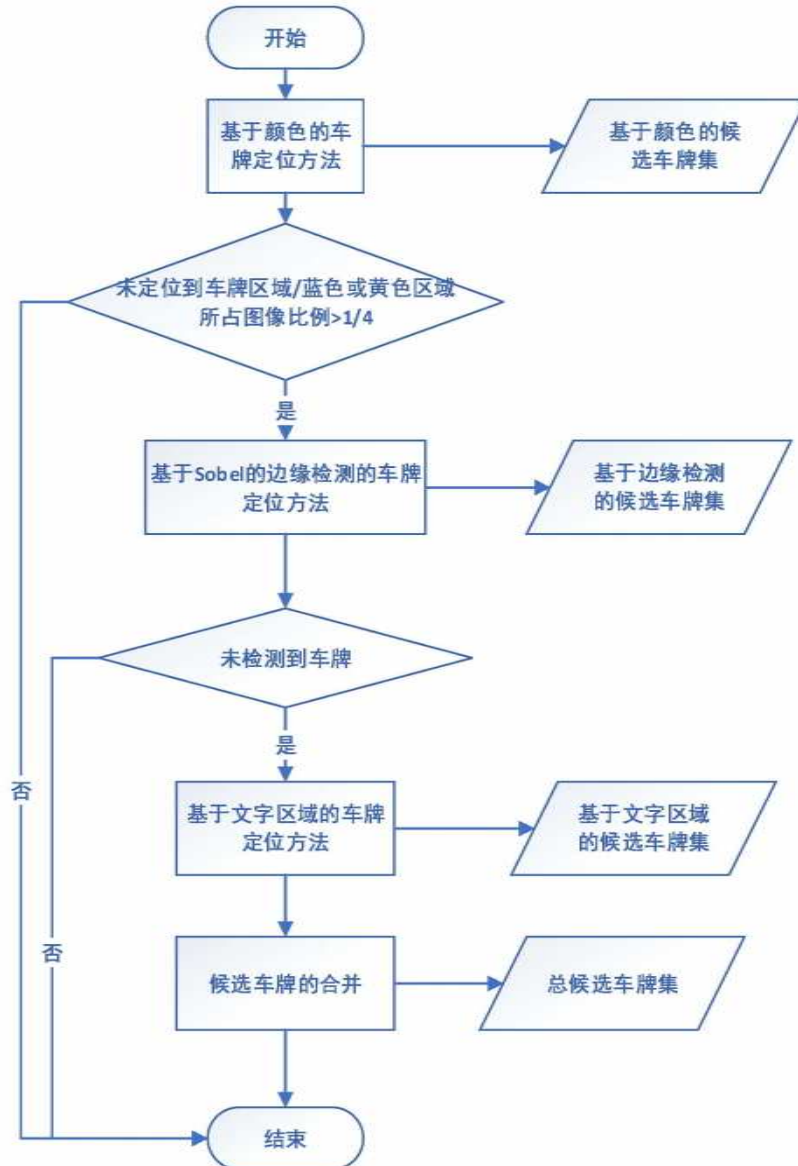


图 3-2 车牌区域粗定位流程图

3.1.1 基于颜色的车牌定位方法

我国大陆地区主要有四种车牌牌照，分别是民用黄底黑字牌照、民用蓝

底白字牌照、军警用白底黑字或红字牌照以及国外驻外机构用黑底白字牌照^[11]。由于我国车牌有 90%都是蓝色和黄色的，本文重点对蓝色车牌和黄色车牌进行研究，并且基于 HSV 色彩模型进行研究。基于颜色的车牌定位方法主要包括以下步骤：HSV 颜色空间转换、直方图均衡化、颜色模板匹配，其流程图如图 3-3 所示。

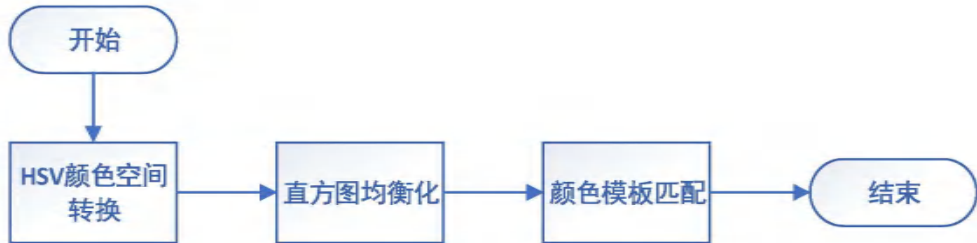


图 3-3 基于颜色的车牌定位方法流程图

(1) HSV 颜色空间转换

由于 RGB 在类似色彩中数值相差太大，且同一颜色容易收到不同光照情况的干扰，难以在复杂情况下保证图像的稳定，导致 RGB 难以作为颜色判断的标准。另一种颜色模型 HSV 模型非常适合解决颜色判断的问题，HSV 颜色空间是由色相(Hue)、饱和度(Saturation)和亮度(Value)3 个属性组成，与 RGB 颜色空间相比，HSV 可以给人们提供感知色彩的认知度，更适合用于图像处理^[12]。HSV 是坐标系下的一个倒置的六边形锥体，最下面的点为 0，表示黑色，最上面锥体底面的中心点呈白色，如图 3-4 所示。

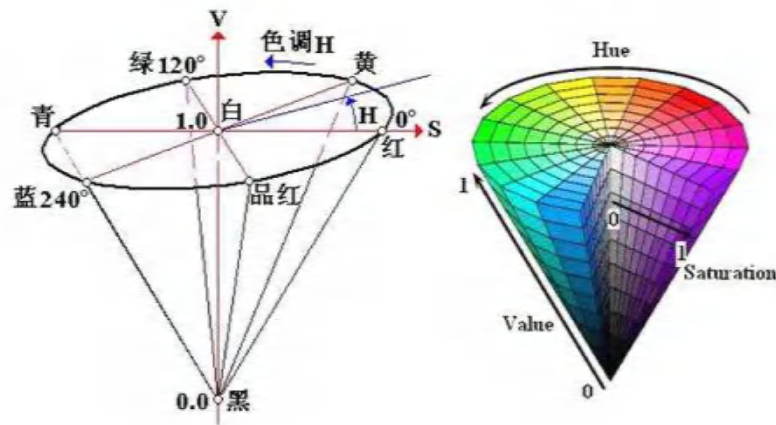


图 3-4 HSV 模型

将图像从 RGB 转换为 HSV，转换公式如 (3) 所示。

$$V = \max(R, G, B)$$

$$S = \begin{cases} (V - \min(R, G, B)) / V, & V > 0 \\ 0, & V = 0 \end{cases} \quad (3)$$

$$H = \begin{cases} \frac{60(G - B)}{SV}, & V = R \\ 60\left(2 + \frac{B - R}{SV}\right), & V = G \\ 60\left(4 + \frac{R - G}{SV}\right), & V = B \\ 0, & V = 0 \\ H + 360, & H < 0 \end{cases}$$

本文运用 OpenCV 中的函数 `cvtColor(InputArray src, OutputArray dst, int code, int dstCn=0)` 进行 HSV 颜色空间的转换，`src` 为原图像，`dst` 为目标图像，`code` 表示转换代码，这里取值为 `COLOR_BGR2HSV`，表示将图像转换为 HSV 图。彩色图像处理后的 RGB 与 HSV 的对比图如 3-5 所示。



图 3-5 HSV 颜色空间转换对比图

(2) 直方图均衡化

由于白天光照太强或者对面来车的情况，导致车牌区域光度较强，干扰了车牌定位的精度，为了降低这些噪声的影响，使用直方图均衡化的方法对转换后的 HSV 图进行处理。直方图均衡化方法是指将当前的灰度分布重新均衡地分布在整个灰度区间内，使图像清晰。

直方图均衡化的算法步骤如下：

① 求出原图 $f(i, j)$ 的灰度直方图，用 256 维的向量 h 表示， n_k 为原图像中灰度值为 k 的像素个数， $N = m \times n$ 为图像的像素总个数。则：

$$h(k) = p(k) = \frac{n_k}{N} \quad k = 0, 1, \dots, L - 1 \quad (4)$$

② 将公式(4)代入到公式(5)中计算，其中 $p(f_k) = h(k)$

设 $f(i, j)$ 为原始图像, $g(i, j)$ 为直方图均衡化处理后的图像

当 $f(i, j) = s$, 则:

$$g(i, j) = \begin{cases} (L-1) \cdot \sum_{k=0}^s p(f_k), & f(i, j) \neq 0 \\ 0, & f(i, j) = 0 \end{cases} \quad s = 0, 1, \dots, L-1 \quad (5)$$

基于直方图均衡化的算法原理, 本文在实现时先采用 OpenCV 的通道分离函数 `split()` 将 HSV 拆分为 H、S、V 三个通道, 再使用 `equalizeHist()` 函数对 V 通道进行直方图均衡化, 最后使用 `merge()` 函数将 H、S、V 三个通道进行合并成为 HSV 图像。直方图均衡化处理后的效果图如 3-6 所示。



图 3-6 直方图均衡化后结果图

(3) 颜色模板匹配

由 HSV 模型的特性可知, H 分量是 HSV 模型中与颜色本质相关的唯一分量, 只要固定 H 的值就能基本固定颜色。通过对调色板的摸索, 可以发现 H 分量在 200—280 之间, 且 S 和 V 的值在 0.35—1 之间时, 颜色表现为蓝色; H 分量在 30—80 之间, 且 S 和 V 的值在 0.35—1 之间时, 颜色表现为黄色; 因此, 可以根据 H 值建立蓝色和黄色的车牌匹配模板, 依次遍历图像中的所有像素, 在蓝色模板中, 将 H 分量在 200—280 之间且 S 和 V 的值在 0.35—1 之间的像素点置为 255, 其他置为 0; 在黄色模板中, 将 H 分量在 30—80 之间且 S 和 V 的值在 0.35—1 之间的像素点置为 255, 其他置为 0。经过上述操作处理后, 车牌图像已变成二值化图像, 车牌区域为白色, 非车牌区域为黑色, 效果如图 3-7 所示。



图 3-7 颜色模板匹配后结果图

3.1.2 基于 Sobel 的边缘检测的车牌定位方法

当车牌颜色与车身颜色相近时导致基于颜色的车牌定位方法失效，即在基于颜色的车牌定位方法中无法检测到车牌区域或者检测时统计的蓝色/黄色所占车牌图像的比例大于 0.25 时，可以采用基于 Sobel 的边缘检测的车牌定位方法对车牌图像进行进一步定位。

边缘是指灰度反差大的集合，反映图像的目标细节和边缘轮廓信息。从车牌特征上可知，车牌区域中包含丰富的垂直边缘特征，对车牌区域进行边缘检测，可以使图像的灰度反差增强，突出边缘和轮廓信息，从而将车牌区域的特征凸显出来。常用于处理不规则形状轮廓的边缘检测算子有 Roberts 算子、Sobel 算子、Prewitt 算子等^[13]。Roberts 算子边缘检测可能会出现半个像素的错位，Sobel 算子对加强目标边缘细节的效果较明显且简单高效，Prewitt 算子对处理灰度渐变和噪声比较多的图像友好，但易丢失边缘点。综合以上几种算子的优缺点以及车牌定位的实际情况，本文采取水平方向的 Sobel 算子进行边缘检测，其流程图如图 3-8 所示。



图 3-8 基于 Sobel 的边缘检测的车牌定位方法流程图

(1) 高斯模糊

高斯模糊是一种图像处理方法，其目的是对图像去噪，其本质上是一种数据平滑技术，对图像进行高斯模糊处理，使后续操作更容易检测出边缘信息。本文采用 OpenCV 函数 `GaussianBlur(src, src_blur, Size(m_GaussianBlurSize, m_GaussianBlurSize), 0, 0, BORDER_DEFAULT)` 对图像进行高斯模糊处理，其中 `m_GaussianBlurSize` 为高斯模糊半径，经过实验得到，高斯模糊半径选择 5 的效果最佳。车牌图像原图如图 3-9 所示，高斯模糊后的效果图如图 3-10 所示。



图 3-9 车牌图像原图



图 3-10 高斯模糊后的效果图

(2) 灰度化

由于图像处理的很多算法都只适用于灰度图像，且图像的色彩信息对边缘检测不能提供有效信息，所以对车牌图像进行灰度化，方便后续操作的进行。本文采用 OpenCV 的函数 `cvtColor(src_blur, src_gray, CV_RGB2GRAY)` 对高斯模糊后的图像进行灰度化，其灰度化处理的效果图如 3-11 所示。



图 3-11 灰度化后的效果图

(3) Sobel 算子

Sobel 算子本质上是对图像进行差分和滤波运算，通过对待处理像素的上、下、左、右四周相邻的像素进行灰度加权运算，突出图像边缘信息的同

时还对噪声有一定的平滑作用。本文采用 OpenCV 的函数 Sobel(InputArray src, OutputArray dst, int ddepth, int xorder, int yorder, int ksize, double scale, double delta, BORDER_DEFAULT)对图像进行边缘检测, xorder 取值为 1 表示对水平方向进行求导, 获得垂直边缘, yorder 取值为 0 表示不对垂直方向进行求导。Sobel 算子检测边缘后的效果图如图 3-12 所示, 由图可看出, 车牌区域含有大量的垂直边缘, 通过边缘检测后车牌区域凸显出来。



图 3-12 Sobel 算子检测边缘后的效果图

3.1.3 基于文字区域的车牌定位方法

当摄像机设备采集到的车牌图像处于夜间低光照的情况下时, 基于颜色和基于 Sobel 的边缘检测的车牌定位算法可能都定位不到车牌区域, 这是采取基于文字区域的车牌定位方法对文字区域进行检测, 以定位到车牌区域。

文字定位方法是采用低级过滤器提取文字后将其组合的一种定位方法, 常用的是最大极值稳定区域(MSER)算法, MSER 算法具有抑制噪声、消除背景影响、仿射无变化等优点, 本文也采用 MSER 算法进行文字区域的车牌定位。基于文字区域的车牌定位方法流程图如图 3-13 所示。

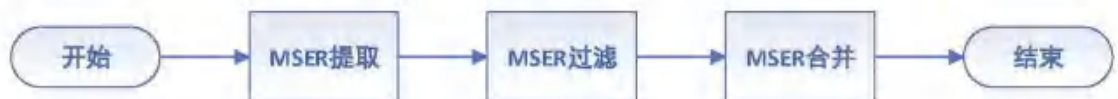


图 3-13 基于文字区域的车牌定位方法流程图

(1) MSER 提取

MSER 区域的提取, 首先要对图像中的所有像素进行排序, 对灰度图像进行二值化, 然后对二值图像求连通域, 最后获取最大稳定极值区域, 即为 MSER 区域^[14]。本文采用 OpenCV 中的 MSER 模块对车牌图像进行 MSER 提取, 获得最大稳定极值区域, MSER 提取的效果图如图 3-14 所示。



图 3-14 MSER 提取的效果图

(2) MSER 过滤

依据车牌字符的特性，对形成的 MSER 区域通过文字过滤器的筛选，剔除掉非车牌区域的字符，保留车牌字符区域，MSER 过滤的效果图如图 3-15 所示，七个绿色的矩形框表示过滤后保留的字符区域。



图 3-15 MSER 过滤的效果图

(3) MSER 合并

完成 MSER 过滤后，车牌区域的字符被分成单个字符区域，因为车牌的七个字符基本在同一条中心线上，对这七个字符区域进行合并可以还原出完整的车牌区域。由于 MSER 提取可能提取不出某些中文字符，若单个字符区域只有 6 个的话，则需增加一个滑动窗口来寻找缺失的中文字符，可通过对最左侧的字符矩形进行区域截取获得中文字符区域。MSER 合并的效果图如图 3-16 所示，红框表示合并后的区域。



图 3-16 MSER 合并的效果图

3.2 车牌区域粗定位的后续处理

在车牌区域粗定位的操作过程中可能会得到若干个候选车牌区域，通过对这若干个候选区域进行后续处理，达到进一步筛选的目的，尽可能地剔除非车牌区域，减少输入到 SVM 模型中的候选车牌区域的个数，从而提高车牌检测速率。后续处理的流程图如图 3-17 所示。

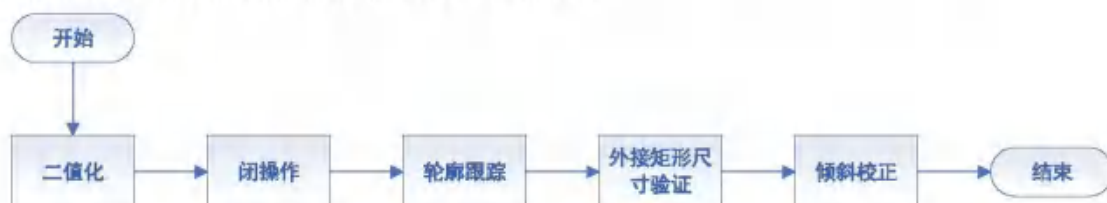


图 3-17 后续处理流程图

3.2.1 二值化

由于候选车牌区域可能存在提取的车牌区域存在伪车牌区域、车牌区域存在粘连或断裂等问题，对车牌图像进行二值化处理能够降低以上问题的影响，而且后续的闭操作与轮廓跟踪都是基于二值图像进行的，因此要先对图像进行二值化处理。本文采用 OpenCV 自带的二值化函数 `threshold(InputArray src, OutputArray dst, double thresh, double maxval, int type)` 进行处理，二值化的效果图如图 3-18 所示。



图 3-18 二值化效果图

3.2.2 闭操作

闭操作的目的是使车牌字母连接成为一个连通域，方便后续的轮廓跟踪操作。闭操作是指对图像先膨胀后腐蚀，在合并断裂车牌字母的同时，能够

基本保持车牌字母的大小。本文采取 OpenCV 的函数 `morphologyEx(img_threshold, img_threshold, MORPH_CLOSE, element)` 对图像进行闭操作处理，其中 `MORPH_CLOSE` 表示闭操作，闭操作的效果图如图 3-19 所示。

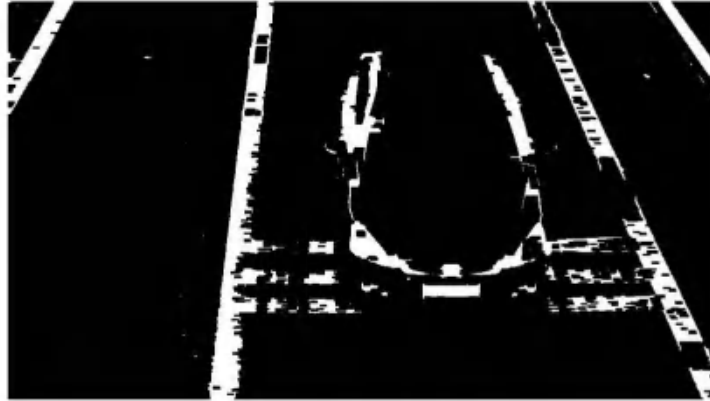


图 3-19 闭操作效果图

3.2.3 轮廓跟踪

对图像进行轮廓跟踪可将连通域的外围勾画出来，以便形成外接矩形。轮廓跟踪的算法较为简单，本文采用 OpenCV 的函数 `findContours(src, contours,`

`CV_RETR_EXTERNAL, CV_CHAIN_APPROX_NONE)` 进行轮廓跟踪处理，轮廓跟踪的效果图如图 3-20 所示，轮廓用红色标识出来。



图 3-20 轮廓跟踪效果图

3.2.4 外接矩形尺寸验证

在我国，车牌的形状规格有 $440\text{mm} \times 140\text{mm}$ 、 $440\text{mm} \times 220\text{mm}$ 、 $480\text{mm} \times 140\text{mm}$ 、 $220\text{mm} \times 140\text{mm}$ 四种^[15]。本文主要研究蓝色和黄色车牌汽车的自动识别，黄底

黑字的大型汽车号牌前车牌的尺寸为 $440\text{mm} \times 140\text{mm}$ ，后车牌尺寸为 $440\text{mm} \times 220\text{mm}$ ，蓝底白字的小型汽车号牌前后车牌的尺寸均为 $440\text{mm} \times 140\text{mm}$ ，车牌识别是对前车牌进行识别。因此，需要研究的车牌尺寸大小只有 $440\text{mm} \times 140\text{mm}$ ，宽高比为 3.14。所以，可以根据上一步轮廓跟踪中提取到的轮廓生成最小外接矩形，然后判断这个最小外接矩形的宽高比和面积是否都满足车牌的要求，达到进一步筛选车牌的目的。

本文首先遍历提取的每一个轮廓，用 OpenCV 函数 `minAreaRect(InputArray points)` 对每个轮廓求最小外接矩形，返回一个 `RotatedRect` 对象，将所有外接矩形用蓝色框标识后的效果图如图 3-21 所示。求得最小外接矩形后，设置一个偏差率 `error`，然后根据偏差率和车牌宽高比以及面积大小，求出最小宽高比和最大宽高比，以及最大面积和最小面积。然后遍历每一个外接矩形，只有矩形的宽高比和面积都分别在最小宽高比和最大宽高比、最小面积和最大面积之间，才被认定为是车牌区域，否则被判定为非车牌区域进行剔除，剔除后保留的外接矩形如图 3-22 所示，图中可看出只有六个外接矩形被保留。外接尺寸验证过程的流程图如图 3-23 所示。



图 3-21 求最小外接矩形效果图



图 3-22 外接矩形尺寸验证效果图

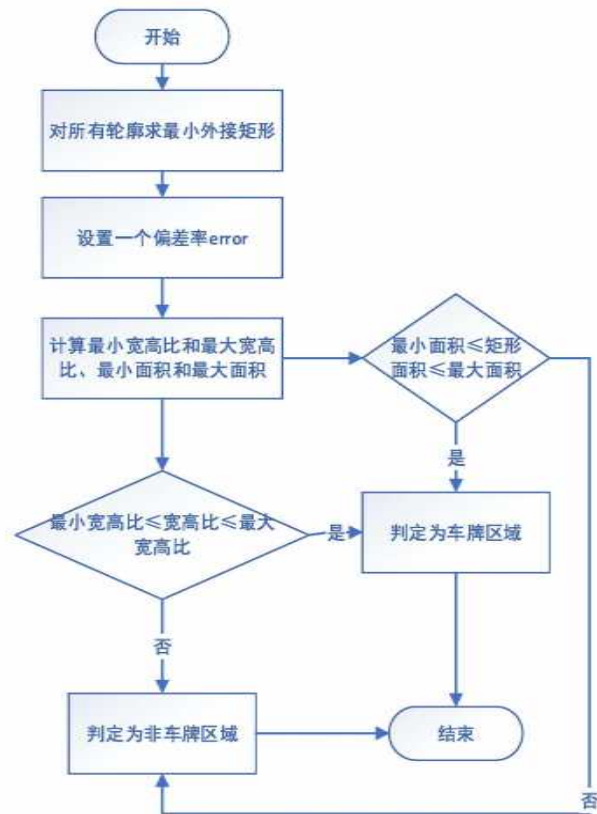


图 3-23 外接矩形尺寸验证流程图

3.2.5 倾斜偏斜校正

车牌图像在采集过程中由于摄像机设备倾斜抖动或者角度拍摄等原因，车牌图像中的车牌区域发生倾斜或偏斜现象，而倾斜和偏斜现象都会影响后续字符和分割的准确性，因此要对车牌区域进行校正。倾斜和偏斜的区别是视角不同，倾斜还是在正视角的范畴内，通过旋转即可校正，而偏斜是属于偏斜视角，旋转后的车牌区域呈平行四边形，处理步骤较为复杂，倾斜和偏斜车牌图像的对比图如图 3-24 所示。



图 3-24 倾斜和偏斜车牌图像的对比图

倾斜与偏斜校正的步骤如下：

(1) ROI 截取：首先对上一步操作中得到的外接矩形 RotatedRect 进行截取，截取时还需做安全区域判断，防止截取的 Rect 中的坐标越界，获得 ROI 图像。

(2) 角度判断：针对截取到的 ROI 图像进行角度判断，若角度在 $[-5, 5]$ 中，则认为为正车牌，无需进行校正操作；若角度在 $[-60, -5]$ 或者 $[5, 60]$ 中，则认为倾斜车牌或者为偏斜车牌，进行下一步偏斜判断。

(3) 偏斜判断：若 ROI 图像为矩形，则判定为倾斜车牌，利用 OpenCV 的仿射变换函数 `warpAffine()` 进行校正；若 ROI 图像为平行四边形，则为偏斜车牌。

(4) 扩大后旋转：若为偏斜车牌，在进行旋转时为了保证旋转后的图像不被截断，要先对截取的图像扩大 1.5 倍后再进行旋转，即扩大后旋转，旋转之后利用 OpenCV 函数 `getRectSubPix()` 进行第二次截取操作，将 RotatedRect 部分从 ROI 图像中截取出来。

(5) 仿射变换：由于经过旋转后，车牌图像区域呈现为平行四边形，截取后继续用 OpenCV 的仿射变换函数 `warpAffine()` 将平行四边形旋转为矩形。

(6) 归一化处理：最后将校正后的车牌进行归一化处理，将车牌统一为同一尺寸。

偏斜的具体操作如图 3-25 所示倾斜与偏斜校正的流程图如图 3-26 所示。

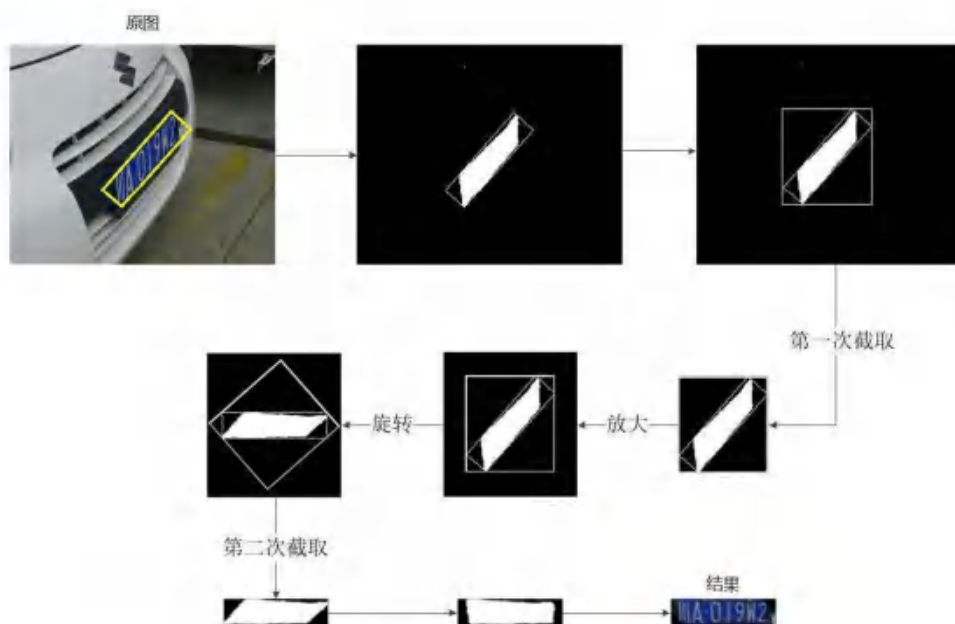


图 3-25 偏斜校正具体操作图

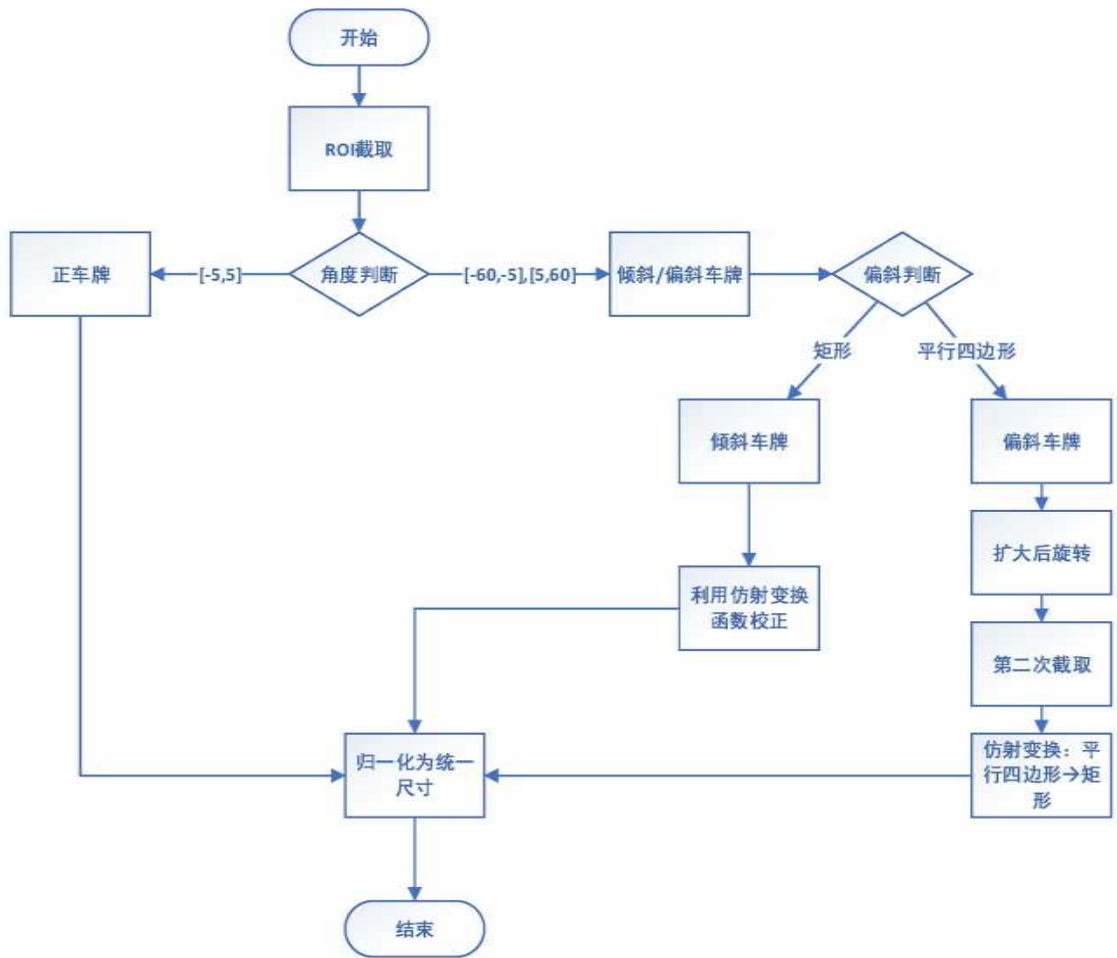


图 3-26 倾斜与偏斜校正流程图

3.3 车牌区域精确定位

经过车牌区域粗定位后，可以得到候选车牌区域集合，这些集合中包含正确的车牌区域和伪车牌区域。本文采用机器学习算法 SVM 对候选车牌区域集合进行分类，过训练 SVM 得到车牌分类模型，模型中只有“是车牌”和“不是车牌”两个类，将候选车牌区域集合输入到训练好的模型中，输出“是车牌”类别的列表。本文使用得是 OpenCV 提供的 CvSVM 类，CvSVM 类中已封装好 SVM 训练和预测方法以及自动测试调优方法，车牌区域精确定位示意图如图 3-27 所示。



图 3-27 车牌区域精确定位示意图

3.3.1 SVM 理论

SVM 属于线性分类器，它试图通过在两类样本中，寻找一种可以将其分开的最优分类平面来解决线性分类问题^[16]。如图 3-28 的线性分类示意图，图中的一条直线将平面上的两类点分割开，分隔超平面的形式可表示为 $w^T x + b$ ，SVM 就是求解一个最佳分隔超平面，最佳分隔超平面就是距离平面最近的点要有最大间隔。

假设超平面的形式为 $w^T x + b = 0$ ，以 $w^T x + b = 0$ 为分割线：当 $w^T x + b > 0$ 时，在这一点能使得 $w^T x + b = 1$ ；当 $w^T x + b < 0$ 时，在这一点能使得 $w^T x + b = -1$ ；根据点到直线的距离公式，求得点到分隔面的长度为 $\frac{|w^T x + b|}{\|w\|}$ ，在 $w^T x + b = 1$ 和 $w^T x + b = -1$ 上任取两点到分割面的距离之和为 $r = \frac{|w^T x + b|}{\|w\|} + \frac{|w^T x + b|}{\|w\|} = \frac{2}{\|w\|}$ 。要求得最佳分隔超平面就是在间隔 r 最大化的条件下求解 w 和 b 的值，即求 $\max \frac{2}{\|w\|}$ ，所以 SVM 的基本形态为： $\min \frac{1}{2} \|w\|^2$ 的 w 和 b 的值^[17]。以下为求解 SVM 的基本思路：采用拉格朗日乘子法转化为对偶问题。

(1) 引入拉格朗日乘子 $\alpha_i \geq 0$ ，得到拉格朗日函数：

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 + \sum_{i=1}^m \alpha_i (1 - y_i (w^T x_i + b)) \quad (6)$$

(2) 求 $L(w, b, \alpha)$ 的极值，即对 w 和 b 求偏导，并令其导数为零，对 w 求偏导得式(7)，对 b 求偏导得式(8)：

$$w = \sum_{i=1}^m \alpha_i y_i x_i \quad (7)$$

$$0 = \sum_{i=1}^m \alpha_i y_i \quad (8)$$

(3) 将求得的偏导代回原函数，得式(9)：

$$\max \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m \alpha_i \alpha_j y_i y_j x_i^T x_j \quad (9)$$

由上式可看出，约束函数得变量只有拉格朗日乘子 α ，求解出 α 后，再求出 w 和 b ，即可得到分隔线模型：

$$f(x) = \sum_{i=1}^m \alpha_i y_i x_i^T x_j + b \quad (10)$$

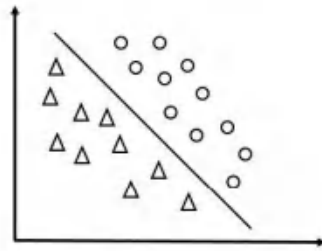


图 3-28 线性分类示意图

3.3.2 SVM 训练

在进行分类之前，要利用 SVM 训练出车牌模型供分类使用，SVM 训练步骤如下：

(1) 贴标签

对车牌区域图像进行贴标签处理，在车牌区域粗定位时会得到很多车牌区域与非车牌区域，将车牌区域图像与非车牌区域图像分别用文件夹 HasPlate 和 NoPlate 来存储，并编号，即完成贴标签处理。本文收集到 1000 多张车牌区域数据，部分如图 3-29 所示，2000 多张非车牌区域数据，部分如图 3-30 所示。



图 3-29 车牌训练数据



图 3-30 非车牌训练数据

(2) 数据分类

在训练之前，需要对数据进行分类，将车牌数据和非车牌数据分别取出 70% 作为训练数据，用于训练模型，其余 30% 作为测试数据，用于测试模型。

(3) 训练模型

首先加载待训练的车牌数据，并将车牌数据的类别标为 0，然后加载待

训练的非车牌数据,类别标为-1,再将两组数据进行合并,得到 trainingData 和 classes。合并之后,需要配置 SVM 模型的训练参数,训练时需要用到 CvSVMParams 对象,即配置 CvSVMParams 的对象中的参数。配置完毕后,则调用 CvSVM 类的训练函数 train (trainingData, classes, Mat (), Mat (), CvSVMParams params)^[18]。训练完后,可以将训练好的模型存入 plate_svm.xml 文件中,使用时可直接加载 plate_svm.xml 文件,节省训练模型的时间。

(4) 测试模型

将分出来的 30%测试数据输入到上一步训练出来的模型中,看输出的预测标签与实际标签的差别,然后根据准确率指标来评判模型的优劣。

(5) 参数调优

在训练模型时需要配置参数,而参数的设置直接影响到模型的优劣,所以参数要进行调优以训练出最佳模型。OpenCV 提供了一个自动训练 SVM 的方法 train_auto(), 在训练时可以不断改变参数,然后训练模型,测试模型,然后选择模型效果最好的参数。

SVM 训练的流程图如图 3-31 所示。

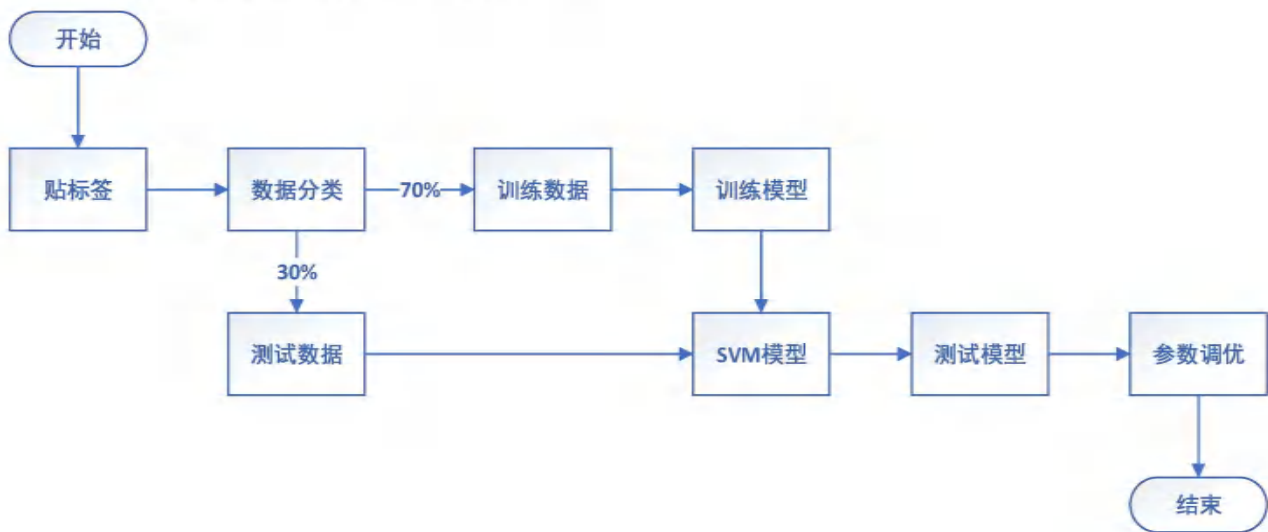


图 3-31 SVM 训练流程图

3.3.3 车牌分类

SVM 训练完毕后,将候选车牌区域集合输入到训练好的车牌模型中,然后调用 OpenCV 的 CvSVM 类中的预测函数 predict(),对候选车牌区域集合进行分类,剔除非车牌区域,输出正确的车牌区域,完成车牌区域的精确定位。本文对车牌区域粗定位得到的候选车牌区域集合进行 SVM 精确定位后的效果

图如图 3-32 所示，图中可看出，候选车牌区域集合中有 5 个数据，经过 SVM 精确定位后，输出了正确的车牌图像。



图 3-32 SVM 精确定位效果图

3.4 本章小结

本章介绍了车牌检测的过程，包括车牌区域粗定位及后续处理、车牌区域精确定位三大过程。车牌区域粗定位采用的是一种综合基于颜色、基于 Sobel 的边缘检测、基于文字区域的车牌定位方法。对粗定位得到的车牌区域进行二值化、闭操作、轮廓跟踪、外接矩形尺寸验证、倾斜偏斜校正一系列操作，以达到进一步筛选车牌区域的目的。后续处理得到截取的候选车牌集合，将候选车牌集合输入到 SVM 模型中进行精确定位，得到正确的车牌块。

4. 字符分割

通过车牌检测的步骤得到了一个完整的车牌，在进行字符识别之前，要先对车牌中的字符进行提取和分割，即字符分割，得到单个的字符块才能进行字符识别，因此，字符分割的效果也直接影响到字符识别的正确率。

4.1 常用的字符分割方法

常用的字符分割方法有基于聚类连通区域的字符分割方法、基于模板匹配的字符分割方法、基于垂直投影的字符分割方法^[19]。

基于聚类连通区域的字符分割方法：根据同一字符的像素点通常能构成一个连通区域的特性，将字符像素联合成一个联通区域，然后结合字符的固定高度和比例进行字符的分割。该方法能够对复杂背景下的车牌字符进行分割，鲁棒性较高，但算法复杂，计算量非常大，不利于操作。

基于模板匹配的字符分割方法：根据车牌的位置和几何特性设计出符合车牌特性的模板，且模板具有一定字符宽度和间隙，然后移动模板，通过比较像素的差值和比值进行分割。该方法能够对字符断裂、受损严重等情况的车牌进行处理，但算法复杂，对模板设计的准确性要求高。

基于投影法的字符分割方法：根据车牌字符大小和车牌字符间隙大小固定的特性，对车牌区域进行边缘检测处理，对其水平方向求导获得垂直投影，垂直投影可确定字符的左右边界，对其垂直方向求导获得水平投影，水平投影可确定字符的上下边界，根据字符的上下周有边界可将字符分割出来。该算法可以解决字符粘连的问题，也可以有效地去除噪声，但不能很好地对汉字进行处理。

综合以上方法的优缺点可知，三种算法比较复杂，效率较低，考虑到各方面因素，本文采用基于字符轮廓提取的分割方法对字符进行分割，该算法简单，效率更高。

4.2 基于字符轮廓提取的分割方法

基于字符轮廓提取的分割方法是根据字符颜色与车牌背景颜色有较大差异性来提取字符的轮廓，从而进行字符的分割，但对中文字符进行提取轮廓时容易出现断裂、粘连的情况，所以在处理中文字符时需进行特殊处理。基于字符轮廓提取的分割方法流程图如图 4-1 所示。

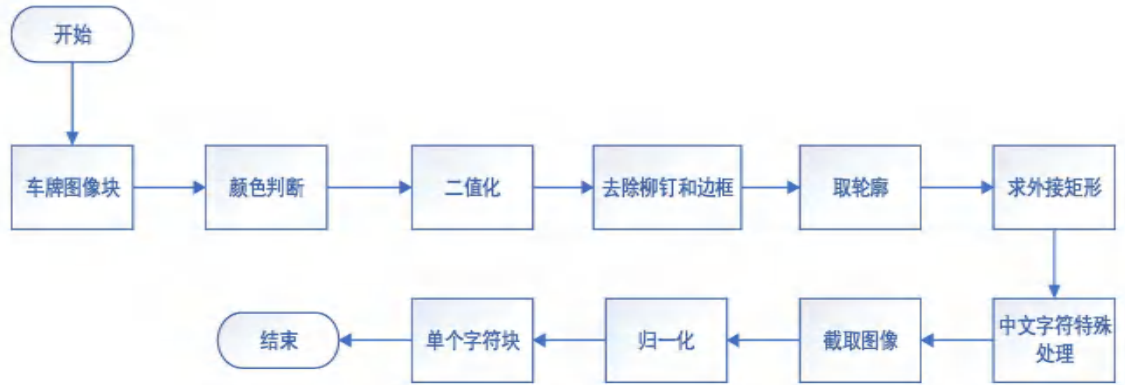


图 4-1 基于字符轮廓提取的分割方法流程图

(1) 二值化：本文采用 OpenCV 自带的二值化函数 `threshold(InputArray src, OutputArray dst, double thresh, double maxval, int type)` 进行处理，进行二值化操作时要对蓝色和黄色的车牌作不同的处理，首先利用上述章节中的颜色定位方法中的颜色模板匹配进行颜色判断，若为蓝色车牌，则 `type` 取值为 `CV_THRESH_BINARY`，若为黄色，则 `type` 取值为 `CV_THRESH_BINARY_INV`。蓝色车牌二值化效果对比图如图 4-2 所示，黄色车牌二值化效果对比图如图 4-3 所示。



图 4-2 蓝色车牌二值化效果对比图



图 4-3 黄色车牌二值化效果对比图

(2) 去除柳钉和边框：由图 35 和图 36 可看出，在二值化的过程中，由于柳钉和边框的影响，给车牌图像带来了一定的噪声，会对后续提取轮廓造成不良影响。在处理过程中，根据柳钉和边框所在行数的跳变数量来去除柳钉和边框，遍历车牌中每一行的像素，然后统计每行图像的黑白跳变点次数，认为跳变点次数非常少的行为柳钉或边框所在行，再将该行变为背景黑色^[20]。将图 4-2 中的蓝色车牌去除柳钉和边框后的图如图 4-4 所示，图中可看出，柳钉和边框对车牌图像的影响已经大大减弱。

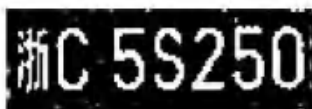


图 4-4 去除柳钉和边框的效果图

(3) 取轮廓和求外接矩形: 取轮廓和求外接矩形两个操作的方法均和上述章节中后续处理方法中一样, 这里不作详细介绍, 采用 OpenCV 的函数 `findContours()` 进行轮廓跟踪处理, 用 OpenCV 函数 `minAreaRect(InputArray points)` 对每个轮廓求最小外接矩形。取轮廓效果图如图 4-5 所示, 求外接矩形效果图如图 4-6 所示。



图 4-5 取轮廓效果图



图 4-6 求外接矩形效果图

(4) 中文字符特殊处理: 提取轮廓时, 由于中文字符笔画的特性, 容易出现断裂现象, 导致不能完整地分割, 所以在处理中文字符时不能再用取轮廓法, 而是要采取特殊的处理方法。通过观察车牌可得到, 车牌中第一个中文字符和第二个英文字符相隔较近, 而第二个英文字符和第三个英文字符距离较远, 可以根据这一特性对中文字符进行提取。由图 4-7 可知, 车牌前两个字符恰好在整个车牌长度的 $2/7$ 处, 则可以对取得的六个英文字符的外接矩形的中点位置进行判断, 若中点在车牌区域的 $1/7-2/7$ 处, 则认为是第二个英文字符的外接矩形, 根据该矩形的位置向左偏移即可提取到中文字符的外接矩形。



图 4-7 车牌区域长度示意图

(5) 截取图块与归一化: 最后对定位的矩形框进行截取, 截取得到单一的字符块, 再将字符块进行归一化, 以便进行下一步字符识别处理, 字符分割后的最终效果如图 4-8 所示。



图 4-8 字符分割效果图

4.3 本章小结

本章主要研究字符分割算法，对检测出来的车牌图像块进行提取分割字符，首先介绍了三种常用的字符分割算法：基于聚类连通区域的字符分割方法、基于模板匹配的字符分割方法、基于垂直投影的字符分割方法，并分析了这三种算法的优缺点。然后重点介绍了本文采取的基于字符轮廓提取的分割方法，通过颜色判断、二值化、去除柳钉和边框、取轮廓、求外接矩形、中文字符特殊处理、截取图像、归一化这一系列步骤，得到单一的字符块，用于输入字符识别模块中。

5. 字符识别

通过字符分割后可以得到单一的字符块，字符识别是将分割出来的字符图像转换为文本信息，输出车牌号码，是车牌识别的最后一个步骤也是重要组成部分，其识别准确度直接影响车牌自动识别系统的准确度。

5.1 常用的字符识别方法

常用的字符识别方法有基于模板匹配的字符识别算法、基于 SVM 的字符识别算法、基于神经网络的字符识别算法^[19]。

基于模板匹配的字符识别算法：利用字符的特征和匹配原理对字符图像进行识别，通过计算车牌字符和样本模板之间的相似度，用样本模板的类别来预测车牌字符的类别。该算法对于相似度较高的字符识别准确率较低，且图像的粘连和断裂情况都会造成影响。

基于 SVM 的字符识别算法：运用基于统计理论的 SVM 分类器进行分类，计算速度快，能够解决小样本的识别问题。

基于神经网络的字符识别算法：运用神经网络模型进行分类，神经网络具有鲁棒性强、分类能力强等优点，对字符识别的准确率也比较高。

综合以上三种字符识别算法的优缺点以及自身项目的特点，本文识别的是中国车牌，中国车牌中含有汉字比其他字符要复杂，所以选取基于神经网络的字符识别算法，对中文车牌的识别更加友好，且识别率较高。

5.2 基于神经网络的字符识别方法

5.2.1 神经网络概述

人工神经网络(ANN)是一种模拟人脑的神经网络进行学习并获得相应的知识与技能的机器学习技术^[21]。人工神经网络包括前馈神经网络和反馈神经网络，前馈神经网络又包括单层神经网络和多层神经网络。人工神经网络是由大量神经元结合而成的，通过权值的调整能够实现对信息的处理。人工神经元模型主要由连接权、加法器、激活函数三部分组成^[22]，其模型图如图 5-1 所示，用数学公式可表示为式(11)， x_i 为输入， w_i 为权值， $f(x)$ 为激活函数， y 为输出。

$$y = f\left(\sum x_i w_i\right) \quad (11)$$

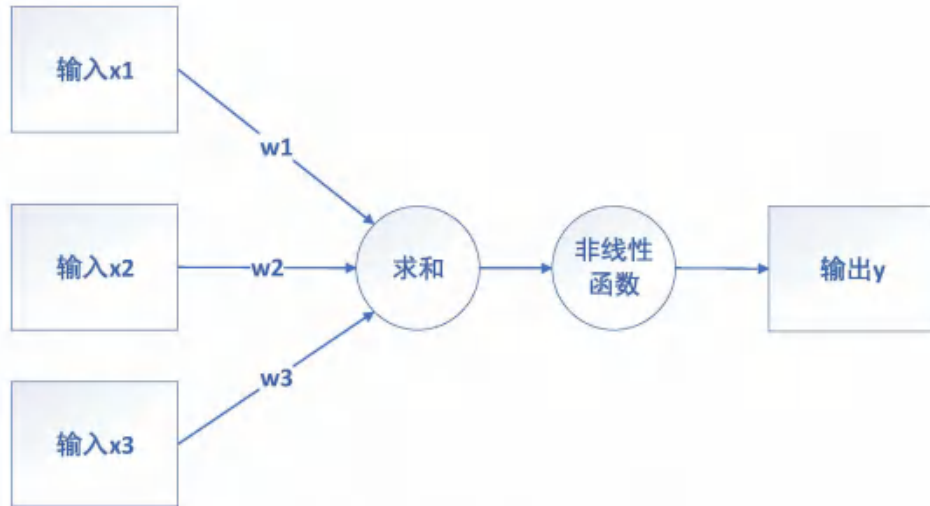


图 5-1 人工神经元模型图

5.2.2 BP 神经网络在车牌字符识别的应用

BP 神经网络就是使用了反向传播 BP 算法的多层前馈神经网络，主要分为正向传播过程与误差反向传播过程，BP 神经网络主要由三层的网络组成，包括输入层、隐含层、输出层，前一层的输出作为后一层的输入使其不断向后计算，其结构图如图 5-2 所示。在计算过程中，由于误差的存在，需要对误差进行校正，而对误差的校正是反向进行的，这样可以逐层导出隐含层的误差来优化神经网络各层的权值^[23]。随着正向传播和误差反向传播过程的进行，可完成对样本的训练，并使 BP 神经网络识别字符的准确率高和鲁棒性强。

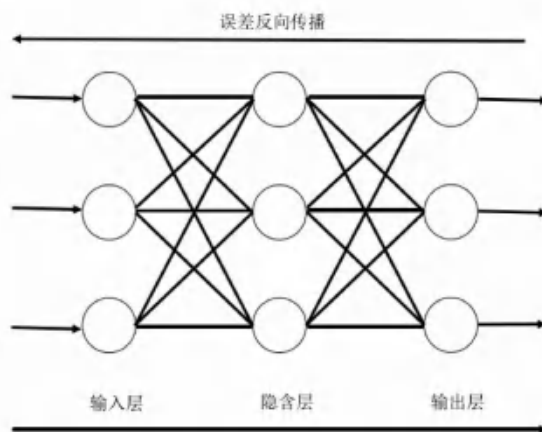


图 5-2 BP 神经网络结构图

基于 BP 神经网络的字符识别主要包含样本采集、神经网络的训练、特征提取、字符识别，其流程图如图 5-3 所示。

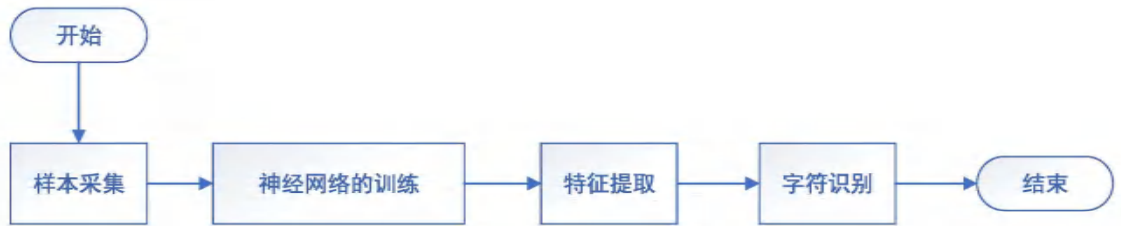


图 5-3 基于 BP 神经网络的字符识别流程图

(1) 样本采集：采集大量的车牌字符图片，并对其进行分类和编号。由于我国的车牌号码大多是由七个字符组成，本文主要研究常见汽车的车牌识别，不考虑特殊车辆，所以在采集样本时要采集 31 个汉字、10 个数字（0—9）、24 个字母（I 与 O 排除）共 65 个类别，采集分类后的数据如图 5-4 所示。

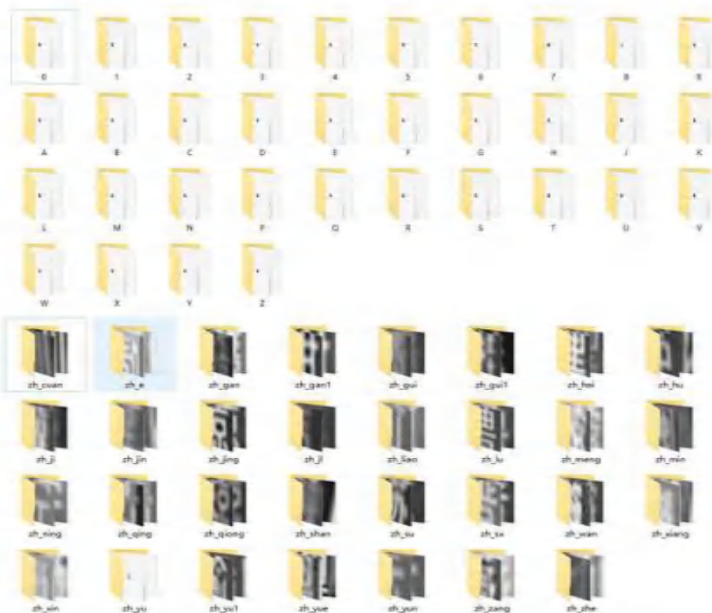


图 5-4 样本数据图

(2) 神经网络的训练：由于 BP 神经网络是一个多层感知网络(MLP)，本文使用 OpenCV 提供的神经网络类 CvANN_MLP 进行神经网络的训练，激活函数采用 sigmoid 激活函数(CvANN_MLP::SIGMOID_SYM)，其流程图如图 5-5 所示。

① 设置参数：OpenCV 神经网络参数的类型存放在 CvANN_MLP_TrainParams 类中，需要设置以下参数：迭代次数(CV_TERMCRIT_ITER)、误差最小值(CV_TERMCRIT_EPS)、训练方法(train_method, BP 神经网络采用的是反向传播算法 BACKPROP)、权值更新率(bp_dw_scale)、权值更新冲量(bp_moment_scale)。

②提取样本图片的特征：对样本图片提取投影直方图特征和梯度直方图

特征。

③训练：对采集好的样本图片依次读入其特征向量，并对每一类别的特征进行编号，将特征存储到 OpenCV 的输入矩阵中，然后调用 `CvANN_MLP::train` 函数进行训练，保存训练好的权值矩阵到训练 `train_ann.xml` 文件中。

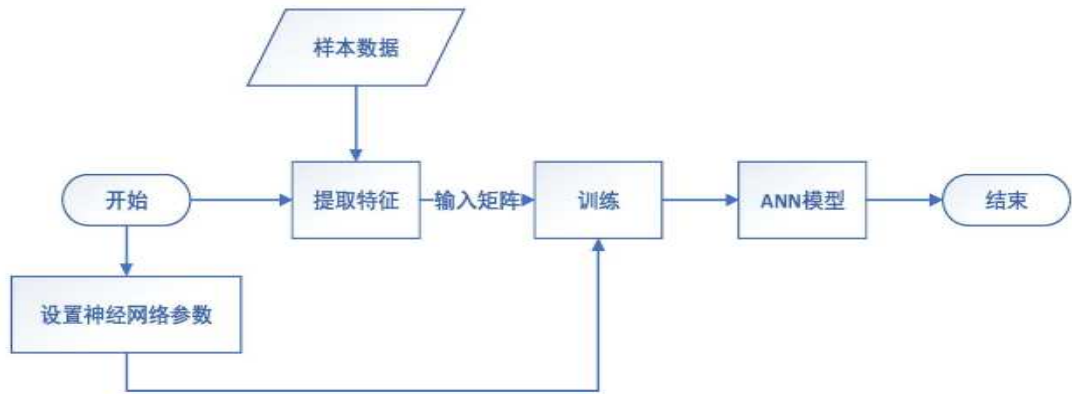


图 5-5 神经网络训练流程图

(4) 特征提取：提取投影直方图特征和梯度直方图特征。

(5) 字符识别：将分割出来的字符图片提取到的特征依次输入训练好的神经网络，调用 `CvANN_MLP::predict` 函数对字符进行识别，得到字符识别结果，组合在一起即为车牌号码。

5.3 本章小结

本章是基于 OpenCV 的车牌自动识别算法的最后一部分，主要研究字符识别算法，介绍了三种基于模板匹配的字符识别算法、基于 SVM 的字符识别算法、基于神经网络的字符识别算法。其中，重点分析基于神经网络的字符识别算法，首先对人工神经网络进行概述，然后阐述了 BP 神经网络在车牌字符识别的应用，主要包括样本采集、神经网络训练、特征提取和字符识别等步骤，通过字符识别后，将识别后的结果组合在一起得到车牌号码。

6. 车牌自动识别系统在 Qt 上的设计与实现

车牌自动识别系统能够将车牌从复杂背景中检测提取出来，并进行字符分割和识别得到其车牌号码，其总体框架如图 6-1 所示。车牌自动识别系统主要有 5 个模块：图像预处理模块、车牌检测模块、字符分割模块、字符识别模块及结果显示模块，如图 6-2 所示。其中，对图像的一系列处理操作都是基于 OpenCV 计算机视觉库，结果显示模块采用 Qt 开发平台完成车牌自动识别系统的 GUI 设计。本系统是在 Win10 下，qt-mingw492-5.5.1 集成开发环境中，基于 OpenCV 3.2.0 版本，采用 C++ 编程语言实现的。

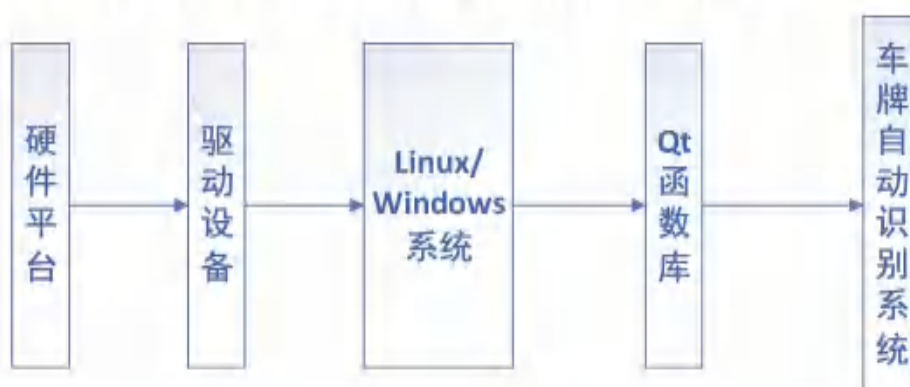


图 6-1 系统总体框架图

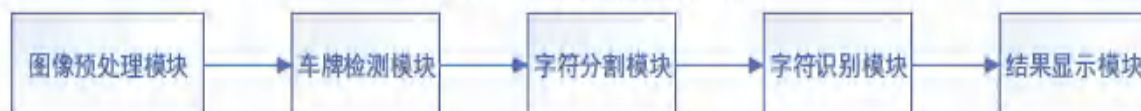


图 6-2 系统模块图

6.1 OpenCV 简介

OpenCV (Open Source Computer Vision Library), 是 Intel 公司在 1999 年建立的开源的跨平台计算机视觉库。OpenCV 可以运行在 Windows、Linux 和 Mac OS 操作系统上，由一系列 C 函数和少量 C++ 组成，同时提供了 Python、Matlab、Ruby 等语言接口，实现了图像处理和计算机视觉方面的很多通用算法^[24]。OpenCV 具有良好的平台无关性，以及强大的图像处理和矩阵处理能力，为用户提供方便灵活的接口。OpenCV 主要包含四大模块：CV 模块（图像处理和计算机视觉算法）、ML 模块（机器学习库）、HighGUI 模块（图像和视频的输入输出）、CXCORE 模块（基本数据结构、XML 支持、绘图函数）。

6.2 Qt 图形化界面的设计与实现

Qt 是一个跨平台 C++ 图形用户界面应用程序开发框架。由于 Qt 具移植性强、运行速度快、便捷易用等优点，本文采取 Qt 作为系统的图形化界面的开发平台。

本系统共设计了 2 个界面：登录界面、车牌识别界面。登录界面可供用户输入用户名和密码进行登录，登录成功后才能使用车牌识别功能，登录界面设置了按钮动画效果，登录界面图如图 6-3 所示。车牌识别界面是系统的主体部分，主要包含上传图像区、显示图像区、显示定位结果区、显示识别结果区四个部分，车牌识别界面还设置了小车动画效果，如图 6-4 所示。在车牌识别界面中，点击 Add Image 按钮可选择车牌图片上传并显示图片至显示图像区，车牌自动识别系统会返回车牌截取图片和车牌识别结果，分别显示在显示定位结果区、显示识别结果区中。



图 6-3 登录界面图



图 6-4 车牌识别界面图

6.3 实验

6.3.1 评估指标

本文采用识别正确率^[25]作为实验的评估指标，识别正确率表示在车牌识别中正确识别的车牌图片数占总车牌图片数的百分比，计算公式如式(12)：

$$\text{识别正确率} = \frac{\text{正确识别的车牌图片数}}{\text{总车牌图片数}} \quad (12)$$

6.3.2 实验方案

本文共进行了 4 组实验，具体实验方案如下：

(1) 实验一：对综合场景下的车牌图片进行识别。在实验中采集了 1000 多张不同场景下的车牌图片，其中包括不同距离、不同角度、不同光照下拍摄的蓝色和黄色车牌图片，将这 1000 多张图片输入到车牌自动识别系统中，然后统计其识别正确率。

(2) 实验二：针对不同距离下的车牌图片进行识别。在实验中采集了距离近距离(1m—5m)拍摄的车牌图片 200 多张，远距离(高速公路上拍摄)的车牌图片 200 多张，且图片的角度和光照条件良好。将这两组图片分别输入到车牌自动识别系统中，然后统计其识别正确率并进行比较分析。

(3) 实验三：针对不同视角下的车牌图片进行识别。在实验中采集了正视角拍摄的车牌图片 200 多张，偏斜视角拍摄的车牌图片 100 多张，且图片的距离和光照条件良好。将这两组图片分别输入到车牌自动识别系统中，然后统计其识别正确率并进行比较分析。

(4) 实验四：针对不同光照条件的车牌图片进行识别。在实验中采集了较亮的车牌图片 100 多张，正常光照的车牌图片 200 多张，较暗的车牌图片 100 多张，且图片的距离和角度良好。将这三组图片分别输入到车牌自动识别系统中，然后统计其识别正确率并进行比较分析。

6.3.3 实验结果与分析

(1) 实验一：统计结果见表 6-1，正常的车牌图片识别效果如图 6-5 所示，距离较远的车牌图片识别效果如图 6-6 所示，偏斜视角下的车牌图片识

别效果如图 6-7 所示，白天光照较亮的车牌图片识别效果如图 6-8 所示，夜晚光照较亮的车牌图片识别效果如图 6-9 所示，光照较暗的车牌图片识别效果如图 6-10 所示。

表 6-1 实验一统计结果

正确识别数	883
总图片数	1038
识别正确率	85.1%



图 6-5 正常车牌识别效果图



图 6-6 远距离车牌识别效果图



图 6-7 偏斜视角车牌识别效果图



图 6-8 白天光照较亮车牌识别效果图



图 6-9 夜间光照较亮车牌识别效果图



图 6-10 光照较暗车牌识别效果图

(2) 实验二：统计结果如表 6-2。

表 6-2 实验二统计结果

距离	0—5m	远距离（高速公路拍摄）
正确识别数	230	183
总图片数	247	208
正确识别率	93.1%	87.9%

(3) 实验三：统计结果见表 6-3。

表 6-3 实验三统计结果

视角	正视角	偏斜视角
正确识别数	239	164
总图片数	256	197
正确识别率	93.4%	83.2%

(4) 实验四：统计结果见表 6-4。

表 6-4 实验四统计结果

视角	较亮	正常	较暗
正确识别数	160	239	149
总图片	176	200	184

数			
正确识别率	90.1%	92.5%	80.9%

从以上四组不同的实验中可看出，单一正常场景的正确识别率可达到 92.2%，综合复杂场景的正确识别率为 85.1%，而在光照较暗和偏斜视角的情况下识别率较低，与单一正常场景相差 10.2%。

6.4 本章小结

本章主要研究了车牌自动识别系统在 Qt 上的设计与实现，介绍了系统的整体设计架构，并对 OpenCV 进行了简介，重点介绍了 Qt 图形化界面的设计与实现，并且进行了四组不同的实验来评估系统识别的准确率和鲁棒性。

7.总结与展望

随着机动车数量的不断增长，智能交通越来越普遍，车牌识别在智能交通中起着重要作用。本文对基于 OpenCV 的车牌自动识别系统展开研究。本文首先从背景和意义，以及国内外研究现状对车牌识别技术进行分析，然后从车牌图像预处理、车牌图像检测、字符分割、字符识别四部分展开研究，最后介绍了车牌识别算法在 Qt 平台上的设计与实现，还进行相关实验与结果分析。

在车牌图像预处理模块中，本文采用非线性动态范围调整的方法进行图像增强，采用中值滤波对图像进行去噪处理。在车牌检测模块中，采用一种基于颜色的车牌定位方法、基于 Sobel 的边缘检测的车牌定位方法、基于文字区域的车牌定位方法相结合的算法进行车牌区域粗定位，获得若干个候选车牌区域。将这若干个候选车牌区域进行一系列的二值化、闭操作、轮廓跟踪、外接矩形尺寸验证、倾斜偏斜校正的后续处理操作以到达进一步筛选车牌的目的。再将得到的候选车牌集合输入到训练好的 SVM 模型中进行车牌区域精确定位。在字符分割模块中，采用基于字符轮廓提取的分割方法进行字符分割，通过颜色判断、二值化、去除柳钉和边框、取轮廓、求外接矩形、中文字符特殊处理、截取图像、归一化这一系列步骤，得到单一的字符块，用于输入字符识别模块中。在提取过程中，由于中文字符的特殊性，需要借助第二个英文字符的位置对中文字符进行提取。在字符识别模块中，采用基于 BP 神经网络的车牌字符识别方法进行字符识别，主要包括样本采集、神经网络训练、特征提取和字符识别等步骤，通过字符识别后，将识别后的结果组合得到车牌号码。最后将车牌识别算法移植到 Qt 平台上通过图形化界面进行实现，方便用户操作。

经过实验表明，本文的基于 OpenCV 的车牌自动识别系统能够适应多种交通场景，能够抵抗一定的干扰，对车牌图形进行识别获取其车牌号码，在单一正常场景下的车牌识别正确率能达到 90%以上，在综合复杂场景下的车牌识别率约为 85%。

同时，本文还存在不足之处：本文主要对常见车辆进行研究，只能识别蓝色和黄色车牌，对军用车辆、新能源车辆以及一些特殊车牌的车辆不能处理；本文在 SVM 和神经网络的训练中，大多都是采取手工贴便签的方式，工程量大且需耗费大量时间，仍待改进，今后在车牌识别部分应向深度学习方向发展，借助卷积神经网络 CNN 来解决问题；本文在神经网络训练时，提取的特征向量较简单，在后续工作中可以尝试提取其他特征向量进行训练，以提高车牌的识别正确率。

参考文献

- [1]陈学保. 车牌字符识别算法的研究[D]. 重庆:重庆大学, 2013.
- [2]杨思源. 基于 OPENCV 的车辆牌照识别系统研究[D]. 西安. 西安电子科技大学, 2013.
- [3]莫玲, 麦康机. 基于机器视觉的车牌识别系统设计[J]. 机电工程技术, 2018, 47(11):112-116.
- [4]Wang Yi. Research on License Plate Recognition Technology in ETC System[A]. Institute of Management Science and Industrial Engineering. Proceedings of 2018 5th International Conference on Electrical & Electronics Engineering and Computer Science(ICEECS 2018)[C]. Institute of Management Science and Industrial Engineering: 计算机. 科学与电子技术国际学会 (Computer Science and Electronic Technology International Society), 2018:6.
- [5]李智超. 基于神经网络车牌图像识别系统[D]. 吉林大学, 2018.
- [6]刘智勇, 刘迎建. 车牌识别(LPR)中的图像提取及分割[J]. 中文信息学报, 2000(04):29-34.
- [7]张庆. 车牌识别算法及其在 QT 平台上的实现[D]. 北方工业大学, 2017.
- [8]朱虹. 数字图像处理基础与应用[M]. 北京:清华大学出版社, 2013.
- [9]刘国兵. 车牌识别系统的研究与设计[D]. 中国石油大学(华东), 2016.
- [10]孙成磊. 复杂场景下基于 OpenCV 的车牌识别算法研究[D]. 曲阜师范大学, 2017.
- [11]朱梦哲, 陈志华, 赵钟, 尤越. 基于 OpenCV 的车牌定位和校正方法[J]. 计算机应用, 2013, 33(S1):223-226+235.
- [12]师晶晶. HSV 色彩空间下三边滤波的 Retinex 图像增强算法研究[J]. 自动化与仪器仪表, 2018(12):12-14.
- [13]暨彩明. 车牌识别算法设计及系统实现[D]. 哈尔滨工业大学, 2016.
- [14]王艳, 谢广苏, 沈晓宇. 一种基于 MSER 和 SWT 的新型车牌检测识别方法研究[J]. 计量学报, 2019, 40(01):82-90.
- [15]毕波, 邵永谦, 孙冬军, 贾思超. 基于 OpenCV 的车牌识别[J]. 电子设计工程, 2019, 27(01):37-41.
- [16]李星, 郭晓松, 郭君斌. 基于 HOG 特征和 SVM 的前向车牌识别方法[J]. 计算机科学, 2013, 40(11A):329-332.
- [17]Peter Harrington. 机器学习实战[M]. 北京:人民邮电出版社, 2013.
- [18]吕昆, 金珣宏. 一种基于 OpenCV 的车牌识别方法[J]. 软件导

刊, 2016, 15(11):193-195.

- [19] 杨莉. 基于 OpenCV 的车牌识别系统研究与实现[D]. 湖南大学, 2017.
- [20] 朱峰. 基于字符区域分割方法的车牌识别技术研究[D]. 长春: 吉林大学, 2011.
- [21] P. Naphon, S. Wiriyasart, T. Arisariyawong, L. Nakharintr. ANN, numerical and experimental analysis on the jet impingement nanofluids flow and heat transfer characteristics in the micro-channel heat sink[J]. International Journal of Heat and Mass Transfer, 2019, 131.
- [22] 李石林. 基于机器学习的车牌识别系统研究[D]. 东华理工大学, 2016.
- [23] 张三友, 姜代红. 基于 OPENCV 的智能车牌识别系统[J]. 软件导刊, 2016, 15(05):87-89.
- [24] 葛笑飞. 基于 OpenCV 的车牌自动识别系统的研究与实现[D]. 东南大学, 2015.
- [25] 马永杰, 马芸婷, 陈佳辉. 结合 CNN 多层特征和 SVM 的车辆识别[J]. 激光与光电子学进展, 2019, 56(14):141001.

致谢

本论文是在我的指导老师潘家辉教授的悉心指导下完成的，首先要感谢潘家辉老师对我的亲切指导与关心，从论文的选题到毕业设计的实现，都离不开老师对我的点拨与帮助，老师的一丝不苟与刻苦的科研精神一直鼓励着我，是我前进方向的指引，也是我学习的榜样！我还要感谢焦新涛老师和吴正洋老师，焦新涛老师引领着我对数字图像处理的学习，吴正洋老师带我打开机器学习新世界的大门，他们的引领对我完成毕业设计有很大的帮助。

同时，我还要感谢我的同学、朋友和家人们在背后的支持与鼓励，在我遇到瓶颈时，给予我最大的鼓励，是你们的关心和帮助让我在完成毕业设计和论文的道路上坚定不移地走下去。感谢 EasyPR 项目组对车牌识别项目的开源，帮助我对车牌识别系统系统地学习，本文的很多思路和代码都借鉴于 EasyPR，是你们的无私奉献让更多人受益匪浅。

大学本科四年即将接近尾声，我即将告别校园走入社会，很感谢在华师度过的大学四年岁月，不仅教会我很多专业知识与技能，还教会我如何做人做事，让我变得更加优秀，成为我走向社会的垫脚石。