

分类号_____

学校代码_____

学号 M201276100

密级_____

硕士学位论文

夜间高速公路路口车辆车牌识别系统的 设计与实现

学位申请人：

学 科 专 业： 软件工程

指 导 教 师： 副教授

答 辩 日 期： 2015.1.13

**A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree for the Master of Engineering**

**Design and Implementation of a License Plate
Recognition System for the Car on Freeway in Night**

Candidate :

Major : Software Engineering

Supervisor: Assoc. Prof.

January, 2015

独创性声明

本人声明所呈交的学位论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除文中已经标明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：

日期： 年 月 日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，即：学校有权保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权华中科技大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本论文属于 ☐ 保密， 在_____年解密后适用本授权书。
☐ 不保密。

（请在以上方框内打“√”）

学位论文作者签名：

日期： 年 月 日

指导教师签名：

日期： 年 月 日

摘 要

随着人们对车辆的使用越来越普及，交通网络的管理在深度和广度上的要求也越来越高，在这样的背景下，智能交通系统应运而生，而车牌信息一直是智能交通系统所需车辆信息的重要部分，与之对应的车牌识别系统自然也是智能交通系统的重要组成部分。由此出发，构建一个夜间高速公路上使用的车牌识别系统，使之能良好的识别车辆的车牌，可以有效地提高该环境下智能交通系统的管理质量与效率。

车牌识别系统针对夜间高速路口车辆这一对象，以算法为主导，包括图像增强，车牌粗定位，车牌字符分割，车牌字符识别一共四个模块。图像增强模块使用线性灰度变换，通过合理选取参数，解决了夜间条件下光照不足导致车牌与背景对比度不足的问题，车牌定位方面，提出一种多点拟合的方法来矫正车牌，通过较为简洁的思路和处理方式，令车牌定位模块在时间和空间上的消耗处于一个较低的水平，提高了车牌矫正的运算效率。字符分割方面，提出一种面向文本的车牌精定位方法和一种基于二分投影的字符分割方法，通过对车牌文本的二次提取和二次分割，有效改善了光照不足条件下字符分割中常见的断裂，粘连等不利现象。字符识别方面，提出一种特征提取与模板匹配相结合的方式，通过提取出的特征在模板匹配所得的解集中进行优选，有效完成了字符识别的任务。

车牌识别系统通过 MFC 框架与 OpenCV 类库相结合的方式来实现，有效完成了对夜间高速路口车辆车牌的识别，经测试可知最终得到了较好的效果。

关键词：图像增强 车牌精定位 字符分割 字符识别

Abstract

The requirement for the management on traffic network in width and depth is higher than ever as the car is used more usually than ever. In this context, intelligent transportation system (ITS) has emerged as the times require. License plate information has always been an important part of information that the intelligent traffic system needed, an important part of naturally in intelligent transportation system is the license plate recognition system (LPRS). The construction of the license plate recognition system using a night on the highway from this perspective, which can recognition the stopping vehicle license plate well, it will improve the management efficiency and the quality of the environment of the intelligent transportation system effectively.

License plate recognition system which vehicles at car on high-speed road in night has four parts that named image enhancement, license plate location roughly, license plate character segmentation, license plate character recognition. The image enhancement part uses linear gray level transformation, solve the lack of night under the condition of insufficient light resulted in contrast with the background of the problem of license plate,. In license plate location part, firstly, uses the Log operator to detect edge, and extraction of the license plate region using mathematical morphology transform on the highway, solves the question of the complex background of license plate is not easy to separate from the object, finally proposed method multi point fitting correction of license plate, license plate correction improves the calculation efficiency. Character segmentation, put forward a text oriented license plate location method and a method for segmentation of two projection based character, effectively improve the fracture common light under the condition of lack of character segmentation, adhesion and other unfavorable phenomena.

License plate recognition system by combining the MFC framework class library and OpenCV way to achieve a complete, effective identification of the night high speed intersection stagnation for vehicle license plate, in each link compared with some similar algorithm results, achieved better effect.

Key words: Image enhancement License plate location
Character segmentation Character recognition

目 录

摘 要.....	I
Abstract.....	II
1 绪论	
1.1 研究背景与意义.....	(1)
1.2 国内外研究概况.....	(2)
1.3 论文的主要研究内容.....	(3)
2 关键技术研究	
2.1 MFC 类库.....	(4)
2.2 OpenCV 类库.....	(5)
2.3 本章小结.....	(5)
3 系统分析与概要设计	
3.1 应用背景分析.....	(6)
3.2 系统需求分析.....	(6)
3.3 系统模块重难点分析.....	(7)
3.4 系统总体设计.....	(8)
3.5 本章小结.....	(10)
4 车牌识别系统的实现	
4.1 预处理模块的实现.....	(11)
4.2 车牌定位模块设计.....	(16)
4.3 字符分割模块设计.....	(28)
4.4 字符识别模块设计.....	(32)
4.5 本章小结.....	(36)

5 车牌识别系统的测试

5.1 测试用例.....	(38)
5.2 测试环境与配置.....	(39)
5.3 车牌图像预处理模块测试结果与总结.....	(39)
5.4 车牌定位模块测试结果与总结.....	(40)
5.5 车牌字符分割模块测试结果与总结.....	(40)
5.6 车牌字符识别模块测试结果与总结.....	(41)
5.7 系统测试结果与总结.....	(41)
5.8 本章小结.....	(43)

6 总结与展望

6.1 全文总结.....	(44)
6.2 展望.....	(44)

致 谢.....	(46)
----------	------

参考文献.....	(47)
-----------	------

1 绪论

1.1 研究背景与意义

随着人民生活水平的不断提高，汽车已经越来越成为生活中不可分割的一部分，车辆在带来方便的同时也带来更多的问题——交通事故，交通环境恶化，环境污染等等等等，过往简单化单一化的人工管理方式已经越来越不能适应新的需求^[1]。

同时，随着计算机技术和通信技术的不断提高，人工智能领域的不断发展，尤其是近几年在机器学习领域下神经网络这一特定方向上的不断创新，细化,现代智能交通系统已经开始步入正轨，拥有越来越多的产品^[2]。人们开始越来越多的将图像的识别，事件的记录和处理交给计算机去做，以便于其应用到实际生活之中。

智能交通系统在现代交通管理中使用非常频繁^[3]，由于它的智能化特性，为交通网络的管理带来了许许多多的方便，下面列举几条：

（1）交通现场监控，使用智能交通系统，可以非常直观的监控道路交通状况，获取车辆信息，便于控制路况。

（2）公路上事故自动测报，使用智能交通系统，可以很方便的发现堵车，拥塞，超速，排队等等路况异常，速度快，准确率高。

（3）对于不合格的车辆，能及时做出检查，对于潜在的安全隐患能及时做出预警，对于公路上的通行车辆可以实时受教应纳费用^[4]。

而要达成上述功能，首先要做的一点便是识别车辆，车辆的样子多种多样，往往难以做出普适性的系统全部识别，所以，对车辆的识别往往立足于对车牌的识别。如今的车牌识别技术，除了能在单一背景静态下识别车牌之外，还要能在复杂背景（如大雨，大雪）的条件下，在高速度导致的车牌失真环境下，在车牌自身有问题（污点，倾斜）的情形下准确识别车牌。

目前，由于种种细节上的技术困难，车牌的多样性以及环境的复杂性，还没有任何一个车牌识别系统能普遍使用于世界上的各个角落，然而与此同时，人们在实际的使用过程中对智能交通系统的要求也越来越高，作为智能交通系统的重要技术，车牌识别技术的研究理所当然的越来越受到重视。

1.2 国内外研究概况

早在上世界 80 年代，对车牌识别技术的研究就开始了，在刚开始的时候，车牌识别技术的自动化水准并不高，研究的方式主要是就某一具体问题作出具体方案，采取的步骤普遍为先用摄像头等图像采集仪器收集图像，用计算机做简单的处理，自此之后的步骤包括最终结果依然需要大量人工干预。

20 世纪 90 年代后，机器视觉方面的算法水准越来越高，越来越贴近于实用，而与此同时，计算机软硬件理论也越来越丰富，实际的计算机从组成到产品都开始呈现出日新月异的发展，国外早早的开始了对车牌识别系统的专项研究。这一阶段的研究成果十分丰富，例如 1990 年 A.S.Johnson 等实现了一个车牌的自动化识别程序^[6]，他们由图像分割开始，之后通过特征提取，模板构造和字符识别这三个步骤得到分割出的字符图像，再由预先设置的标准化模板来进行模式匹配，以当时的条件而言，他们研制的车牌识别系统能拥有 91% 的识别率足以令其成为那一时代同类产品的楷模。

继 A.S.Johnson 之后，在企业产品方面，典型代表有英国 IPI 公司基于数字信号电平与反向通道总噪声功率比研发的的车牌识别系统，新加坡 Obtasia 公司的 VLPRS 系统，HSTOL 公司的 LPR 系统，以色列 Hi-Teck 公司的 See/Car System 系统^[7]，香港 AsiaVisionTechnology 公司的 VECON 系统，其中试图在普适性上有所突破的是以色列的 Hi-Teck 公司的 See/Car System^[8]，Hi-Teck 公司基于该系统推出了多个系列不同类型的产品^[3]，这些产品都有其各自的字符模板库，适用于不同国家的语言，他们希望通过这种方式来使该系统能用于不同国家的车牌识别。

此外，日本，英国，美国，加拿大都有适应自己国家车牌的识别系统，如实时车牌识别系统等等，系统应用的场合比较广泛，用途多样，适应性较强。各国的系统的共同之处是这些系统的图像采集都是基于车辆探测器的处理系统，故而当需要对实际情况作出系统改进的时候，往往也是在前端的车辆探测器作出改进，如针对夜间低俗车辆运动，选取红外实时拍摄设备进行图像采集。

上世纪 90 年代，各国基于各种技术各种不同情形下车牌的识别系统争相登场，而国内对于数字图像的研究也有了长足发展，加上国内交通网络方面的管理越来越

高，于是我国也开始了对车牌识别技术的研究。

就产品而言,国内最早,软件能力成熟度也较高的产品是中科院推出的“汉王眼”^[9],该产品在软硬件上均有高度集成,一经购买,不需要任何其他辅助组件便可以付诸使用^[10],是国内相对成熟的产品,现在主要应用于高速公路收费站,停车场等等。

1.3 论文的主要研究内容

本次设计针对高速公路路口处车辆车牌这一特定对象的识别问题,采用MFC+OpenCV 开发了一个能对车牌图像进行迅速反应并识别的程序,全文的主要工作如下:

(1) 针对夜间能见度低,采光差的特点,对比两种图像增强手法,优化参数,完成车牌图像增强部分的设计。

(2) 针对采光差导致的车牌灰度较高的问题,设计一种能在较暗的状态下对车牌进行粗定位的算法,进行初步定位,得到车牌区域。

(3) 针对采光差导致的车牌文本区域模糊,设计一种能克服字符粘连断裂的车牌字符分割算法,进行车牌文本的精定位,得到车牌文本区域。

(4) 针对汉字识别率较低的情况,设计一种能提高汉字识别率的车牌字符匹配算法,进行车牌文本识别,得到最终车牌识别结果。

本次设计立足于夜间高速公路路口这一实际情况,在这五点的基础上,逐步完成各个预定目标的算法,设计相应模块,完成对整个车牌识别系统的设计和实现。

2 关键技术研究

本次系统设计是针对夜间公路路口处车辆的图像处理系统，主要用到 MFC 类库和 OpenCV 类库，本章将概括性的阐述这两个类库。

2.1 MFC 类库

MFC（Microsoft Function Classes）是微软推出的封装了 Windows 底层接口的一套类库，它基于所见即所得的可视化编程理念^[11]。

对于顶层的界面编程，MFC 提供高度封装的一系列工具，这一点集中的体现是所创建的工程在创建之后已经是一个可以运行的程式^[10]，这个程式的界面控件可以由编程人员通过工具箱这一 MFC 工程的特有属性直接进行增加，删除和更改，并直接反应到最终的程式上，这即是所见即所得的编程理念。

对于底层的计算管理，例如 Windows 消息驱动机制，Windows 内存管理，Windows 进程控制，MFC 均提供更加简明快捷的编程接口和方法。在 Windows 消息驱动机制方面，MFC 以消息路由机制来封装原有的 Windows 消息传递机制，编程人员可以通过类向导这一 MFC 工程的特有属性，直接向程式里添加消息，MFC 会自动生成消息映射，消息响应函数声明和消息响应函数，编程人员只需要将针对消息的具体执行函数代码段填入消息响应函数即可^[12]。

MFC 的消息路由机制如图 2-1 所示。

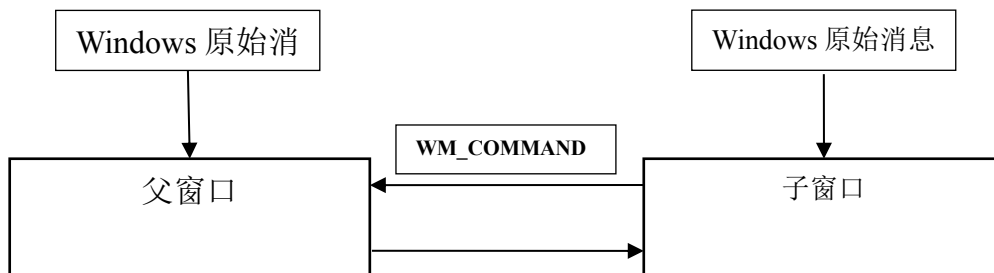


图 2-1 MFC 消息路由机制

2.2 OpenCV 类库

OpenCV (Open Source Computer Vision Library) 是由 Intel 公司提供的, 拥有跨平台特性的, 封装了一系列计算机视觉处理函数接口的开源类库^[13]。

由于开源类库强大的生命力, OpenCV 类库一直保持着高速的更新速度, 本次所用 OpenCV 类库的版本号是 v3.0,3.0 版 OpenCV 能满足当前主流操作系统, 如 Windows, Linux, MacOS 等等, 并支持多种编程语言, 除了最开始的 C/C++ 之外, OpenCV 还提供 Java, Python 等等不同语言的接口以适应不同的编程需求^[14]。

OpenCV 专注于计算机视觉领域, 由于该领域的基础即是数字图像处理, 故 OpenCV 提供一整套数字图像处理接口。OpenCV 提供超过 500 个数字图像处理的函数接口^[11], 这些接口包含数字图像处理中方方面面, 运算速度快, 实时性高, 在实际的使用过程中, 由于 OpenCV 类库强大的开源特性带来的代码高透明度, 可以根据实际情形, 很方便的做出优化和改进^[15]。

2.3 本章小结

本章主要介绍了系统所使用的两个类库, MFC 类库和 OpenCV 类库, 分析了这两个类库的特点, 阐述了这两个类库在应用中的优势, 在随后的系统设计中, 这两个类库都会用到。

3 系统分析与概要设计

3.1 应用背景分析

在夜间，由于光照不足，驾驶员视野恶劣，或者驾驶员临时有事，例如用餐，如厕等情况，时常出现滞留在公路路口的车辆，这些车辆不仅会消弱了该路口的车流量承受能力，还容导致公路路口转弯处的交通事故，这种情形下，出现车祸的几率远远高于白天相同路段出现车祸的几率，为交通网络的正常工作带来隐患，是必须要处理的主要问题之一^[16]。

有鉴于此，夜间公路路口处的车辆必须被监控，尤其是夜间停滞在公路路口的车辆，为了在发生此处情形前相关单位能获取第一手资料采取预防，在发生此种情形后相关单位能在第一时间做出反应，特为智能交通平台添加一个专门针对夜间公路路口车辆的车牌识别系统。

3.2 系统需求分析

一个完整的车牌识别系统，从最开始的车牌图像收集，到最后车牌识别结果的反馈，普遍意义上应能完成的任务有：

- (1) 后台通过前台的图像收集仪器收集视频。
- (2) 后台收集到视频后能将视频按帧分离出二维图像，传递给与图像情形相应的组件和模块，完成相应的数字图像处理和识别。
- (3) 后台处理完收集到的视频后，能将识别结果发送给相关单位，及时预警或通报，使他们能通过结果做出相应反应。

而在本次系统设计中，对前台的图像收集仪器和终端结果图像接受仪器的管理，主要是智能交通平台的底层来进行。同时，整个智能交通平台能为其所有下属子系统按照不同情况提供三维重建，图像还原，发送结果等一系列功能。

同时，在夜间，光照条件较差，路灯位置复杂，车辆位置随机性较大，故而系统在性能上还需要如下要求：

(1) 能实时的分离出照度场，能有效进行复杂照度场下的车牌识别。夜间的场景往往是在多个路灯多种光照的条件下形成的，如果不能有效分离出多个光照场，夜间的图片的背景往往会模糊。

(2) 区分照度场后，在做车牌定位时，对于公路背景要能有效做出屏蔽，分离车牌区域和背景区域。公路背景十分复杂，如果在车牌定位后，图像中还有公路背景，往往会对后面的字符分割操作带来很大困难，例如。蓝底白字汽车车牌，如果在车牌定位后，公路背景没有去掉，则车牌在做字符分割的时候，公路背景会加大字符特征提取的难度。

(3) 光照不足条件下，分离出的车牌总体灰度偏高，如果处理不当，字符往往会出现粘连现象，在这种情况下，要能有效分离出车牌字符区域和车牌背景区域，并避免字符粘连，为后续字符识别工作打下良好基础。

3.3 系统模块重难点分析

本次设计的模块与所需实现的功能相对应，模块一共划分为四块：

(1) 车牌图像预处理模块。通过数字图像处理，使得车牌与图片无关背景的对比更加明显，为后面的继续处理打下基础。

(2) 车牌图像定位模块。在预处理之后，定位出所需识别的车牌，一般分为粗略定位，车牌校正定位两部步。本模块在整个车牌识别系统中承担一个承上启下的作用，在此之前，我们所处理的图全部是车辆放置在各个不同的环境下的全景图，而在此之后，我们所处理的图全部是全景图中车牌区域的图像。

(3) 车牌图像字符分割模块。定位出车牌之后，为了方便字符的分割，需要在做一次图像预处理，也可以理解为车牌的精确定位，然后将车牌上的字符分离出来，并对字符作归一化处理，方便后续识别。

(4) 车牌图像字符识别模块。字符分离出来之后，字符的识别，尤其是汉字字符的识别，直接影响到整个系统的结果。现行系统有 2 种方法进行识别，一种是神经网络，一种是模板匹配，这一点将在第六章详细阐述。

就整个系统计算过程而言，各个模块之间的流程如图 3-1 所示。

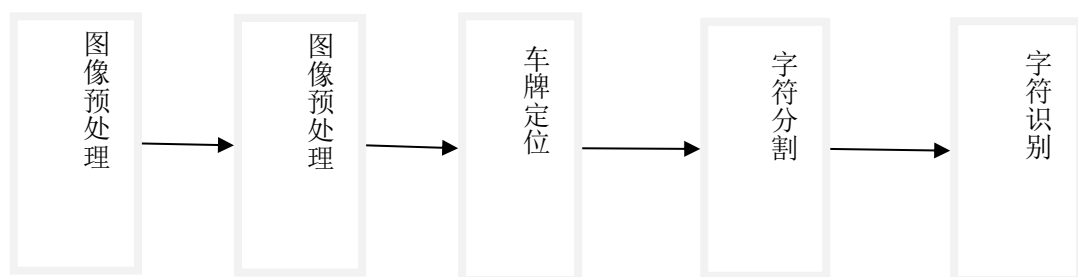


图 3-1 车牌识别系统流程及模块关系

为了符合实际应用——能准确快速识别出白天公路路口停滞车辆，系统在各个模块之中各面临不同的重点，这必然导致各模块有各自不同的要求，下面一一叙述：

（1）车牌图像预处理模块。这是整个车牌识别系统的启动环节，也是为后面一系列处理打下坚实基础的重要环节。这一模块的重点是，在完成车牌图像预处理的运算之后，车牌图像与无关背景之间的对比需要非常明显，方便后面的一系列数字图像处理操作。

（2）车牌图像定位模块。这一模块分为两步：车牌定位和车牌校正，车牌定位的重点是必须具有相当的准确性和可靠性，结果图中尽量只有车牌。车牌校正的重点是，车牌校正之后，所得结果中，字体必须整齐，切割后方向上能与模板相匹配，其难点是如何提高车牌校正响应时间和效率。

（3）车牌字符分割模块。这一模块的重点是，字符分割所得的结果必须清晰准确，不能有字符断裂，字符粘连，字符模糊的现象，在图像质量上能与模板相匹配，其难点也在于此。

（4）车牌字符识别模块。这一模块的重点是，字符识别的准确程度必须具有相当的水准，理想状态下能准确无误的识别每一个字符。

3.4 系统总体设计

车牌识别系统使用摄像头来采集视频序列，由视频采集卡通过网络传送至计算机或者 DSP 处理器，对视频进行快速处理，处理结果通过网络传输到服务器进行备份，计算机终端可以通过网络随时查看数据。

3.4.1 系统硬件平台

数字图像处理的计算量向来比较大，由于系统自身只是平台上的一个组件，并不考虑并行化计算，于是对于 CPU 的主频和存储空间也有进一步的要求，CPU 主频过低，计算缓慢，运行时间无法接受，存储空间太小，频繁换页会导致抖动，致使程序卡顿甚至崩溃。调试结果表明，为了适应前文中的算法，CPU 主频最低不能低于 800MHZ，而内存最低不能低于 512M。本次系统实现中 CPU 主频为 2.5GHZ，内存为 6G。系统为 Windows7，程序编写的平台为链接 OpenCV 类库的 VisualStudio2010。

3.4.2 系统软件平台

本系统基于 MFC 对话框程序，使用 OpenCV 类库，依据算法步骤，逐步展开，车牌图像预处理模块主要采用灰度线性变换，车牌定位模块分为 4 步，边缘检测使用 Log 算子，边缘检测完成之后使用数学形态学对车牌的区域进行筛选，定位出车牌之后，依据车牌字符分隔的效果来判断是否应该进行车牌矫正，矫正采用一种基于多点拟合的数值计算方法。车牌的字符分割是先对车牌文本进行定位，得到文本所处区域，在该区域内分为两步，第一步是由垂直投影法得到一个解集，第二步是在解集内进行再次识别。车牌的字符识别则是模板匹配和特征提取相结合来进行。

各模块内部流程已在各章实现相关小节中说明，总体模块划分如图 3-2 所示。

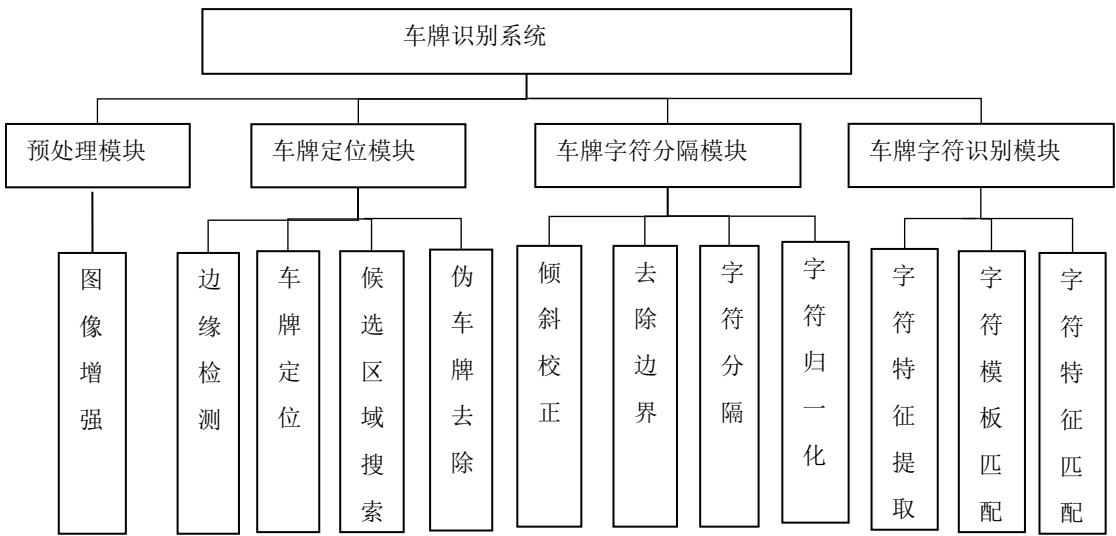


图 3-2 车牌识别系统框架图

3.5 本章小结

本章先叙述了这次系统设计的适用背景，其后就此背景展开，依次提出了任务描述，功能需求，概要设计和重难点分析。在概要设计中，就每一个模块的重要性展开描述，然后做出系统概要流程。在重难点设计上对每一个模块展开，详细叙述了每一个模块的重点及难点。

4 车牌识别系统的实现

本次设计车牌识别系统是一个串行计算系统，本章将按照计算流程，依次介绍图像预处理模块，车牌图像粗定位模块，车牌图像字符分割模块和车牌图像字符识别模块。

4.1 预处理模块的实现

公路路口区域，尤其在夜间，由于光照不均，背景复杂，往往需要在车牌处理之前先做出相应的预处理^[13]。这是因为该条件下获得的图往往存在很多不利于处理的因素，所以所获得的彩色数字图像往往需要一系列的预处理操作。

预处理模块是整个系统的起步操作，它在整个系统中承担为后续所有操作提高图片质量的任务，主要包括图像的灰度化和图像增强两个步骤，下面依据步骤逐一实现该模块。

4.1.1 彩色图像到灰度图像的转换

由于彩色图像信息复杂，直接进行图像增强，会导致系统变慢，存储空间也会急剧增加，通常的做法是将彩色图像转换为灰度图像来进行处理。

要得到灰度图，一般意义上所需要采取的操作为^[17]：

$$GrayScale(i,j) = Resource(i,j).red * R + Resource(i,j).blue * B + Resource(i,j).green * G \quad (4-1)$$

GrayScale 代表转换后的灰度图。

Resource 代表原图。

red 代表每个像素值中的红色分量，

blue 代表每个像素值中的蓝色分量，

green 代表每个像素值中的绿色分量。

R, B, G 分别代表由原始图像向灰度图像转化过程中的加权系数，R, G, B 必须全部大于 0，且 $R+G+B=1$ 。

本文采用经验权值， $R=0.299, G=0.584, B=0.117$ 。结果如图 4-1 所示。



图 4-1 原始车牌图像图（左）原始车牌灰度图像（右）

4.1.2 针对车牌的图像增强模块设计

图像增强操作依据思路的不同主要分为频域法和时域法。前者是一种基于统计量的宏观算法思路，把图像看成一种二维信号，对其进行信号增强，主要理论涉及离散正交变换，离散傅里叶变换，沃尔什变换，小波变换等等，其主要手段有低通滤波和高通滤波。后者是一种基于像素点阵的微观处理方式，主要算法有线性灰度变换，同态滤波，中值滤波法等等，它们可用于去除或减弱噪声。

由于本次系统设计所使用的线性灰度变换在思路隶属于时域增强，所以这里重点介绍时域增强的手段，时域增强是一种面向像素点，反应像素数值与像素坐标关系的图像表达方法，它可以按照公式 4-2 进行描述^[18]：

$$Result(i, j) = Function(Resource(i, j)) \quad (4-2)$$

i, j 代表像素在图像中的坐标。 $Result$ 和 $Resource$ 分别代表结果图和原图。 $Function$ 为增强操作所采取的函数，如果是针对某一个像素点， $Function$ 为一个点操作函数，如果是针对某一块区域， $Function$ 代表模板操作函数。

图像增强的算法较多，其中最常见的是高斯同态滤波和线性灰度变换，这两种算法各有其优势，但适用的场合各不相同。

下面将就这两种算法分别介绍，算法中的参数选取可以依据实际情形适应，对于不同的情形，需要多次试验来选择最佳的参数。这里以夜间公路路口处图片实验效果最佳为原则来进行参数选择。

（1）高斯同态滤波算法介绍

同态滤波基于这样一个事实：图像是由物体对光线进行反射来决定的，它的本

质是物体在光源下的投影^[19]。光线照射所形成的场称为照度场，物体反射的投影称为反射场。

$$F(x, y) = R(x, y) * r(x, y) \quad (4-3)$$

公式3-3表示了图像与照度场和反射场之间的函数关系， $F(x, y)$ 表示图像中的像素点集合， $R(x, y)$ 表示光线照射到物体上的点集合， $r(x, y)$ 表示物体对光源的反射场。

同态滤波一般采取如下步骤：

- (1) 对于公式4-1两侧取对数，将卷积表达转换为两个相加的分量：

$$\ln(F(x, y)) = \ln(R(x, y)) + \ln(r(x, y)) \quad (4-4)$$

- (2) 对上式两侧同取傅里叶变换，由于 $y=\ln(x)$ 是一个严格单调递增函数，故可以保证傅里叶变换之后的结果依旧分别处于各自的频域之内。

$$F(u, v) = \text{Fourier}[F(x, y)] = \text{Fourier}[R(x, y)] + \text{Fourier}[r(x, y)] = R(u, v) + r(u, v) \quad (4-5)$$

- (3) 由于在夜间，灯光强烈，为了能突出车牌，我们首先要做的就是减小照度场所对应分量 $R(x, y)$ ，加大反射场所对应分量 $r(x, y)$ ，所以，所选滤波器的数学仿真图形应该如图4-2所示。

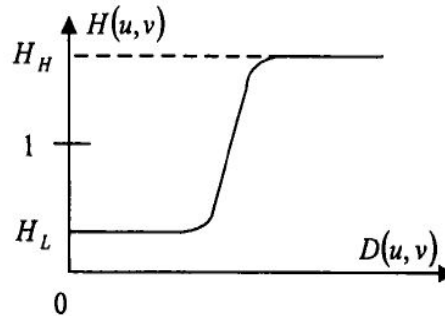


图 4-2 同态滤波函数平面投影

图4-2中 $D(u, v) = u^2 + v^2$

$$G(u, v) = H(u, v) F(u, v) = H(u, v) R(u, v) + H(u, v) r(u, v) \quad (4-6)$$

于是 $G(u, v) = \text{Fourier}\{\ln[g(x, y)]\}$

- (4) 由于前面对图像做了傅里叶变换和对数变换，这里需要取它们的逆运算来进行还原，如公式4-7所示。

$$g(x, y) = \exp\{\text{Fourier}^{-1}G(u, v)\} \quad (4-7)$$

为了增强反射场，应该取 $H_h > 1$ ，同时为了压缩照度场，应该取 $H_l < 1$ ，本文实际取值为： $H_h = 3.0$ ， $H_l = 0.1$ ，实验结果如图4-3所示。



图 4-3 原始灰度图（左）和基于高斯函数的同态滤波结果（右）对比

(2)灰度线性变换算法介绍

基于点运算的图像增强又名灰度变换，该方法可以由频率域上各个灰度的不同运算而达到不同的增强结果，其目的主要是为了提高图像对比度，增强图像反差，突出我们所关心的对象^[20]。

线性灰度变换，往往用于已知一个图像的灰度区间分布，需要将它的分布调整到另一个区间下。线性灰度变换的公式描述是：

$$i = \frac{B - A}{b - a} (i - a) + A = \frac{B - A}{b - a} i + \frac{A b - B a}{b - a} \quad (4-8)$$

车牌一般情况下位于车辆的中间偏下部分，而车辆往往上半部分光照较强，灰度较低，车牌处于中间，灰度适中，车牌以下就是车辆阴影，灰度较高，故而可以采用拉伸中间灰度，压缩两端灰度的方法来突出车牌信息，该方法可以如下函数描述：

$$G(i, j) = \begin{cases} \frac{a'}{a} G(i, j) & 0 < G(i, j) < a \\ \frac{b' - a'}{b - a} [G(i, j) - a] + a' & a \leq G(i, j) < b \\ \frac{c' - b'}{c - b} [G(i, j) - b] + b' & b \leq G(i, j) < c \end{cases} \quad (4-9)$$

上述表达式中， $|a'| > a$, $|b'| > b$, $|b' - a'| > b - a$, $|c' - b'| > c - b$, a 取不超过整个图像

10%像素点数的最大灰度级， b 取不超过整个图像像素点数 90% 的最大灰度级。 a 取 $a/3$, b 取 $5b/3$, $c=c=256$ 。

结果如图 4-4 所示。



图 4-4 原始灰度图（左）与线性灰度变换的结果（右）的对比

由图 4-4 中两幅图可以清晰的发现，线性灰度变换结果显著好于同态滤波结果。

其原因主要是因为，同态滤波结果适应于分离多照度场背景，而在此次取材过程中，多为夜间取材，阳光极弱，导致背景自身的反射场极弱，而灯光较强，近似可以认为是单照度场。反观线性灰度变换，由于直接作用于时间域进行图像处理，突出了车牌处于中间灰度这一特点，该方法无论背景照度场如何，均能直接弱化背景，这等同于屏蔽了照度场，故而令车牌更加明显的显现出来，与背景之间的差异也变大。

由此可以看出，对于夜间公路路口这一特定条件，线性灰度变换显著好于高斯同态滤波，故选用线性灰度变换来进行车牌原始图像增强。

4.1.3 预处理模块流程图

由于所选取线性灰度变换这种算法是以单个像素的灰度值来进行分类计算，故而需要先遍历灰度图，逐一读取像素点的灰度值，再一一判断处于哪一个区间，分段对单一像素点进行处理。

在实际的设计中，这里的处理在公式，方法，参数上各有不同，均以实际实验结果最佳为依据，而在本次系统设计中，这里的处理用的是一次线性关系，如公式 4-7 所示。

在完成上述处理之后，输出结果图。

从输入原图到输出结果图，图像增强模块的算法流程比较简单，如图 4-5 所示。

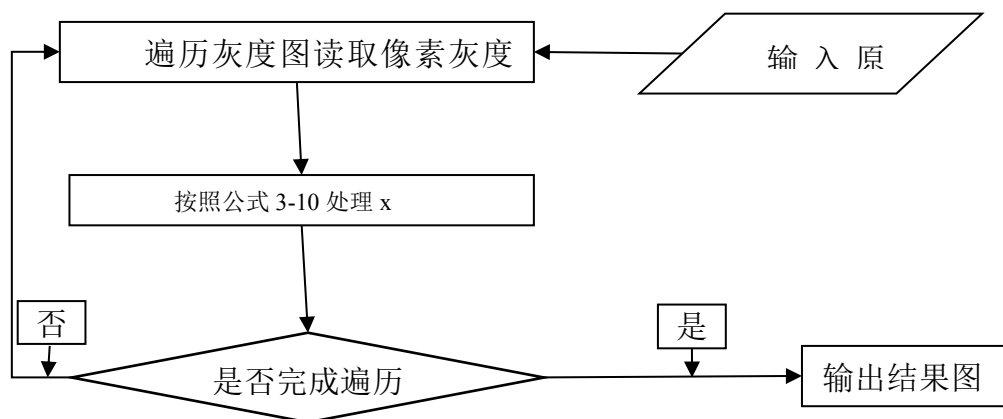


图 4-5 图像预处理模块流程图

车牌定位的一般步骤是先做边缘检测，使对象和背景分离开，然后通过一定算法隔离出车牌区域，最后矫正车牌，为字符分割做准备^[22]。

车牌定位模块在整个系统中承担一个承上启下的作用，在这一模块运算前和运算中，整个系统的处理对象是采集得来的原始图像，而在车牌定位模块计算完成之后，整个系统的处理对象就变为车牌及误差导致的车牌周边一小块区域所组成的图像。

本次设计的难点在于，夜间车牌光照条件差，背景较为复杂，在这种条件下，由于边缘自身模糊，边缘检测容易出现断裂，粘连等不利现象，同时，由于检测出的边缘不能保证处处连续，导致车牌定位不易准确，容易受到背景干扰，所以，要有一种行之有效的方案在这两者之间做出优化，使之能准确的定位出车牌。

4.2 车牌定位模块设计

4.2.1 车牌定位模块流程设计

此次设计中，车牌粗定位模块如图 4-6 所示，接受图像增强结果后，使用 Log 算子进行边缘检测，然后使用数学形态学方法来提取车牌区域。判断车牌是否倾斜的方法是先假设图像是倾斜的，直接使用多点拟合，并定下一个阈值 K 。若多点拟合所得的直线斜率的绝对值大于 K ，则认为图像是倾斜的，进行校正再输出图像，否

则直接输出图像。

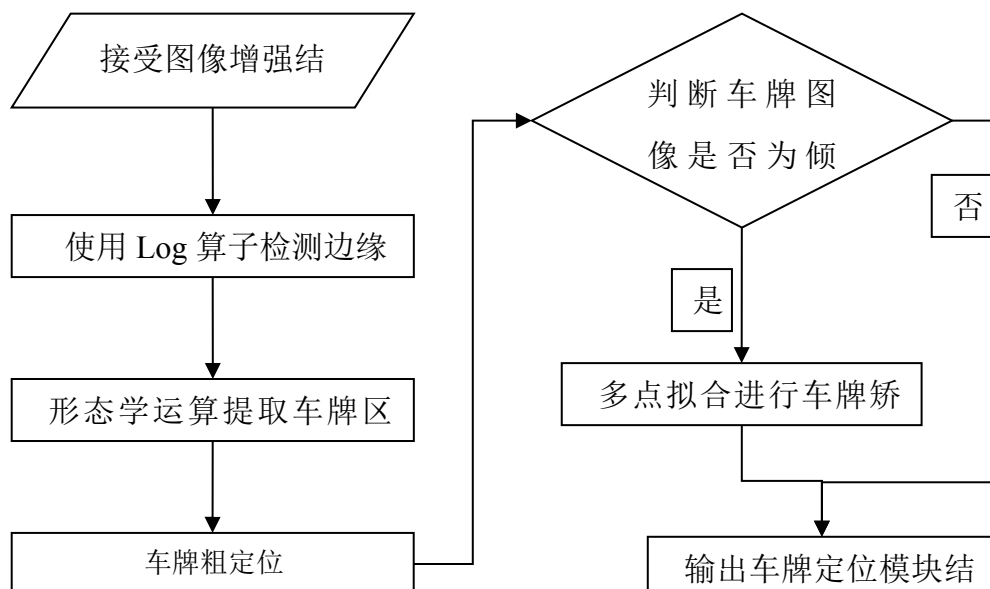


图 4-6 车牌定位模块流程图

下面依据图 4-6，依次阐述所需的子模块。

4.2.2 边缘检测子模块设计

为了标示出车牌区域，首先要对车牌的边缘进行检测，现在边缘检测方面的理论十分丰富，总体而言，都是利用差分算子模拟像素梯度变化来进行检测。

更具体一点来说，一幅图像的各个区域总有不同程度的色彩幅度变化，在色彩变化幅度不大的地方，梯度变化缓慢，像素之间灰度的差值不大，而在图像边缘处，色彩剧烈变化，梯度变化也随之激增，像素之间灰度的差值变化迅速增大，下面将分别对四种常用微分算子进行介绍，针对夜间公路路口处采光弱的特点进行优选^[23]。

(1) Roberts 算子

Roberts 算子是一种利用局部差分寻找边缘的算子，它基于一阶导数，在一阶导数下，认为若是依据 Roberts 理论所得的卷积结果大于一个阈值 Thresh，则该处即为边缘所处的区域。

对于一阶导数而言，梯度对应一个向量，可以用公式 4-10 描述^[25]：

$$\nabla f(x, y) = [G_x, G_y]^T = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right]^T \quad (4-10)$$

在实际的系统设计过程中，往往使用数字信号代替模拟信号，从而起到简化计算的效果。该向量在数字信号下的表示为：

$$g(i, j) = |\nabla f(i, j)| = \{[f(i, j) - f(i+1, j+1)]^2 + [f(i+1, j) - f(i, j+1)]^2\}^{\frac{1}{2}} \quad (4-11)$$

常见的 Roberts 算子的表达式为：

$$|\nabla f(i, j)| = (G_x^2 + G_y^2)^{\frac{1}{2}} \quad (4-12)$$

可以使用如图 5-1 所示的两个数字模板来表达 Roberts 算子，在理论上，这两个数字模板的实际意义是 Roberts 算子的两个卷积核。

1	0	0	1
0	-1	-1	0

图 4-7 Roberts 算子数字模板

Roberts 算子提取结果如图 4-8 所示。

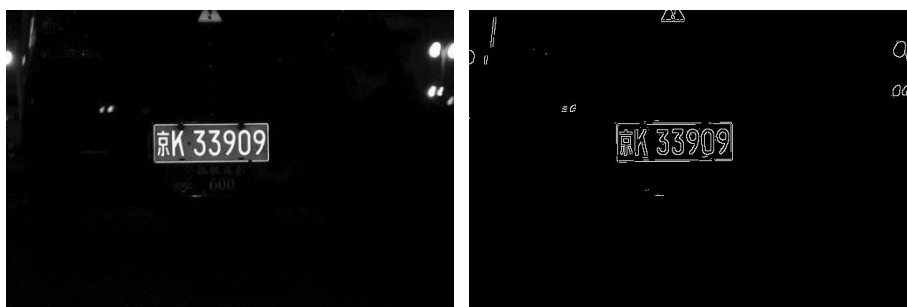


图 4-8 灰度图（左）与 Roberts 算子边缘检测（右）的对比

由 Roberts 算子的模板和最终的实验结果可以看出，它只利用了 4 个相邻的点来和原图像进行卷积，算子中有效地加权系数较少，故而在实验结果上可以看到，它对噪声比较敏感，所得的边缘时常伴有断裂，粘连等现象，效果不佳。

(2)Sobel 算子

Sobel 边缘检测共有两个卷积核，分别针对垂直和水平两种情况，这两个卷积核

分别与图像卷积。

Sobel 的算子模板如图 4-9 所示^[26]:

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

图 4-9 Sobel 算子的两个卷积核

Sobel 算子的检测实验结果如图 4-10 所示。



图 4-10 原始灰度图（左）与 Sobel 算子边缘检测（右）的对比

然而由实验结果可以看出，Sobel 算子往往会出现边缘粘连，夜间公路上采光不足，边缘模糊，不适应于使用 Sobel 来进行边缘检测。

(3) Prewitt 算子

Prewitt 算子也拥有两个卷积核，它是一种利用一阶导数得到的算子，其卷积核如图 4-11 所示^[27]:

-1	-1	-1
0	0	0
1	1	1

1	0	-1
1	0	-1
1	0	-1

图 4-11 Prewitt 算子的两个卷积核

Prewitt 算子实验结果如图 4-12 所示。

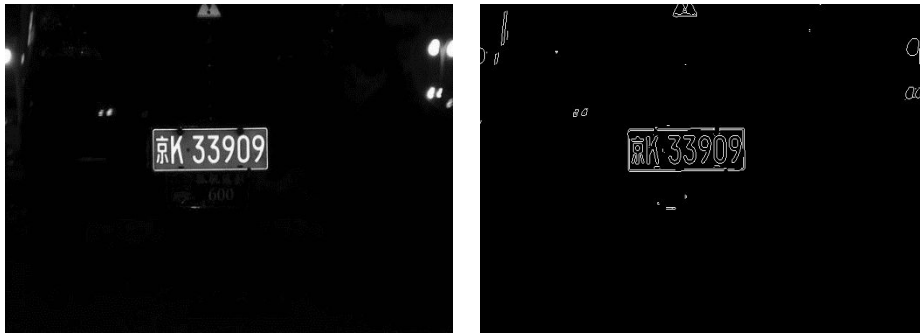


图 4-12 原始灰度图（左）与 Prewitt 算子边缘检测（右）的对比

Prewitt 算子的优点是速度快，利用平均可以有效抑制噪声，但是从 Prewitt 算子模板上看，该算子不具有各向同性，由实验结果可以看到，由于夜间采光不足导致的边缘模糊，使得 Prewitt 算子的结果也出现了种种问题，最大的一个问题是锯齿边缘，这是由于由于边缘和背景在一部分区域中的像素在数值方面区分不大，致使其检测出来的结果往往出现锯齿边缘，并伴有边缘不连续的状况。

(4)Log 算子

拉普拉斯算子是一种经由二阶导数得到的算子，它考虑 x 和 y 两个方向上的偏微分值，它的原始数学公式（4-13）所示^[28]:

$$\nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2} \quad (4-13)$$

拉普拉斯算子便是经由公式 4-4 推导所得到的一种模板，它实际只需要一种模板，但由于使用过程中依据实际情况各有不同，这里列举出它的两种常用数字模板，如图 4-13 所示。

0	-1	0
-1	4	-1
0	-1	0

-1	-1	-1
-1	8	-1
-1	-1	-1

图 4-13 拉普拉斯算子的两个常用模板

一般而言，由于拉普拉斯算子抗噪能力不强，对噪声比较敏感，在使用之前，

可以先用该算子与高斯函数卷积，高斯函数可以如下描述：

$$h(x, y) = \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (4-13)$$

如果 令 $r^2 = x^2 + y^2$ ，卷积可以如下描述：

$$\nabla^2 h = \left(\frac{r^2 - \sigma^2}{\sigma^4}\right) \exp\left(-\frac{r^2}{2\sigma^2}\right) \quad (4-14)$$

由公 式 5-6 可以得到高斯拉普拉斯算子的一种

种 5x5 整合模板，这种算子的英文简称为 Log，最终的 Log 算子模板如图 4-14 所示。

-2	-4	-4	-4	-2
-4	0	8	0	-4
-4	8	24	8	-4
-4	0	8	0	-4
-2	-4	-4	-4	-2

图 4-14 高斯—拉普拉斯算子模板

由公式 4-14 可知函数在 $\pm\sigma$ 处有零点，可知用该函数与图像卷积会模糊图像，该函数形成的算子在与图像做卷积时，实质上是先对图像做了高斯模糊，去掉由于夜间采光问题造成的噪声，并且，高斯模糊还能对公路背景的干扰做出一定程度上的补救，在整个图像排除了这两点不利因素之后，对于模糊之后的图像，再用拉普拉斯算子来进行边缘检测，效果如图 4-15 所示。

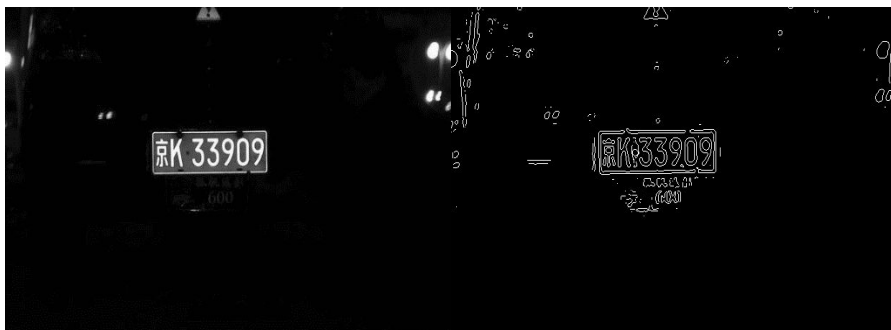


图 4-15 原始灰度图（左）与 Log 算子边缘检测（右）的对比

Log 算子采用二阶偏导，考虑到两个方向上像素变化的程度，相比于使用一阶导

数为理论依据的 Roberts 算子，它的边缘检测效果更加精准，相比 Sobel 与 Prewitt，Log 算子对方向并不敏感，能很好的适应各个方向的特征提取需求，故而本文采用 Log 算子。由图 4-15 可以看出，Log 算子的效果相对较好，在实际的程序中，该算子能有效抑制噪声，显著提高信噪比。

4.2.3 车牌粗定位模块设计

在夜间，公路路口上，光照条件差，边缘模糊，导致形态学变换结果往往不是精确的车牌结果，然而背景复杂情况下，车牌一次性精确定位的难度太大，所以不妨分两次来实现，第一次称之为粗定位，它用于去除无关的背景。车牌的粗定位这一模块上，目前主要的思路有使用特征提取直接进行处理的思路，也有使用神经网络让机器学习到粗定位的思路，下面分别介绍这两种思路。

车牌图像定位模块的现有思路分析如下：

（1）对车牌做特征提取直接进行处理的思路

我国车牌有其独特的特征，例如每张车牌的字符包含汉字，字母和阿拉伯数字在内一共有7个，这7个字符一定排布于车牌中间部分，间距相等，字符外围的大小相等^[29]。故而相应的也有各式各样特征提取的方法，例如统计二值图像上的跳动，或二值图像上的白色像素点分布情况等等，统计跳动主要是利用字符分布集中，在二值图像上，车牌字符区域较之车牌背景区域在各个行向量上均拥有更多的跳动次数。白色像素点数则是因为图像处理的过程中，为了把字符和背景分开，常常想方设法将这两者分别变换为黑色和白色两种颜色呈现在二值图像中。

该方法的局限性在于，若是背景中噪声较大，背景区域也出现了密集的跳动分布，则该方法在得到车牌区域时误差较大。

（2）利用神经网络让机器自行学习车牌位置的思路

利用神经网络来进行定位也是一种常用的思路，一般情况下，该思路是先编写类似于人脑中神经元的函数，称为激发函数，然后使用若干实例对激发函数进行训练，就车牌识别系统而言，这些实例便是车牌图像，训练完成后得到对于车牌图像较为敏感的激发函数，随后使用这些函数构成的神经网络进行车牌定位^[30]。

基于神经网络的思路由于方法复杂，要求较高，而神经网络的收敛速度较慢，维护不易，所以这里选择使用特征提取的思路。

在边缘检测结束之后，该模块接收到如图 5-10 的结果图，采用一种基于数学形态学的定位方法，这种方法主要使用了腐蚀操作，图像闭运算操作和图像开运算操作，腐蚀操作能有效去除车牌背景中的无关边缘，使边缘检测结果更加准确，闭运算操作能有效去除车牌中一部分小噪声区域，开运算操作能有效独立出车牌所在区域。

(1) 接受 Log 算子处理后的图像结果，得到图像中的边缘信息。如图 4-16 所示。

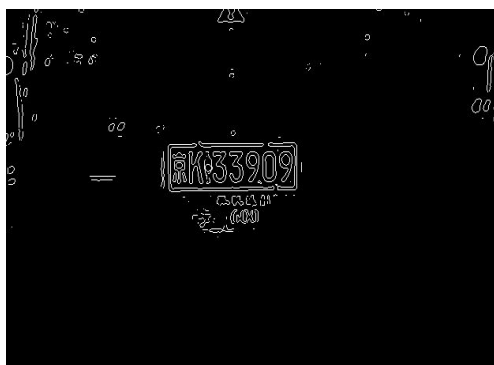


图 4-16 Log 算子检测结果

(2) 对提取后的图像进行腐蚀运算，去除噪声和一部分小对象，如图 4-17 所示。



图 4-17 图像腐蚀结果

(3) 腐蚀后的图像进行闭运算，消除车牌对象内部小空洞，如图 4-18 所示。

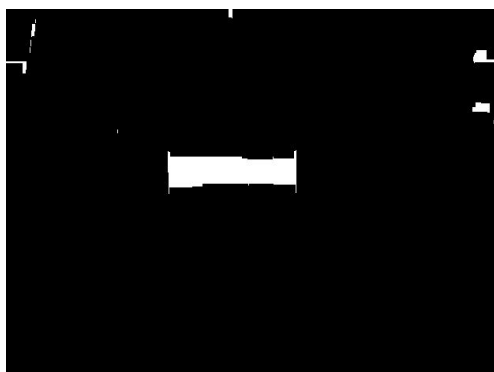


图 4-18 图像闭运算结果

(4) 对闭运算后的图像进行开运算，进一步去除小对象和无关细节，如图 4-19 所示。

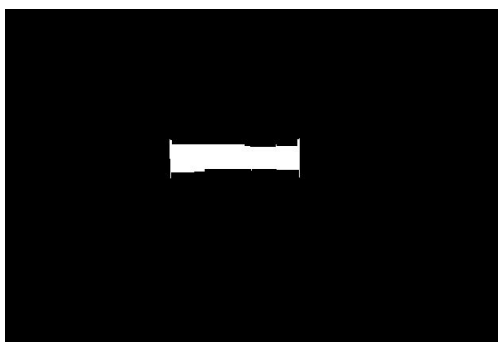


图 4-19 图像开运算结果

在最后的图 4-19 中已经可以明显的区分出车牌区域与其他区域的区别，于是可以开始统计车牌区域。

车牌的粗定位子模块设计如下：

本文定位车牌采用一横一竖两根扫描线分别遍历全图，依据图像特征，分别定位出上下左右四个方向的车牌边缘，从而定位车牌。定位车牌步骤如下：

(1) 在 Matlab 下定义“ $[m,n]=\text{size}(\text{Source})$ ；”，*Source*代表图 5-13 的形态学运算结果，则 *m* 为该图长度，*n* 为该图宽度。

(2) 建立一个横向向量 $A = \{a_{11}, a_{21}, a_{31}, \dots, a_{n1}\}$ ，由下至上遍历图 5-13，若图中某点 *Source*(*x*,*y*) 处值为 1,则该点为白色像素，在 *A* 中相应分量 a_{x1} 处加一，处理完成之后，*A* 的各个分量便已经记录了图 4-19 上每一行向量中白色像素点的个数。

(3)对于车牌区域的上界,可以认为若 A 中有某一向量 a_{x1} 的值小于一个阈值 a , 这里 $a=5$.并且该行的下一行所对应的 A 中向量值 $a_{(x-1)1}$ 大于 a , 则该行是车牌上界, 于是由下至上对 A 进行分析: 若该行白点像素多于 5, 并且不是第一行 (若为第一行, 代表遍历结束), 则继续向上, 定出车牌上界。

(4)对于车牌区域的下界,可以认为若 A 中有某一向量 a_{x1} 的值小于一个阈值 a , 并且该行的上一行所对应的 A 中向量值 $a_{(x+1)1}$ 大于 a , 则该行是车牌下界。

于是由下至上对 A 进行如下分析: 若该行白点像素多余 5, 并且不是最后一行 (若为最后一行, 代表遍历结束), 则继续向下, 定出车牌下界。

(5) 建立一个纵向向量 $B = \{b_{11}, b_{12}, b_{13}, \dots, b_{1m}\}$, 由左至右遍历图 5-13, 若图中某点 $Source(x,y)$ 处值为 1, 则该点为白色像素, 在 B 的该分量 b_{1y} 处加一, 如此处理之后, B 成为一个记录图 4-19 上每一列向量中白色像素点个数的向量。

(6)对于车牌区域的左界,可以认为若 B 中有某一向量 b_{1x} 的值小于一个阈值 β , 并且该行的左边一行所对应的 B 中向量值 $b_{1(x-1)}$ 小于 β , 则该行是车牌左界, 这里 $\beta=3$ 。

于是由右至左对 B 进行如下分析: 若该行白点像素小于 3, 并且不是第一列 (若为第一列, 代表遍历结束), 则继续向左, 定出车牌左界。

(7)对于车牌区域的右界,可以认为若 B 中有某一向量 b_{1x} 的值小于一个阈值 β , 并且该行的右边一行所对应的 B 中向量值 $b_{1(x+1)}$ 小于 β , 则该行是车牌右界。

于是由左至右对 B 进行如下分析: 若该行白点像素小于 3, 并且不是最后一列 (若为最后一列, 代表遍历结束), 则继续向右, 定出车牌右界。

在上下左右都定出之后, 已经可以知道车牌所处的区域, 这一步完成后, 便可以直接对原图进行截取, 得到车牌的具体图像。最后结果如图 4-20 所示。



图 4-20 车牌定位结果

4.2.4 车牌倾斜校正模块设计

车牌识别系统进行至此，下一步就要开始字符分割和字符匹配了，但字符模板均为标准水平模板，为了能与该模板相匹配，字符分割的结果也必须是水平模板，于是在字符分割之前，若是车牌粗定位结果如图 4-21 所示，车牌所处的位置并不正，切割出的字符无法参与字符匹配，就必须对车牌进行倾斜校正。



图 4-21 车牌倾斜示例

车牌的倾斜校正有其若干种常用方法，在实际的应用中，都是以各自的不同情形来进行选择，如果图像清晰，可选方法中典型的为重心法^[32]，如果图像并不十分清晰，可选方法中最为典型的一种为 Hough 变换，在这里，将对这两种方法分别展开介绍。

1) 重心法，重心法算法步骤如下：

(1) 图像二值化，图像的主要对象和无关背景在二值化后的图像上是区分得最为明显的，更为重要的是，二值化之后，图像的像素只分为‘1’左右的值和‘255’左右的值，简化了数值规律，方便后面处理。

(2) 对于每一列，认为对象的值为‘1’，背景的值‘255’，取出每一列上的多个‘1’序列，将其均视为质点，将‘1’序列的坐标中心视为各个质点坐标。

(3) 计算每一列的重心，首先求取每一列上质点的最大坐标 Max_X 与 Min_X ，将每一个‘1’序列的长度视为重量。

如果本列有且仅有一个质点，那么质点坐标 $X = \text{Min_X} + (\text{Max_X} - \text{Min_X}) / 2$ ；

如果本列有多个质点，则依据重量对每一个质点坐标进行加权考量，得到本列的质点坐标。

(4) 对于每一列的质点坐标，使用最小二乘法拟合出一条直线，该直线的倾斜角的即为所需旋转的角度。

重心法思想简单，易于实现，但重心法对于噪声非常敏感，而且车牌识别进行

至此，主要目的是为了识别字符，汉字字符的图像中，质点坐标往往不代表字符中心坐标，并且每一列的质点坐标跳跃很大，重心法基本无法完成车牌的校正。

2) Hough 变换^[20]，霍夫变换是一种以参数空间转换为主要手段的校正方式，假设一条直线的解析式为 $y=kx+b$ ，我们取出若干个点，想象这样一个空间，即这若干个点都分布在 $y=kx+b$ 这条直线的附近，那么，这些点就是一个以 x 和 y 为基准的空间下的解集，但若反过来，将 k 和 b 视为未知数， x 和 y 视为参数，那么会自然形成一个以 x 和 y 为基准的空间，该空间下，若干条直线共同相交于 (k, b) 这个点。

Hough 变换抗干扰能力很强，所得结果往往比较好，但是 Hough 变换需要一个清晰的车牌边框，然而，实际情况下，由于拍摄角度，光照，车牌自身受到污染等等因素，车牌边框往往不是完整的，图 5-16 即是一例，同时 Hough 变换需要大量空间，计算缓慢，实时性差。

这里提出一种多点拟合的校正方法

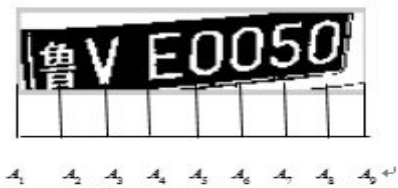


图 4-22 校正中的取点区域划分

首先将图 4-22 所示的图像横向平均分割为 8 个部分，如图 4-22 所示， A_i ($i=1, 2, \dots, 6, 8, 9$) 标示各区域横向端点。对于这 8 个部分分别由下往上遍历，并统计每一行中白色点的数目，直到出现第一个大于阈值 T （此处 T 取 2，排除噪声干扰）行的 a_i 。自左向右遍历，找到第一个像素值在‘1’附近的点之后重复该过程，在每一个区域中分别记录这个点 $\text{Source}(i, j)$ ($i=2, 3, 4, \dots, 8$)，依次找到。对于这 8 个点，使用最小二乘法拟合出它们所最接近的直线，得到该直线的斜率，这个斜率的反正切运算所对应的角度就是图像应该选取的值。



图 4-23 车牌校正前后的对比

由图 4-23 中可以看出本方法效果较好，相比重心法，本方法采用直线拟合，更贴近实际车牌模型，结果自然更为精准，相比 Hough 变换，本方法的数据量大为减少，无需巨大的数据存储空间，运算更快，方法更加简便，实时性也更好，实验结果表明，与 Hough 变换相比，本方法会有 $1^{\circ}\sim 2^{\circ}$ 的误差，但后文将会看到，所得的最终结果已经可以与模板相匹配，所以该误差在可接受范围之内。

4.3 字符分割模块设计

字符分割为整个系统能获得正确结果的必要工作和准备，是字符识别的准备阶段，良好而适当的字符分割结果，对于整个车牌识别系统而言，能起到一个良好的由图像向文字转换的作用，其重要性不言而喻，下面，就字符分割的三个步骤逐一说明。

4.3.1 字符分割模块流程图

按照字符分割步骤，在接受车牌图像粗定位的结果之后，首先要对车牌的文本区域进行再一次定位，也就是车牌精定位，得到文本区域后，开始对文本区域进行分割，得到 7 张车牌字符的分割结果图，之后必须对字符的分割结果图进行归一化，使它们成为固定大小，最后输出归一化之后的字符分割结果图。

字符分割模块流程设计如图 4-24 所示。

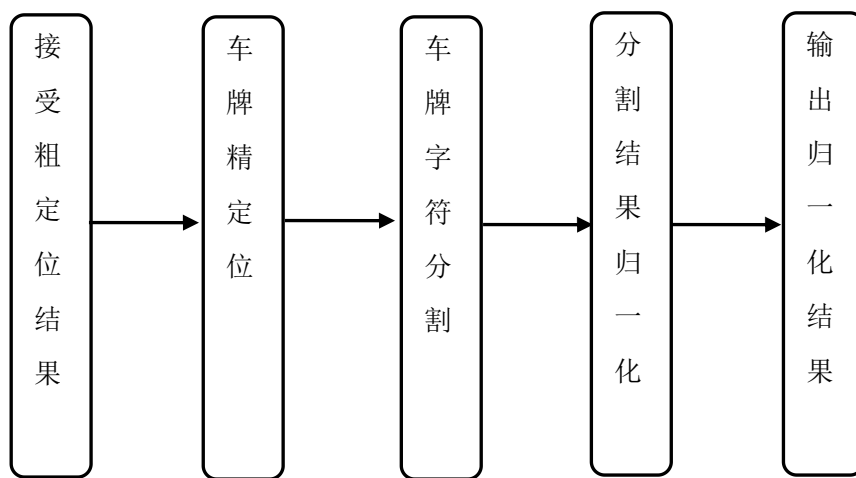


图 4-24 字符分割模块流程图

4.3.2 车牌精定位模块设计

车牌识别进行到这里，所得的已经是一个初步定位的车牌。但由图 4-18 可以看出，此时的车牌图像依旧存在大量不利于识别的因素——伪边缘，锯齿边缘，噪点，甚至有的车牌图像进行到这里，还会有大块的白斑。

究其原因是因为，车牌的图像往往是在夜间公路上拍摄的，在实际的条件下，存在林林总总的原因使得图像存在干扰，例如，有些车辆常年缺乏保养，导致车牌较脏，污染较严重。然而图像识别至此，下一步就要开始识别汉字，字母或者数字这样的文本，相对于标准文本，这些累计起来的污点尺寸并不小，有些甚至比文本本身还大。还有一些原因是拍摄硬件原因，例如存在于高空的公路摄像头，虽有定期维护，但是有的摄像头维护间隔比较长，镜头上已经沾上了不少污迹，这些污迹相对于车牌文本而言也会造成强干扰^[32]。

在这种情况下，本次系统设计对车牌进行了二次定位，这个步骤可以称为车牌的精定位。

车牌图像的精定位是为后续的字符分割与字符识别做准备的步骤，换言之，系统进行至此，我们的主要注意力已经可以由车牌本身转移到车牌上文本本身。

依据这一思路，本系统采用一种基于文字大小的车牌精定位方法，这种方法的全部目的就在于定位出文字位置，当该算法在系统中运行完成之后，就可以依据文字所处位置开始进行文字的分割。

车牌文字区域和背景区域的最大区别就在于二值图像上颜色跳动方面，文字区域密集的部分，笔画集中，二值图像上颜色跳动次数显著增多，而在车牌背景区域，车牌情况良好的情况下，二值图像上一直呈现为黑色区，即使有污染，污点所造成的图像颜色跳动也远远小于文字导致的颜色跳动次数。

于是算法设计上可以选定一个阈值 L ， L 是一个针对二值图像中颜色跳动的临界值，以它来确定当前所扫描的区域是不是车牌文本区域，首先统计每一行中的跳动次数，若该次数小于 L ，则超出了阈值范围，则可以认为是车牌背景区域，若该次数大于 L ，则在阈值范围以内，则可以认为是车牌文本区域。

本算法的核心步骤描述如下：

(1) 对于每一幅图像从上到下进行扫描，对于当前的一行，统计本行中从‘1’到‘0’和从‘0’到‘1’的变化次数，定下一个 A 向量，用 A 的各个分量来记录所对应行的跳动次数。

(2) 给定一个阈值 L，在 A 中寻找第一个记录次数大于 L 的行数 a。

如果连续十行都大于 L，那么，a 行就是车牌文本上界。否则继续向下遍历，直到找到 a 为止。

(3) 同第二步的阈值 L，在 A 中寻找最后一个记录次数大于 L 的行数 b。

如果连续十行都大于 L，那么，b 行就是车牌文本下界。否则继续向上遍历，直到找到 b 为止。



图 4-25 精定位以前的车牌（左）与精定位后的车牌（右）对比

该算法完成之后，车牌上下部分干扰完全去除，车牌左右部分依然存在噪声，但由于下面的字符分割算法，左右的噪声已经不会干扰到字符的识别。

4.3.3 字符分割模块设计

常用的字符分割方法为垂直投影法。垂直投影法是基于这样一个事实：车牌图像二值化之后，在文本区域的白色像素点随文本变化，与文本区域的白色像素点相比之下，在背景区域的白色像素点较少。在此事实的基础上，垂直投影法以列为基准，统计每一列中有多少个白色像素点，并记录每一列中的白色像素点的数目，以每一列上的白色像素点个数为纵坐标，以列数为横坐标，得出统计图，然后设定一阈值，凡有连续若干列均大于此阈值的统计区间，则认为该区间即是字符出现的区域，否则即为背景区域^[33]。

这里提出一种基于二次分割思想的字符分割算法设计

在夜间公路上，由于背景晦暗和采光较差这两个主要不利因素，车牌识别进行到这里往往所得的字符轮廓较窄，这主要是因为车牌定位的开始阶段，Log 算子是先进行了模糊才开始检测边缘，光线暗淡的情况下，边缘处像素值梯度的改变程度不均匀，

在这种情况下，较窄的轮廓进入投影之后，白色像素点的分布并不能完全代表字符的分布方式，所以，若是单纯的使用垂直投影，时常会出现边缘粘连，不利于后继的字符识别工作。

这里介绍一种二分投影法，这种投影法的主要思想是将垂直投影所得结果视为一个又一个解集，在解集中通过自适应阈值来进行优选和调整。

(1) 首先使用垂直投影法，直接对全图进行投影，得到 7 个分割结果，视每一个分割结果为一个解集，找到其中最宽的一个，记录该解集宽度为 L_Max 。

(2) 按照具体的情形选定阈值 $Thresh$ ，在本次系统设计中，此模块的 $Thresh$ 为 7 个分割结果的算术平均值乘以修正系数 1.3。

(3) 找到 L_Max 所对应的解集，若大于 $Thresh$ ，则再次对该解集进行垂直投影，找到该解集中白色像素点数最少的一个列向量 Low ，以 Low 列为基准，左右邻域内扫描，找到此区域内包含的无关字符与有效字符的边界，以此边界为基准分割该解集。

(4) 再次扫描所有分割结果，重复 (1) ~ (3)，当第 (3) 步找不到 Low 值之后，便终止循环。

(5) 列举所有已知的解，合并相关解，即得所需的字符分割结果。

(6) 输出结果图。

字符分割如下图所示。



图 4-26 字符分割结果

4.3.4 字符归一化

由于所得字符的图片大小大小往往不一致，为了标准模板大小的统一，往往需要重订所得字符图片的大小，这个大小重订的过程称为字符的归一化，本次字符归一化的大小取 40x20，即所得所有字符的图片最终都转化为 40x20 大小的图片。



图 4-27 字符归一化结果

4.4 字符识别模块设计

在中国，车牌识别系统较之国外同类产品，其最困难之处就在于汉字的识别，汉字笔画众多，且笔画排列没有固定的规律。这里将先介绍车牌字符识别的常用思路——特征提取和字符匹配，针对夜间公路路口这一特定条件，提出一种将二者相结合的思路来进行字符识别。

4.4.1 车牌字符识别常用思路分析

尽管汉字的图像具有很多不确定性，但是我国车牌中文本由于拥有自己的规格约束，还是呈现出一系列特征，有许多方法已经被证实可以有较好的识别效果，我国车牌文本的主要特征如下^[35]：

（1）中国车牌文本一共有 7 个字符，其中在一般使用场合下，多数车辆第一个字符为汉字，第二个字符为字母，剩下五个为字母和数字的随机排列。

（2）第一个字符一定为汉字，用以代表各省和直辖市，用在此处的汉字一共有 59 个，其他字符位可以认为无固定规律，但是英文字母一共仅 26 个，数字字符一共仅 10 个。

常用的车牌字符识别的思路有两种，模板匹配和特征提取，这里分别介绍。

基于模板匹配的字符识别方法是原理最简单但是应用最广泛的一种，其原理是，先用一个标准的规格来对分割出的字符进行归一化处理，再用预先设置好的标准模板来进行匹配。

模板匹配一般以二值图像为主，无论是由对象提取的归一化字符图像，还是所需匹配的标准模板，最终都应该以二值图像的方式呈现。对于车牌图像，一般采取的思路是，先将车牌做出一系列处理，得到图 4-27 中的结果，再将所得的图像和预先设好的二值图像相匹配，匹配的方法多种多样，典型的是图像相减，以图像是否

接近于一张全白点阵图为标准，最接近的结果即认为是匹配成功，输出该图结果。

该方法在图像质量优秀，字符分割准确，字符样式清晰的情况下结果较好，但实际的情况下，由于种种干扰，图像上噪点未能处理完毕，字符位置与标准模板不同，字符断裂，字符粘连，字符模糊都是常见的情形，故该方法实际是一种理想情形，实际效果较差。

与传统的模板匹配相比，基于统计量和特征的字符识别具有良好的鲁棒性，其方法主要是先由原始的字符分割图像提取出一系列特征数值，统计这些数值并构造决策函数，然后依据确定的决策函数来进行识别^[36]。它的鲁棒性主要体现在它将局部上的各种不利情形消除在最终的统计方法和决策函数中，具体的消除能力由统计方法和决策函数的合理性来决定。

这种方法的主要步骤是，先由字符图像的主要参数（包括像素块，字符轮廓，轮廓之间种种距离等等）提取出字符图像的特征，再由这些特征归纳出所需的统计量，最后依据该统计量和预先设置好的模板相应的统计量进行对比，来决定该字符识别的结果。此方法重点在于统计量的确定方式和匹配方式，这一点由各厂家依据不同情形自行决定，目前没有统一定论。

该方法在模板匹配的基础上做出了一定改进，但在实际应用中依旧很少单独使用，主要是因为该方法是基于这样一种假设——同一字符的所有分割图像都是相同的。但在实际的应用过程中，由于噪点，污迹等一系列不利因素，同一字符的分割图像也有可能不同，从而造成部分文字无法识别。

4.4.2 基于综合考虑的字符识别改进算法设计

由于我国车牌文本复杂，包含汉字，数字，英文字母三大类字符，而汉字字符又呈现出结构不一，笔画排列随机性大等特点，在实际应用的过程中，模板匹配往往导致二义性，需要一定的特征来进行辅助识别，故综合以上两种思路做出本系统的字符识别。

车牌文本中的数字和英文字母，由于其结构简单，样式单一，特征易于提取^[27]。

汉字结果复杂，样式繁多，特征不易提取，故将这两种字符分类，字母和数字的分割图像对应字母模板和数字模板进行识别，汉字的分割图像对应汉字模板进行识别。

本次系统具体的字符识别方法为：先用预先准备好的标准模板取匹配得到的字符图像，字符图像往往一个对应多个解，在相应的解空间之中使用特征匹配来进行优选。特征提取方法为：对于字母和数字，采取十三点特征提取，对于汉字，采取单独的特征提取，下面分别介绍这两种方法。

一种是字母和数字特征提取。对于数字和字母的特征，本次系统设计将它们多次划分为多个区域，先统计第一次和第二次分出的区域里的白色像素点的个数，再统计第三次分出的区域内二值图像像素跳动的次数，以此作为特征，具体步骤为：

(1) 创建向量 A ， A 中共有 13 个分量，分别为 $a_1, a_2, a_3, \dots, a_{13}$

(2) 将数字或者字母图像分为 2×4 一共 8 个小格，分别统计每个小格中的白色像素，取 A 的前 8 个分量作为记录数据的地址，即将它们分别记录到 a_1, a_2, \dots, a_8 中。



图 4-28 a_1, a_2, \dots, a_8 特征提取方法示例

(3) 将字母或者数字图像还原，在竖直方向上三等分之，记录两条贯穿字符的直线上所含有的白色像素点个数，取 A 的第 10 个和第 11 个分量作为记录数据的地址，即将它们分别记录到 a_9, a_{10} 中。



图 4-29 a_9, a_{10} 特征提取方法示例

(4) 将字母或者数字图像还原，在水平方向上三等分之，记录两条贯穿字符的直线上所含有的白色像素点个数，取 A 的第 12 个和第 13 个分量作为记录数据的地址

址，即将它们分别记录到 a_{11} ， a_{12} 中。



图 4-30 a_{11} ， a_{12} 特征提取方法示例

(5) 将字母或者数字图像还原，统计其中总共含有多少个白色像素，将其结果记录到 a_{13} 。

一种是汉字特征提取对于汉字的特征提取，本系统采用一种白色像素点数与跳动相结合的方法。

汉字结构复杂，不易区分，但有些特点是显而易见的^[40]，例如汉字的笔画是全随机的，但如果对于图像划分足够细致，则笔画在细小区域内所占的比例不同，通过这一点能够在全随机的过程中找到字与字之间的不同之处。

本次汉字特征提取算法的具体步骤为：

(1) 在第四章中已经将汉字的分割结果归一化到 40X20，这里我们先将汉字的分割图像划分为 8x4。

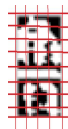


图 4-31 汉字特征提取示例——“鲁”字图像

(2) 创建一个 8X4 的向量 B。

$$B=\{b_{11},b_{12},b_{13},b_{14},b_{21},b_{22},b_{23},b_{24},\dots\dots b_{81},b_{82},b_{83},b_{84}\}$$

(3) 对于步骤 1 中第 m 行第 n 列的方格，开辟一个临时一维向量 $C(c_1, c_2, \dots, c_5)$ ，由上至下遍历，将第 i 列中由黑色像素点向白色像素点跳动的次数记录到 c_i $i=\{1,2,3,4,5\}$ 中，求取 $S=C_1+C_2+C_3+C_4+C_5$ ，并将 S 记录到 b_{mn} 中。

这里提出一种基于这两种常用思路改进而来的字符识别算法流程设计

模板识别应用于实际情况下的字符往往存在很多不利因素，如图像破损，车牌

污迹，字符断裂，字符粘连等等，纯粹的模板识别最终可能导致部分字符结果不准确或根本无法识别，于是本系统在完成模板匹配时，适当放宽了筛选条件，令其成为一个解集，然后结合所提取的特征，在解空间之中进行优选。

本系统字符识别部分流程如下：

第一个字符识别（汉字识别）：读取车牌字符分割结果,建立用于字符匹配的模板库,读取车牌字符分割结果,将第一个字符与汉字模板库相匹配，逐一相减，设定一个阈值 L ，在所有结果图片中，若图片中的非白色像素点个数小于 L ，则认为该图片可以选入解集。读取第一个字符的汉字特征与解集中所有解相应的特征，进行匹配，对于字符图片的特征矩阵 B ，与模板图片的特征矩阵 B^* ，列举所有 $R=B-B^*$ ，取所有 R 中迹最小的一个 R_{Min} 为最优解， R_{Min} 所对应的 B^* 即为所求解,输出 B^* 。

其他字符识别:读取车牌字符分割结果,建立用于字符匹配的模板库,将当前字符图像与数字/字母模板库中各个图像相减，设定阈值 L ，在所有结果图片中，若图片中的非白色像素点个数小于 L ，则认为该图片可以选入解集。读取当所有模板图片的特征与解集中所有解的特征，将特征向量逐一匹配相减，即对于字符图片的特征向量 A ，与模板图片的特征矩阵 A^* ，列举所有 $R=A-A^*$ ，所有 R 中向量内全部元素和最小的一个所对应的 A^* 即为所求解，输出 A^* 。

4.5 本章小结

本章第一节主要研究了车牌图像的图像增强，首先阐述了图像增强的一般表达式，并介绍了图像灰度化的一般方法，然后依据图像增强表达式在不同情况下的差异，依次给出了两种图像增强方法——高斯同态滤波和线性灰度变换，并给出这两种方法的实验结果，最后通过这两种方法在理论上的侧重点和实际实验的结果，给出了一个合适的选择。

本章第二节先阐述了图像边缘检测的一般方法，通过对一般性边缘检测理论分析和结果对比，针对夜间公路路口这一特点，使用 Log 算子来完成本系统边缘检测，然后使用数学形态学来对图像进行处理，通过一系列形态学操作确定车牌的边缘，然后通过一组相互垂直的向量分别遍历，依次确定了车牌四个方向上的边缘，最后，通过取点做直线拟合的思路，完成了对于车牌的校正。

本章第三节首先介绍了字符分割的概念，难点及重要性，然后介绍了字符分割的常用方法—垂直投影法，之后根据垂直投影法的缺陷给出了一种改进的字符分割方法并举出示例，最后列举出实验的结果，说明了算法的可用性和稳定性。

本章第四节首先介绍了车牌字符识别的概念，意义及难点，然后介绍了两种字符识别的思路—基于模板匹配的思路和基于特征提取的思路，从这两种思路出发，将两种思路相结合完成一种改进的模板匹配方法，最后给出了相应的实验结果和分析。

5 车牌识别系统的测试

本次测试，将 1000 张车牌图片作为测试用例，逐一输入系统，观测最后的输出结果。在最终所得的结果中，所识别得到的字符串与实际的车牌文本相一致则为正确，否则即为失败，针对失败的结果分析原因并作出改进，达到最初所预想的系统实现效果。

5.1 测试用例

本次测试分为两个部分第一个部分是依据识别步骤，按照第三到六章所述，逐步完成测试，观察每一步所得结果是否能正常进入系统后续工作。然后对系统总体输出结果进行测试，观察系统总体测试结果是否正确。

每一个测试步骤所定测试用例如下：

（1）车牌图像预处理模块测试中，输入车牌图像，观察所得结果，若是车牌与背景对比明显，且能用 Log 算子检测出较为完整的边缘，即边缘能正常包络车牌区域，则认为该用例测试结果为成功，否则即为失败。

（2）车牌图像定位模块测试中，输入预处理完成后的图像，观察所得结果，若是车牌文本能清晰完整的定位在图片里，不影响后继的车牌文本定位，则认为该用例测试结果为成功，否则即为失败。

（3）车牌图像字符分割模块测试中，输入车牌定位后的图像，观察所得结果，若是车牌所得字符清晰完整，字符在垂直方向和水平方向上位置正确，不影响后继的车牌文本识别，则认为该用例测试结果为成功，否则即为失败。

（4）车牌图像字符识别模块测试中，输入车牌字符分割后的图像，观察所得结果，若是车牌字符识别结果准确，7 个字符全部识别成功，则认为该用例测试结果为成功，否则即为失败。

（5）车牌识别系统测试中，输入车牌原始图像，观测整个系统所得结果，若是整个系统能成功识别出全部的 7 个字符，则认为该用例测试结果为成功，否则失败。

5.2 测试环境与配置

由于整个系统是在 SeeYourCar 智能交通系统平台上，SeeYourCar 平台已经集成一系列交通网络监控硬件和部分软件，并为其所有的子系统提供视频分解，三维重建，图像还原等一系列功能，故本次测试的环境无需考虑硬件层。

软件上，本次系统测试环境是 Windows7+VisualStudio2010。

5.3 车牌图像预处理模块测试结果与总结

在全部的 1000 张图片的车牌预处理模块的测试结果中，经 Log 算子检测后，边缘检测结果能正确包络车牌的车牌图像预处理结果有 999 张，不能正确包络车牌的有 1 张，正确率 99.9%。

发生错误的那一张图片主要原因是车牌沾有大量沥青，严重遮挡了车牌自身，属于个别情况。

综上所述，本次车牌预处理模块成功率较高，可以投入使用。

预处理部分测试结果如图 5-1 和图 5-2 所示。



图 5-1 图像预处理模块部分测试结果一(原始灰度图(左)，预处理结果(中)，Log 检测结果(右))



图 5-2 图像预处理模块部分测试结果二(原始灰度图(左)，预处理结果(中)，Log 检测结果(右))

5.4 车牌定位模块测试结果与总结

在全部的 1000 张图片的车牌定位模块的测试结果中，能正确包络车牌文本，不影响后继车牌文本识别的车牌定位结果有 987 张，不能正确包络车牌的有 13 张，正确率 98.7%。

发生错误的主要原因是部分车牌图像中，车牌区域沾有大量大块的污迹，这些污迹形成的噪声令整个检测到的车牌边缘向车牌内移动，使得所得边缘变小。

综上所述，车牌定位模块结果可靠，成功率较高，可以投入使用。

车牌定位模块部分结果如图 5-3 和图 5-4 所示。

（左边为原始图像，中间为车牌粗定位结果，右边为车牌矫正结果，若无需矫正则没有右图。）



图 5-3 车牌定位模块部分测试结果一



图 5-4 车牌定位模块部分测试结果二

5.5 车牌字符分割模块测试结果与总结

在全部的 1000 张图片的车牌字符分割模块的测试结果中，字符图片清晰完整，

字符本身在垂直方向和水平方向上位置正确，不影响后继的车牌文本识别的车牌字符分割结果有 957 张，不能正确包络车牌的有 43 张，正确率 95.7%。

发生错误的主要原因是部分车牌的车牌文本区域出现了大的污迹，这些污迹往往使得字符出现粘连断裂的状况，影响最终结果。

综上所述，车牌字符分割模块所得结果可靠，成功率较高，可以投入使用。

车牌字符分割模块部分结果如图 5-5 和图 5-6 所示。



图 5-5 京 KR6196 车牌字符分割结果



图 5-6 京 P553K3 车牌字符分割结果

5.6 车牌字符识别模块测试结果与总结

在全部的 1000 张图片的车牌字符识别模块的测试结果中，汉字图片共有 1000 张，成功识别的有 945 张，成功率 94.5%，英文字母和阿拉伯数字图片共有 6000 张，成功识别的有 5941，成功率 99.0%。

汉字识别发生错误的主要原因是因为汉字结构复杂，部分汉字在分割后出现变形，令识别失败。字母和阿拉伯数字的识别发生错误的主要原因是因为部分字母和数字上有噪声，令识别结果发生混淆。

综上所述，车牌字符识别模块所得结果可靠，成功率较高，可以投入使用。

车牌字符识别模块部分结果即是整个车牌识别系统的结果，部分测试结果如下一节图 5-7 和图 5-8 所示。

5.7 系统测试结果与总结

1000 张图片全部输入到车牌识别系统中，在全部的 1000 张图片的测试结果中，识别结果全部正确的有 896 张，正确率 89.6%，具备一定的实用价值，能参与到智能

交通系统平台的整体运作中，保证了在夜间公路路口上车辆车牌识别功能的完善性，达到预定需求。

点击识别‘预览’按钮，界面上各个板块按照各自所对应的模块结果进行显示，其中，左上图为智能交通平台通过路口终端摄像头采集的原始图像，右上为智能交通平台在完成一系列处理后传递给车牌识别子系统的图像，左下为车牌识别子系统结



果 列
表，右
下为车
牌定位
信息及
由智能
交通平
台通过
运算所
得一系

列相关结果。如图 5-7 和图 5-8 所示。



图
系

5-7
统整

体测试结果一

图 5-8 车牌识别界面设定二

5.8 本章小结

本章主要叙述了车牌识别子系统的测试，通过 1000 张图片的输入，逐一完成相应模块的测试和系统总体的测试，除去少量图片由于种种原因无法达标，绝大部分均已达到较好效果，验证了系统的可用性。

6 总结与展望

6.1 全文总结

文章针对夜间公路路口处车辆的识别问题，设计了智能交通平台下夜间公路路口处车辆车牌的识别系统，完成了车牌图像增强，车牌图像定位，车牌字符分割和车牌字符识别四个方面的内容并形成系统下的模块，全文的主要内容如下：

（1）在车牌图像增强阶段中，列举出同态滤波算法和线性灰度变换算法，比较了两者不同之处及公路背景的车牌在同态滤波算法下存在的问题，从而优选出线性灰度变换这一算法，通过特定的参数完成了图像增强，有效解决了夜间采光不足导致的车牌与背景对比度不够的问题。

（2）在车牌图像定位阶段中，介绍了若干种车牌定位的思路，通过 Log 算子来检测边缘，有效抑制了夜间车牌图像边缘模糊的特点，通过数学形态学来进行车牌粗定位，有效弱化了公路背景带来的定位影响，并使用一种更加简便的方式来定位，大大缩减了定位这一模块在时间和空间上的消耗。

（3）在车牌字符分割阶段中，详细阐述了一种以车牌文本区域的特点为出发点，并最终成功获得车牌文本区域的算法，通过这种算法有效完成了对于文字区域的提取，并提出一种二次分割的分割方法，使得模块的结果更加适合于字符匹配。

（4）在车牌字符识别阶段中，介绍了常用的识别手段，详细叙述了一种新的字符匹配方法，通过这种方法所得的识别结果更加准确。

6.2 展望

车牌识别即是模式识别和数字图像处理的重要交叉领域，也是现代智能交通系统的重要组成部分，同时由于国内外的情形差异，国内的车牌识别遇到了很多困难，本文虽然提出了改进措施，但实际的情况下依旧还是有局限性较强，不能完全满足使用要求的地方，例如：

(1) 在边缘检测中，结果往往并不尽如人意，一者是因为适应于不同背景不同车牌图像的阈值不同，而阈值的选择往往决定一个边缘检测算法的最终结果。二者系统中采用的 Log 算子在原始图像信噪比很低的情况下会出现边缘断裂。

(2) 在字符识别中，依据本文的算法，字符‘D’和‘O’的识别效率并不高，很容易混淆，这主要是因为该算法是以小格内像素变化次数为依据，形状结构均很相似的字符不易分清。

综上所述，算法依旧有很多需要改进的地方，我们既要抱着对于“未来系统会越来越完善”这一憧憬的坚定信念，也要脚踏实地的研究，找到不足并改进它。

致 谢

两年半的生活和学习中，我的每一点进步都离不开帮助过我的人们，在此我要衷心的感谢他们。

感谢我的导师，胡雯蕾老师在这两年半的时间里给予我莫大的帮助和鼓励，正是她教导我软件工程学习的正确方法，为我提供方向和道路，在我学习最为困惑的时候为我答疑解惑，在我论文最为困难的时候成为指引的明灯。胡老师严谨的治学态度，渊博的专业知识，奋发的人生精神无时不刻的感染我，指引我，激励我，令我这两年半的时光过得既充实又愉快。

感谢那么多教导过我课程的老师们，没有各位老师兢兢业业的教导，就没有我这两年半的收获，向你们致以由衷的感谢和敬意。

感谢学校的领导，是你们为学校默默的做出良好的管理，才有华科大今天孜孜以求的学风，才有我们这些学子赖益成才的校园净土。

感谢我的父母这么多年来对我努力的不懈支持和鼓励，正是你们的支持和鼓励，才有孩子的今天。

最后，感谢伟大的祖国和人民，正是有着和谐安定的大好局面，才有学子们安心学习的优越环境。

参考文献

- [1] 陶军. 车牌识别技术研究 with 实现: [硕士学位论文]. 江苏: 南京理工大学图书馆, 2014
- [2] 刘同焰. 车牌识别系统相关算法研究 with 实现: [硕士学位论文]. 广东: 华南理工大学图书馆, 2014
- [3] 顾钰彪. 车牌识别系统的研究: [硕士学位论文]. 宁夏回族自治区: 宁夏大学图书馆, 2014
- [4] 蒋传伟. 车牌识别系统中车牌定位 with 字符分割的研究: [硕士学位论文]. 湖北: 华中科技大学图书馆, 2014
- [5] 陈智慧. 复杂背景下车牌识别技术研究: [硕士学位论文]. 山西: 太原科技大学图书馆, 2014
- [6] 丁伟. 改进神经网络算法在车牌识别中的应用. 计算机仿真, 2011, 8: 359-362
- [7] 赵丹, 孙云莲. 基于 ICA 的车牌牌照字符识别系统. 武汉大学学报, 2005, 12: 208-210
- [8] 杨龙华. 基于 Matlab 的车牌定位及分割技术的研究 with 实现: [硕士学位论文]. 陕西: 西安电子科技大学图书馆, 2013
- [9] 贺瑜飞, 乔宝明. 基于小波变换的车牌定位算法研究. 中国电子商务, 2012, 4: 104-105
- [10] 朱皓. 基于易混淆字符集神经网络的车牌识别算法研究: [硕士学位论文]. 武汉: 华中师范大学图书馆, 2014
- [11] 特征提取 with 图像处理. 石英, 杨高波译. 北京: 电子工业出版社, 2010: 1-324
- [12] Discrete. Fourier. Analysis. and. Wavelets. Applications. to. Signal. and. Image. Processing, . Broughton, . Bryan, . Wiley, , 2008 S. ALLEN BROUGHTON KURT BRYAN. Rose-Hulman Institute of TechnologyDepartment of MathematicsTerre Haute, IN, 2010: 1-324
- [13] 李文举, 梁德群, 王新年等. 质量退化的车牌字符分割方法. 计算机辅助设计

与图形学学报, 2004, 16(5): 697-700

- [14] 冈萨雷斯. 数字图像处理(第 2 版). 电子工业出版社, 2007: 2-15
- [15] 胡小锋, 赵辉. VC++/MATLAB 图像处理与识别使用案例精选. 北京: 人民邮电出版社, 2004: 9-18
- [16] 郁梅. 基于视觉的车辆牌照检测. 计算机应用研究, 1999(5): 65-67
- [17] 叶晨洲, 廖金周. 一种基于纹理的牌照图像二值化方法, 1999(6): 28-29
- [18] 朱学芳等, 一种自适应细化方法, 模式识别与人工智能, 1997: 1140-145
- [19] 杨万山等基于 BP 神经网络的工程图纸图形符号的识别, 2000: 2100-106
- [20] 袁志伟, 潘晓露. 车辆牌照定位的算法研究. 昆明理工大学学报, 2001. 26(2): 56-60
- [21] 刘阳, 伊铁源. 数字图象处理应用于车辆牌照的识别. 辽宁大学学报, 2004: 65-68
- [22] 许志影, 李晋平. MATLAB 及其在图像处理中的应用. 计算机与现代化, 2004. 4: 130-142
- [23] HONG-SHANG WENG, JING-HUA WANG. Contrast Enhancement Based on Divided Histogram Manipulation , 2000, IEEE Conf on Systems, Man and Cybernetics. Vol2, 2000
- [24] Hu PF, Zhao YN, Yang ZB, et al. Recognition of gray character using Gabor filters. Proceedings of the Fifth International Conference Information Fusion, 2002
- [25] Barroso J, Refael A. Number Plate Reading Using Computer Vision, IEEE Transaction on Vehicular Technology, 2001
- [26] Hegt HA, Dela RJ, Khan, NA. A high performance license plate recognition system. IEEE Transaction on Pattern Analysis and Machine Intelligence, 1998
- [27] Amit Y, Geman D, Fan X. A coarse-to-fine strategy for multi class shape detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004
- [28] Nijhuis JAG, Ter Brugge MH, Helmholt KA, Pluim JPW, Spaanenburg L, Venema RS, Westenberg MA. Car license plate recognition with neural networks and fuzzy logic. Proceedings of IEEE International Conference on Neural Networks, 1995

-
-
- [29] Gary Bradski , Adrian Kaehler, Learning OpenCV: Computer Vision with the OpenCV Library, 2004, 51(1): 2-45
- [30] aniel Lélis Baggio , Shervin Emami , David Millán Escrive , Mastering OpenCV with Practical Computer Vision Projects, 2010, 30(1): 97-135
- [31] 崔屹, 图像处理与分析——数学形态学方法及应用, 北京: 科学出版社, 2000: 1-59
- [32] 程正行, 杨守志, 冯晓霞. 小波分析的理论, 算法、进展和应用. 北京: 国防工业出版社, 2007: 1-11
- [33] BarroSo J, Dagles s E L, Rafe1 A, et al. Number plate reading USing computer vi siOn [C]. Proc 1EEE International Symposium on Indust rial Elect rollics. 1997
- [34] Nomura S, Yamanaka K, Katai O, etal. A Novel Adaptive Morphological Approach for Degraded Character Image Segmentation. Pattern Recognition, 2005, (11): 1961-1975
- [35] 莫海宁. 自然条件下车牌字符识别方法的研究: [硕士学位论文]. 哈尔滨: 哈尔滨工业大学图书馆, 2006
- [36] HE Bin et al. , Visual C++ Digital Image Processing, Posts & Telecom Press, 2001, 4: 473-477
- [37] Storn R, Price K. Differential Evolution—a Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Space. International Computer Science Institute, Berlaey, 1995
- [38] Tubbs J D. A note on parametric image enhancement. Pattern Recognition. 1997, 30(6): 617-621
- [39] 黄山车牌识别技术的研究和实现: [博士学位论文]. 四川: 四川大学图书馆, 2014
- [40] 田晶车牌识别算法研究: [硕士学位论文]. 西安: 长安大学图书馆, 2014
-