

分 类 号: TP 31  
研究生学号: 201351E037

单 位 代 码: 10183  
密 级: 公 开



# 吉 林 大 学

## 硕士学位论文

基于数字图像处理技术的车牌识别系统研究

License Plate Recognition System Based On Digital Image

Processing technology research

作者姓名: 隋君

专 业: 电子与通信工程

研究方向: 数字图像处理

指导教师: 顾玲嘉 教授

培养单位: 吉林大学电子科学与工程学院

2016 年 9 月



基于数字图像处理技术的车牌识别系统研究

**License Plate Recognition System Based On Digital**

**Image Processing technology research**

作者姓名：隋君

专业名称：电子与通信工程

指导教师：顾玲嘉 教授

学位类别：工程硕士

答辩日期：2016 年 12 月 03 日

未经本论文作者的书面授权，依法收存和保管本论文书面版本、电子版本的任何单位和个人，均不得对本论文的全部或部分内容进行任何形式的复制、修改、发行、出租、改编等有碍作者著作权的商业性使用（但纯学术性使用不在此限）。否则，应承担侵权的法律责任。

### 吉林大学博士(或硕士)学位论文原创性声明

本人郑重声明：所呈交学位论文，是本人在指导教师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，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：师君

日期：2016 年 12 月 3 日

## 摘要

### 基于数字图像处理技术的车牌识别系统研究

随着社会经济和科学技术的不断发展,集信息技术、通信技术、传感技术和控制技术于一体的智能交通系统(Intelligent Traffic System)已愈近成熟。作为其重要组成部分的车牌识别技术已成为当前的研究热点,许多研究人员都对此项技术进行深入的研究。车牌识别融合图像处理,模式识别等多门技术,多应用于停车场车辆管理系统、城市十字路口的“电子警察”和高速收费管理系统等领域,节省了大量的人力和物力资源,对国家的安全发展意义重大。

本文在分析了车牌识别系统的基础上,主要包括数字图像预处理算法研究、车牌定位算法研究、车牌字符分割算法研究和字符识别算法研究。搭建了一套完整的车牌识别系统,并利用 MATLAB GUI 界面完成整个车牌识别的流程和显示。在图像预处理方面,首先需要将彩色图像转化为黑白图像,然后分别进行拉伸、均匀化和中值滤波等方面的处理,并进行了二值化处理。在车牌定位方面,本文研究了前人根据颜色特征以及纹理特征的车牌定位的方法基础之上,结合两种方法的优势,提高了定位的精度,并有效的降低了定位所需要的时间。在字符分割方面,仍然在充分分析了前人所提算法的基础上,分别添加了标准归一化,倾斜几何校正和去除边框等方法,很好的进行了字符的分割。在车牌字符识别方面,在研究其他几种的字符识别的方法之外,本文详细介绍了基于 BP 神经网络的字符识别方法,并随机选取了 60 张车牌照片进行识别,识别正确率达到 90%。

关键词: 数字图像处理 ;BP 神经网络;字符识别;特征提取

## **Abstract**

### **License Plate Recognition System Based On Digital Image Processing technology research**

With the continuous development of social economy and science and technology, information technology, communication technology, sensing technology and control technology in one of the intelligent transportation system (Traffic System Intelligent) has been more and more mature. As an important part of the vehicle license plate recognition technology has become the current research hot spot, many experts and scholars both at home and abroad have carried out in-depth research on it. The license plate recognition based on image processing and pattern recognition technology, is widely used in the field of parking lot management system, city crossroads e-police and save a lot of manpower, high-speed toll management system and material resources, the development of the semantic security of the state's major.

In this paper, the four key steps of license plate recognition system, digital image preprocessing, license plate location, license plate character segmentation and character recognition, are studied. A complete license plate recognition system from image acquisition to image recognition is built, and the process and display of the whole license plate recognition is completed by using GUI MATLAB interface. In the aspect of image preprocessing, this paper firstly transforms the color image into black and white image, and then it is processed in the aspects of drawing, uniformity, median filtering, etc., and carries out the two value. In the aspect of license plate location is studied in this paper according to the previous method based on the license plate location color feature and texture feature, combining the advantages of the two methods, to improve the accuracy of positioning, and reduce the time required for the position. In character segmentation, it is still on the basis of the previous algorithms, which add the standard normalization, the tilt geometry correction and the removal of the frame, and so on. In the license plate character recognition, in depth study of the method of character recognition based on several commonly used, I introduces the character recognition method based on BP neural network is used, and randomly selected 60 pieces of photos for license plate recognition, the correct recognition rate reaches 90%.

Keywords:

digital image preprocessing;BP neural network; License plate recognition; Feature extraction .

## 目 录

|                           |    |
|---------------------------|----|
| 第一章 绪论.....               | 1  |
| 1.1 研究背景及意义.....          | 1  |
| 1.2 国内外研究现状.....          | 1  |
| 1.3 车牌识别系统的构成.....        | 3  |
| 1.4 本文的主要工作及章节安排.....     | 5  |
| 第二章 系统整体方案设计.....         | 7  |
| 2.1 系统的基本原理.....          | 7  |
| 2.2 硬件设计.....             | 7  |
| 2.3 软件设计.....             | 8  |
| 第三章 车牌定位.....             | 8  |
| 3.1 车辆图像预处理.....          | 10 |
| 3.2 基于边缘的定位方法.....        | 17 |
| 3.3 基于颜色的定位方法.....        | 18 |
| 3.4 车牌倾斜矫正.....           | 19 |
| 3.5 本章小结.....             | 22 |
| 第四章 车牌分割.....             | 23 |
| 4.1 车牌字符的规格.....          | 23 |
| 4.2 字符分割方案分析.....         | 23 |
| 4.3 边框的去除.....            | 23 |
| 4.4 单个字符的分割.....          | 25 |
| 4.5 字符归一化.....            | 27 |
| 4.6 本章小结.....             | 27 |
| 第五章 车牌字符识别.....           | 28 |
| 5.1 引言.....               | 28 |
| 5.2 车牌字符识别常用方法.....       | 28 |
| 5.3 BP 神经网络的结构.....       | 31 |
| 5.4 BP 神经网络在车牌识别中的应用..... | 32 |
| 5.5 实验结果.....             | 35 |
| 第六章 总结与展望.....            | 40 |
| 6.1 总结.....               | 40 |

|                      |    |
|----------------------|----|
| 6.2 展望及下一步工作计划 ..... | 40 |
| 参考文献.....            | 42 |
| 致谢.....              | 45 |
| 附 件: .....           | 46 |



## 第一章 绪论

### 1.1 研究背景及意义

伴随着经济的持续发展,人民生活质量的日益改善,汽车数量的急速增加导致道路交通越来越拥挤。在这种情况下,还依靠投入大量的资金到公路等交通基础设施的建设之中,已经远不能解决高交通事故发生率、环境污染、交通拥堵等一些日益严重的问题。此外,由于城市空间的限制、修建新的道路所需要的巨额资金以及环境的压力,建设更多基础设施必然受到严格的限制。因此,如何改善道路交通现状已经成为现代交通系统中刻不容缓的问题。所以,要想从根本上解决问题必须利用高新技术(如计算机、网络、通信技术、自动化技术等)来改造现有的道路运输系统及其管理体系。此时,智能交通系统(Intelligent Traffic System, 简称 ITS)顺应时代要求出现了,它大大地改善了道路交通的现状<sup>[1-3]</sup>。

在很多公共场合,车牌识别系统被大量的应用,例如测速系统、收费系统等等。目前可使用的汽车图像检测方法有很多,其中利用率比较高的是:使用红外探测装置进行检测的方法、使用超声波装置识别的方法、使用感应线圈检测等等。

### 1.2 国内外研究现状

起初,智能交通系统(Intelligent Transportation Systems, ITS)是由智能车辆道路系统(Intelligent Vehicle Highway System, IVHS)发展而来<sup>[4]</sup>。智能交通系统是一个实时的综合管理系统,它涉及了计算机技术、控制技术、信息技术等领域的知识,应用于各种场合<sup>[6]</sup>。

自 20 世纪 80 年代开始,人们就已经开始利用图像处理技术领域的相关知识来解决车牌识别的问题,只不过此时的研究还处于起步阶段,未能达到理想的效果。直到 20 世纪 90 年代初,随着计算机视觉的发展,ITS 才开始成为一个正式的具有一定意义的名词。进入 90 年代后,美国、欧洲、日本、加拿大、韩国、新加坡和香港等国家和地区为了改善越来越严峻的道路交通情况,开始投入巨额资金和大量精力到 ITS 的开发和研究中,这一举措极大地推动了 ITS 的深入研发。在当今世界的交通领域里,智能交通系统的发展始终代表着最先进的课题研究方向,发达国家正在深入研究和开发车牌识别系统,目的是治理交通问题并改善环境,运用传感器、计算机等领域的知识解决难题更好地实现现代道路交通系统的智能化。不可避免地,我国也正在大力地深入研发智能交通系统,并将其作

为交通运输领域的长期的研究课题<sup>[7-10]</sup>。智能交通系统被广泛地应用于各种公共场所,例如停车场、机场、公路实时管理系统等等。ITS 在全世界范围都被大量应用,其中包括一些发达国家和地区,例如美、日、欧、北上广(国内的北京、上海、广州)。ITS 是一个融合了多个子系统的综合性体系,其中包括了监控系统、控制系统、运营管理系统等。

在国内外研究人员的深入研发,车牌识别技术自提出到发展至今,已经形成了比较完备的体系并得到了广泛的应用。但是由于外部环境的影响、摄像技术有待提高、清晰度不够带来的问题、破损污旧车牌识别难度较大、车牌缺乏统一的标准等不利条件的影响,导致车牌识别难以达到预想的识别效果,而且由于通常需要大量数值计算,难以满足实时性上的要求。国外研究人员很早就开始致力于研发车牌识别系统,目前国内研发出的 ITS 还不如国外的先进。

直到今天,国外研发出的 ITS 已经比较完备,例如日本开发的 VICS 系统以,香港的 Asia Vision Technology 公司推出的 VECON 等等。我国的汽车牌照中包含汉字、大写的英文字母和阿拉伯数字,识别难度比国外的汽车牌照高得多,所以国外的车牌识别系统不能适用于我国的车牌识别。

我国研究车牌识别系统比国外晚了将近 20 年左右,但是我国的研究成果也不容小觑,郭杰,施鹏飞研制出了一种精准的定位算法,这种方法借助车牌的颜色信息和纹理特征实现功能;赵学春,戚飞虎采用颜色信息对待分割物体进行字符分割,研究出了一种高效的自动识别技术。此外,国内的许多科研单位已经开发出了比较完备的车牌识别系统,例如亚洲视觉科技有限公司,北京弗雷德科技有限公司等<sup>[11]</sup>。

近年来,在国内,有一些大学和科研机构也对于本课题的研究进行了尝试。从目前已经发表的论文来看,主要有上海交大、浙大、同济大学等院校正在研究这一课题。伴随着模式识别和信息技术的发展,车牌识别系统的发展日臻完善,本论文采用了一些近几年才提出的算法,例如利用数学形态学算法对车牌进行边缘检测,采用 BP 神经网络等等。但是因为该课题研究存在着一定的难度,并且车牌易受噪声和光照等干扰条件的影响,所以到现在为止,还未研发出一套极为完备且识别率高达百分百的算法。随着对车牌识别系统迫切需求的不断增加,人们对车牌识别率的期望值也不断变高,目前,对该领域的研究仍然很活跃<sup>[12]</sup>。

根据目前已应用车牌识别系统的场合的反馈情况来看,其存在的不足之处主

要是车牌识别时，执行时间太长导致实时性不高以及识别效果不理想。因此，车牌识别率的高低和执行时间的长短是未来车牌识别技术中亟待解决的问题和两个重点研究技术。它们是衡量 ITS 的性能优劣的重要指标，它们受系统的硬件性能和识别算法的制约。鉴于本文研究的基于软件仿真基础的车牌识别技术，所以只需考虑综合优化的算法来改善系统性能即本文研究的关键技术就是车牌的识别率和执行速度。

● **目前市场中车牌识别系统主要应用范围：**

- (1) 整个城市交通十字路口的实时监控与管理；
- (2) 高速公路上交通信息采集；
- (3) 各高速公路口的自动识别和收费系统；
- (4) 城市环境小区内、停车场、星级宾馆的汽车出入口管理；
- (5) 高速道路治安卡口抓拍识别，车速限定。

### **1.3 车牌识别系统的构成**

车牌识别技术是一门图像处理、数据压缩以及模式识别等方法的综合性技术，其核心技术就是把准确地提取出图像中的车牌，并且正确地识别出车牌。ITS 涉及到原始图像识别、图像预处理、车牌区域的定位、车牌区域的字符分割以及车牌区域的字符识别等几个模块，本论文研究的基于数字图像处理的车牌识别系统设计的识别流程如图 1.1 所示<sup>[13-19]</sup>。

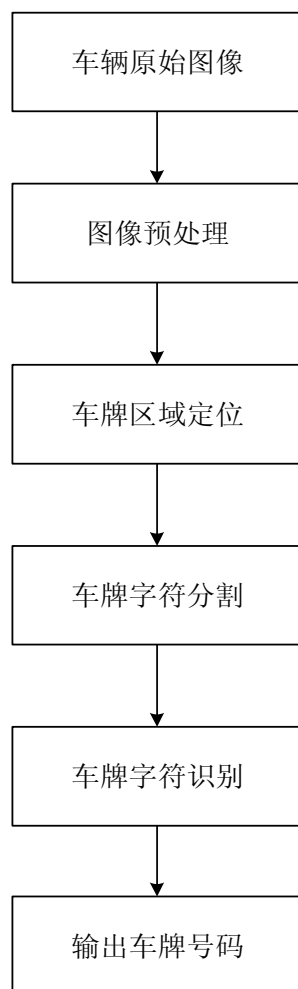


图 1.1 车牌识别流程图

本文通过 MATLAB 软件设计了一个车牌识别的仿真系统，该程序成功地达到了准确识别实际车辆照片中的车牌的目的，其中不包括硬件采集部分。

本仿真系统所需实现的功能分别为：

(1) 图像预处理：可以利用智能手机自带的相机功能在各种不同的环境下拍摄车牌照从而获得待识别的汽车牌照的图像，这一过程不可避免地存在着一定的噪声等不确定的干扰因素，为此，必须采用车牌图像预处理技术对车牌进行处理，如灰度化、滤波、灰度拉伸、二值化、矫正等，为下一步定位车牌做好充分的准备。

(2) 车牌定位：包含车牌的粗定位和精确定位，以及裁剪出车辆图像中的车牌区域的图像的功能，这一步比较重要，正确的定位有助于识别率的提高。

(3) 字符分割：由于车牌中字符可能会出现不同程度的倾斜，所以有时需要对倾斜的车牌进行矫正，然后把车牌中的字符精准地分割成单个的独立字符。

(4)字符识别：将分割处理后的所有独立字符逐一识别，使它们从原来的图像字符变成文本字符。通过编写程序实现功能，最终显示出车牌号码。其中每个过程都有不同的算法，不同的算法实现的效果都不尽相同。本论文研究的车牌识别系统是在 MATLAB 软件上进行仿真实现识别功能的。系统设计结构图如图 1.2 所示<sup>[20-21]</sup>。

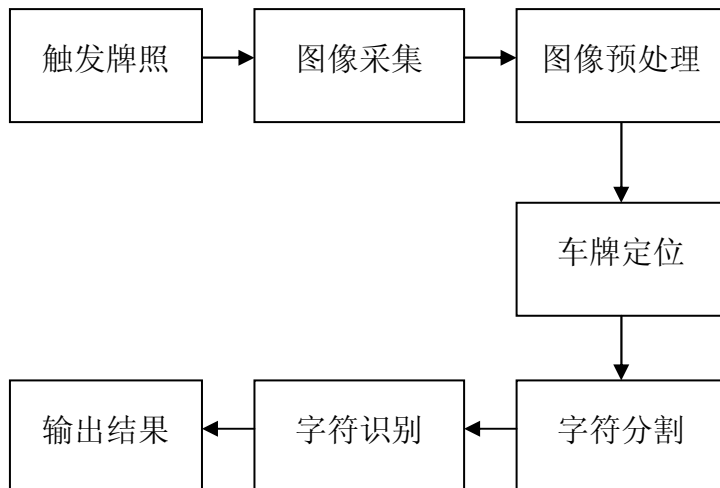


图 1.2 系统设计结构图

#### 1.4 本文的主要工作及章节安排

本文在分析前人在车牌识别系统的研究基础上，深入研究了如何实现车牌识别系统中各功能模块的算法，本文利用 MATLAB 软件平台，成功编写了程序实现上述功能。本论文的章节安排如下：

第一章首先介绍了本课题的主要研究背景及意义，然后分析了在车牌识别系统上国内外的发展现状，主要介绍了车牌识别系统的各个功能模块和其作用。

第二章主要从总体方面，介绍了车牌识别系统的整体设计方案，包括硬件设计方案和软件设计流程。

第三章主要介绍了车牌定位方面的内容，在车牌定位之前需要先进行车辆图像的预处理过程，并介绍了几种常用的车牌定位算法。

第四章主要介绍了车牌分割方面的内容，包括：车牌图像的二值化，车牌倾斜校正、字符归一化等一系列功能。

第五章主要深入研究了字符识别方法和优化办法，在此首要介绍了几种常用的字符识别算法，本文采用了 BP 神经网络的字符识别算法，得到较高的识别正



确率。

第六章主要是对本课题的总结和对下一步工作的展望。

## 第二章 车牌识别系统整体设计

### 2.1 系统的基本原理

车牌识别系统的摄像头通过对经过指定区域的机动车辆进行拍照，因为照片会受到光照、拍摄位置和车辆行驶速度的影响，导致拍摄的图片不能准确的确定汽车的车牌。而车牌识别系统就通过对机动车辆的照片进行图像预处理、车牌定位、字符分割、字符识别等技术手段，从而得到清晰的机动车牌照的照片，从而提高现代智能交通的管理效率，可以说车牌识别系统对于现代智能交通至关重要。

具体步骤如图 2.1 所示：

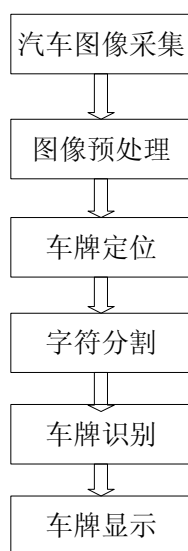


图 2.1 系统基本原理示意图

### 2.2 硬件设计

系统的硬件主要是对车辆号码牌进行图像采集的设备，其主要作用就是对车辆号码牌进行采集和存储，车牌识别系统的基本硬件配置由 PC 主控机、高速摄像机、照明系统和图像采集设别。

本文所设计的车牌识别的硬件系统主要用来探测车牌上字符的识别，首先需要通过红外或者微波雷达确定前方车辆的所在位置和行驶状况。然后为了能够更好获得车牌的图像，需要在高速摄像机旁安装照明系统。并能通过外界环境实时控制曝光量。为了能够增强整个系统的电磁兼容能力，增强抗干扰能力，利用

红外探测器或电磁感应环形探测器，并利用编码调制信号。这样就能够保证摄像系统得到一个较好的车辆车牌的原始图像。

本课题主要软件的研究即对 matlab 的运用，主要工作是设计软件，对已摄取到的车辆照片进行一系列处理，实现车辆号码牌的识别。

### 2.3 软件设计

本文在软件部分主要采用 MATLAB 平台对硬件设备采集到的图像进行处理，进而达到识别车牌图像号码的目的。现阶段利用的硬件平台摄像机或者高速相机进行采集图像。之后我们需要将采集到的图像进行图像格式转换，方便后续的图像预处理。目前常用的图像格式主要有 \*.PNG、\*.GIF、\*.TIF、\*.JPEG、\*.JPG、\*.BMP。本文选择利用 MATLAB GUI 平台进行软件编程，因为 MATLAB 具有以下优点：(1)、用户使用方便 (2)、编程效率高 (3)、移植性和开放性很好 (4)、扩充能力强，交互性好 (5)、语句简单，内涵丰富 (6)、方便的绘图功能。

图像采集主要通过摄像头或数码相机等采集设备拍摄车牌图像来获取的。由于原始图像常常是彩色图像，这样使得其中包含了大量的彩色信息，需要较大的存储空间而且图像的尺寸大小一般不一致，在处理时往往也会降低系统的执行效率，因此需要把彩色图像转变成灰度图像。

同时由于图像在采集的过程中，在车牌图像的周围环境的多变性和车辆的速度等多种因素的影响，以致车牌图像的质量不高，存在着不同程度的干扰及噪声。因此，在车牌图像的定位前，对图像质量进行提高而进行图像预处理的工作，不仅能消除噪声、改善图像质量，而且还可以有效地减少系统存储空间的需要。

车牌定位是整个车牌识别系统对图像处理较为重要的第一步，能否正确定位出车牌位置直接影响到车牌识别率。车牌定位就是从原始的车辆照片上根据车辆牌照的长宽比和颜色组合信息等特征来检测出面积尽可能小而完全包围车牌区域的图像，并将其从原始图像中分割出来。现实环境下，不同的光照以及不同的拍摄距离、角度的影响，采集到的车牌图像可能模糊不清甚至文字笔画不连续，所以要排除干扰，准确地定位出车牌位置有一定的难度。

车牌的字符分割就是指将车牌号码的 7 个字符从定位出的车牌图像中一个一个地分割并提取出来。字符分割的依据是字符边缘、字符的高宽比、字符间间距等特征信息。这里简要介绍投影法和模板匹配法。投影法包括水平投影法和垂

直投影法，通常情况下，使用垂直投影法来进行车牌字符分割。垂直投影法就是将字符在垂直方向进行投影，然后利用在该方向上的投影，字符间和字符内的间隙能取得局部最小值的方法，切割局部最小值附近，就可以实现字符分割。水平投影的方法就是对图像从上至下进行逐行扫描，当扫描到的像素点开始出现白色时，记录此时的位置，然后再从下往上扫描，同样地，当遇到第一个白点时，记录位置，从而基本可以确定出图像的高度范围。用同样的方法，左右方向扫描可以确定出图像的宽度范围。

车牌识别系统中车牌字符识别模块是最为重要的核心部分<sup>[23]</sup>。车牌字符识别就是要将分割出的字符图像识别成文本字符并将其输出，它包括了字符预处理、字符特征提取、分类匹配判决三个环节，其中分类匹配判决最为重要。车牌字符识别不同于普通的字符识别，车牌有自己独特的特征，如高宽比、每个位置出现的字符包括了汉字、阿拉伯数字和大写的英文字母。目前，国内外可使用的车牌字符识别方法有：模板匹配法、神经网络法以及 SVM 等算法。软件系统的设计框图如图 2.2 所示。

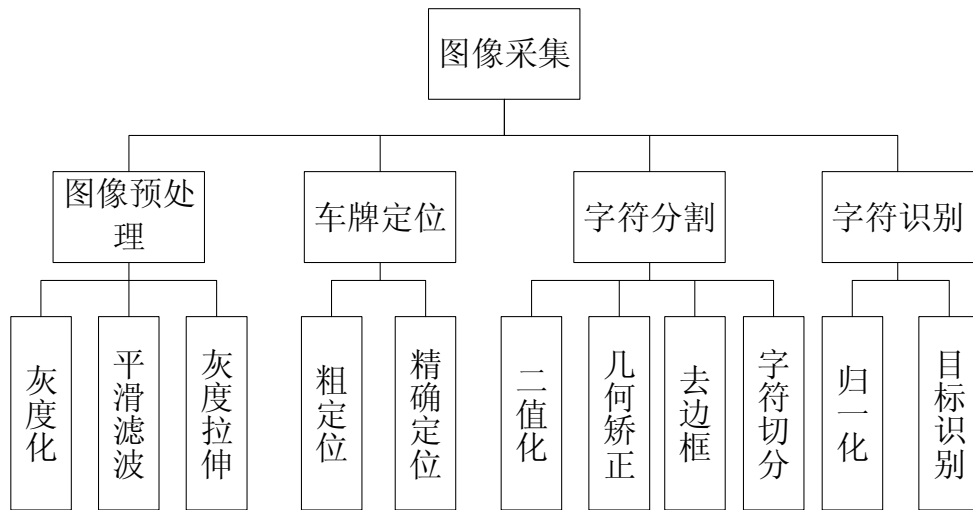


图 2.2 系统软件设计框图

## 第三章 车牌定位

车牌定位是整个车牌识别系统的首要,它能够定位的好坏直接影响整个系统车牌识别率的高低,如果车牌定位错误,那么后面所做的所有工作都是无效的,并且车牌定位的快慢也对整个系统的识别速度有很大影响。车牌定位是指将拍摄到的整个图像中切分出车牌部分,一般分为图像预处理、图像搜索和车牌定位三个部分。

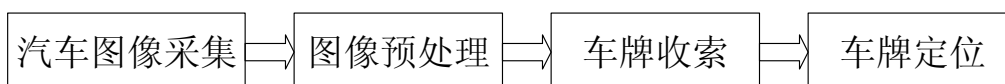


图 3.1 车牌定位程序框图

### 3.1 车辆图像预处理

预处理的第一步是要对图像的格式进行转换及尺寸进行归一化处理,图像预处理能减少提取图像特征时的计算量,并且能避免存储空间的浪费,提高利用率和实时性。图像格式转换和尺寸变换有几种方法:方法 1 是单纯用手工操作图像处理类的专业软件来调整格式和大小;方法 2 是加入相应的处理模块到系统中,然后根据需要系统自动处理完成。因为不同的图像处理类的软件采用的算法不完全相同,所以处理出的效果虽然表面看不出什么差别,但是其在特征匹配时的差异非常大,有的时候甚至可能导致图像失配。因此,应该尽可能采用统一的尺寸变换和格式转换算法。

#### 3.1.1 图像灰度化

一般情况下,用相机拍摄到的车牌图像都是彩色图样的,为了加快识别速度,需要先对图像进行灰度化处理。彩色图、样又被称为 RGB 图像,它是用 R、G、B 三个字母来表示一个像素的颜色的,R、G 和 B 可以分别代用来表红色、绿色、蓝色,通过组合这三个基本颜色就可以合成出任意的颜色。那么,对于一个大小为  $M \times N$  的彩图而言,就需要用一个  $M \times N$  在乘以 3 的三维数组来存储这样的图像。

彩色图像中存在不少颜色信息,不仅需要较大的空间进行存储,还会大大延长系统的识别时间,使实时性不达标。因为彩图的每个像素都包含着 R、G、B 三种不相同的分量,识别时,不需要使用其中很多无关信息,这也不利于进一步识别。所以,在进一步地处理图像之前,需要将图像由彩色转换为灰度,目的就是为加快处理的速度和节省存储的空间。在 RGB 的模型中,若 R 三个值都相



等，那么它代表一种灰度的颜色，而相等的那个值就被称为灰度值<sup>[8]</sup>。上面介绍的这个将彩图变为灰度图的过程就是这一小节的重点对图像进行灰度处理。将采集到的车牌图像进行灰度化处理后，此时的车牌图像值包含亮度信息，不包含彩色信息<sup>[9]</sup>。一个二维的数据矩阵就可以实现存储和处理一个灰度图像的功能，矩阵中的所有独立元素代表相应像素的灰度值。灰度的图像的像素色是  $RGB(r,r,r)$ ，彩色的图像的像素色是  $RGB(R,G,B)$ ，彩色图像的颜色可分解为  $R$ 、 $G$ 、 $B$ 。 $R$ 、 $G$ 、 $B$  取值的范围和灰度图像的转换方法如下：

(1)最大值法：取  $R$ 、 $G$ 、 $B$  中的最大值并将其赋给其他两个分量，即：

$$R = G = B = \max(R, G, B) \quad \dots \dots \dots (3.1)$$

(2)平均值法：使  $R$ 、 $G$ 、 $B$  的值等于三个值的和的平均值，即：

$$R = G = B = \frac{R + G + B}{3} \quad \dots \dots \dots (3.2)$$

(3)加权平均值法：按照某一指标给  $R$ 、 $G$ 、 $B$  赋权值  $W_R$ 、 $W_G$ 、 $W_B$ ，并使  $R$ 、 $G$ 、 $B$  的值等于它们的值的加权平均值，其中  $W_R$ 、 $W_G$ 、 $W_B$  为 0.3，0.59 和 0.11，即：

$$R = G = B = \frac{0.3R + 0.59G + 0.11B}{3} \quad \dots \dots \dots (3.3)$$

一般情况下，人类的双眼最容易辨别出绿色，较容易三个分辨出红色，但是不容易辨别出蓝色，所以通常利用第三种方法进行灰度化处理。经转换后的图像只包含强度信息，因此灰度模型不能重新转换为 RGB 模型。

经灰度化处理之后的彩色图像保存了图像的主要信息，不会丢失车牌的主要特征信息，同时图像经灰度化处理后为车牌的字符分割提供了简单的处理对象。

车牌灰度化处理前后图如下：



(a) 原始彩色图像



(b) 经过灰度化之后的图像

图 3.2 原始彩色图像和经过灰度化之后的图像

### 3.1.2 图像二值化

在研究车牌识别系统时需要多次利用灰度图像的二值化算法,如图像灰度化后的图像增强阶段、对车牌区域进行粗定位阶段、精定位之后的字符分割阶段等,可以使用不同的二值化算法。二值图像指图像的灰度值只有 0 和 255 的图像,不存在其他灰度值。原因有两方面:一、一些待处理的图像本身就是二值图像,如工程图像、指纹图像等;二、尽管有些图像是灰度的,为了提高处理速度,也会将其转化为二值化的图像。此外,二值图像能应用几何学中的特征描述,具有巨大的优势。

图像的二值化是根据图像中待提取目标与背景之间灰度差异计算出一个阈值,然后图像中的各个像素的灰度值与这个阈值的大小进行比较,然后将大于阈值的像素归为一类并用‘1’来表示,小于阈值的则为另一类用‘0’表示<sup>[12]</sup>。这种理论基于背景与物体的灰度值存在差异。因而,阈值分割的算法分为两步:一是计算出阈值;二是做对比然后取值。确定阈值是以上步骤中关键的一步,恰当地选取好阈值,可以有效地除去噪声,并且可以将图像中的目标和背景分割开,从而大大减少信息量并提高处理速度。

目前确定阈值的方法有多种,依据阈值的应用范围常用的有以下三种方法:

#### 1、直方图技术

如果在一幅图像中,背景是浅灰色的,但其中包含一个深色像素的物体,这类图像的双峰直方图如图 3.3 所示。灰度级的两峰之间存在一个灰度级的数量极少的一个部分,从而产生了谷,以谷值作为灰度阈值。

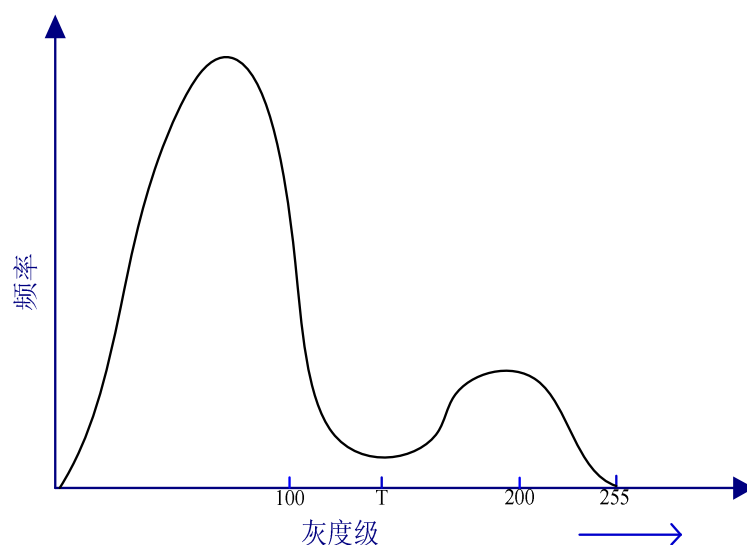


图 3.3 双峰直方图

二值化公式如下：

$$g(x,y)=\begin{cases} A, \text{if } f(x,y) \leq T_1 \\ B, \text{if } T_1 < f(x,y) < T_2 \\ A, \text{if } T_2 \leq f(x,y) \end{cases} \dots\dots\dots (3.4)$$

## 2、全局阈值法

全局阈值法，就是选择一个常数作为阈值并将其作用于整幅图像中的每一个像素。其中，比选定阈值大的像素标记为背景，反之为物体。这种方法处理速度快，尤其适用于背景单一的图像，但是当图像的背景复杂时，就不能很好地达到预设的目标了。

### (1)极小值点阈值

如果把图像所对应的灰度直方图看成是曲线，那么可以借助求取曲线的极小值来选取直方图的谷。设  $h(z)$  表示图像的灰度直方图，那么极小值点应该满足以下两个条件：

$$\frac{\partial h(z)}{\partial z} = 0 \quad \text{和} \quad \frac{\partial^2 h(z)}{\partial z^2} = 0 \quad \dots\dots\dots (3.5)$$

### (2)迭代阈值分割

取灰度中值作为初始值  $T_0$ ，借助下面的公式进行迭代计算：

$$T_{i+1} = \frac{1}{2} \left\{ \frac{\sum_{k=0}^{T_i} h_k \cdot k}{\sum_{k=0}^{T_i} h_k} + \frac{\sum_{k=T_i+1}^{L-1} h_k \cdot k}{\sum_{k=T_i+1}^{L-1} h_k} \right\} \dots\dots\dots (3.6)$$

在公式 3.6 中， $h_k$  的含义是指灰度取值为  $k$  的像素的数目， $k$  的取值范围为 0 到  $L$ 。当  $T_{i+1}=T_i$  时迭代结束，将此时的  $T_i$  值为阈值。

### (3)OTSU 法

Ostu 法是最常用的全局阈值法，该方法的主要思想是：首先利用阈值  $T$  将二值化处理后得到的车牌图像的灰度值分为两类，然后计算出相应的类间方差和类内方差，将两者相比并求出最大的比值，以此确定出阈值  $T^{[13]}$ 。

如果将二值化处理后得到的车牌图像的灰度值有  $L$  个取值范围，并且将此时相应的阈值假设为  $T$ ，那么灰度就可以分为两类，其中将小于  $T$  的灰度值设为

类 1，显然，大于  $T$  的灰度值设为类 2<sup>[14]</sup>。那么此时，可以利用相应的公式和已知值分别计算出均值  $M(T)$  和方差  $\sigma(T)$ 。在对车牌图像的灰度值进行分类之前，如果假设所有灰度值的平均取值为  $M_T$ ，那么此时的类内方差可以利用公式 3.7 计算求得。

$$\sigma_{\omega}^2 = \omega_1(T)\sigma_1^2(T) + \omega_2(T)\sigma_2^2(T) \dots \dots \dots (3.7)$$

类间方差由下式计算：

$$\begin{aligned} \sigma_b^2 &= \omega_1(T)(M_1(T) - M_T)^2 + \omega_2(T)(M_2(T) - M_T)^2 \dots \dots \dots (3.8) \\ &= \omega_1(T)\omega_2(T)(\omega_1(T) - \omega_2(T))^2 \end{aligned}$$

将公式 3.7 和公式 3.8 计算所得的值相比即  $\sigma_b^2 / \sigma_{\omega}^2$ ，要想在类内方差  $\sigma_{\omega}^2$  保持不变的情况下得到最大的比值，就必须尽量取到  $\sigma_b^2$  的最大值，此时对应的像素值就是希望得到的阈值  $T$ 。

### 3、局部阈值法

一般来说，在经过二值化处理后得到的车牌灰度图像中，图像的背景的像素值是不是固定不变的，当然图像中的物体的像素值和背景的像素值的比值也不是一成不变的<sup>[15]</sup>。正是因为这个原因，阈值  $T$  也会随之改变，这时就不能再采用全局阈值法来处理了，这时就得采用局部阈值法了。局部阈值法是对所有像素取一个阈值，并且保证所取阈值只与其周围的像素有关。

因为车牌图像中的所有像素都要确定一个阈值  $T$ ，所以最后的识别效果比较理想，适用范围广，但是执行时间长实时性不好。对图像进行局部二值化处理，有一个不可避免且难以克服的问题就是图像容易不连通<sup>[16]</sup>。

#### (1) Niblack

这是一个行之有效并且非常简便的方法，借助计算车牌图像中局部位置像素的均值和方差来确定中心位置的像素的阈值。例如，对一个中心位置为  $(i, j)$  的  $r \times r$  的局部图像而言，可以很轻松地计算出它的像素的均值和方差值： $m(i, j)$ ， $s(i, j)$ ，那么该像素的点的阈值为：

$$T(i, j) = m(i, j) + k \times s(i, j) \dots \dots \dots (3.9)$$

$r$  的大小非常敏感，若  $r$  太大，图像的细节容易被删去，反之，若  $r$  太小，图像会存在很多噪声。



## (2) Bernsen

对中心为  $(i, j)$  的  $r \times r$  的窗口, 其最大和最小像素值为  $Z_h, Z_l$ , 若  $Z_h - Z_l \geq l$ , 则其相应阈值为:

$$T(i, j) = (Z_h + Z_l) \div 2 \quad \dots \dots \dots (3.10)$$

否则该窗口属于同一类。

目前可使用的二值化算法有很多, 本文仅就其中几种做了简单介绍, 每种方法各有优缺点。当采用全局二值化方法来处理图像时, Ostu 法是最常用且效果还不错的一种方法。因此本次毕业设计, 就应用了 Ostu 法进行图像二值化处理, 这也为后续的车牌识别等过程提供了一个高质量的车牌图像。

## 3.1.3 数学形态学处理

在车牌识别系统中, 利用形态学的知识和相关算法可以排除一些直观上的干扰因素, 有利于准确定位车牌位置。它的原理是首先找到一个有一定形态的结构元素, 接下来利用它对车牌图像中的相关形状进行量度和提取, 从而实现对车牌图像的分析 and 识别的功能<sup>[17]</sup>。数学形态学主要采用了领域运算形式, 可以简化图像数据, 保留图像中的有用信息, 去除无关信息。最基本的形态学运算符有膨胀运算、腐蚀运算以及二者组合而成的开运算和闭运算等。

## (1) 膨胀运算

膨胀运算的基本原理是将与物体接触的部分背景点合并到物体中<sup>[18]</sup>。经过图像膨胀, 图像边界变大, 元素面积相应增加, 而且图像膨胀可以填充图像中的空隙和连接断续点, 使其成为连通区域。膨胀运算的公式如下:

$$D = B \oplus S = \{x, y | S_{xy} \cap B \neq \emptyset\} \quad \dots \dots \dots (3.11)$$

## (2) 腐蚀运算

腐蚀运算的基本原理是在结构元素的制约下, 删减物体的边界点和边界上的突出部分, 使其向内收缩。其主要应用于消除分割图像时产生的微小且无意义的点。腐蚀运算的公式如下:

$$E = B \otimes S = \{x, y | S_{xy} \subseteq B\} \quad \dots \dots \dots (3.12)$$

## (3) 开运算

开运算也就是对待处理图像进行腐蚀操作和膨胀操作的过程。它的作用是在

不改变物体面积的条件下，滤除图像中的细小孤立点、毛刺，擦除在两块区域的之间的连接小点，平滑图像的大致轮廓。同时在保证图像面积基本不变和相对完整地保留结构元素点的条件下，不改变图像总的位置和形状。

#### (4) 闭运算

闭运算即先膨胀后腐蚀。该运算可以在不影响图像中的明亮细节的条件下，去除暗细节部分。经闭运算后，图像中的细小空隙得以填充，连通图像中临近的物体，图像边界更为平滑，在不明显改变物体面积的情况下，图像总的位置和形状保持不变。

本论文中对字符区域进行形态学腐蚀运算，去除图像中不连续的边界点，再应用形态学闭运算，连通图像中的小块区域，最后删除干扰因子初步定位车牌区域<sup>[19]</sup>。确定车牌区域的流程图如下：

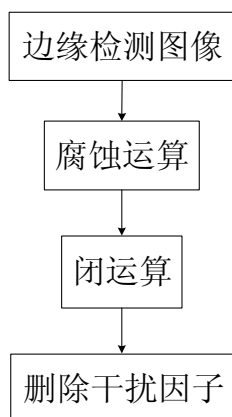


图 3.4 确定车牌区域的流程图

### 3.2 车牌定位算法简介

目前常用的车牌定位算法有数学形态学法、边缘检测法、小波分析法、灰度值法<sup>[20]</sup>。

#### (1) 基于数学形态学的定位算法

基于数学形态学的定位算法就是利用腐蚀运算和闭运算处理二值化图像，确定出目标，滤除噪声，实现车牌的定位。其中腐蚀运算可以删去图像中的边界点和边界上的突出部分，闭运算可以连通图像中的小块区域，最后依据车牌图像的特征知识来实现功能。这种方法可以自动调整阈值，从而适应光照等外界因素的干扰，并且可以快速定位。在本文中详细解释了数学形态学的算子。

#### (2) 基于边缘检测的定位算法

边缘是指图像中灰度级发生阶跃变化的部分，由于边界的灰度值和物体的灰

度值有差异，所以可以根据车牌内部字符的边缘特征，找出车牌的大致区域。这种方法有一定的局限性，就是对图像的质量要求较高，如果图像中的干扰太多，就有可能发生误判，将干扰强的边缘标记为车牌边界。本系统中采用的是 Canny 算子。

### （3）基于小波分析的定位算法

小波分析的定位算法结合了数学形态学的一部分知识，先用小波提取边缘边缘中的小块区域，在不同的分辨率层次上分割图像；再利用形态学运算滤除噪声，从而实现精确定位，这种方法计算量较大，定位速度较慢。

### （4）基于灰度值的定位算法

基于灰度值的定位算法采用了线段探针搜索定位车牌区域，其基本原理是利用车牌底色、边框颜色以及背景颜色在灰度化之后产生的灰度值有差异，形成了灰度值阶跃变化的边界。车牌的边框灰度值比背景灰度值高，并且边框为平行四边形，可以使用边缘检测算法，将车牌区域提取出来。

## 3.3 本文车牌定位算法

本文采用数学形态学方法实现车牌定位的效果，由于车牌定位的准确度受到选择的结构元素的制约，太大的结构元素使车牌的区域粘连成连通域，会带入一定的噪声；而太小的结构元素则不能将车牌字符区域粘连在一起，以至于无法准确定位出车牌区域，所以必须将数学形态学定位算法和其他算法结合起来。边缘检测可以去除大量的噪声，突出车牌的边缘部分，此时再采用数学形态学算法处理图像，就可以很好地将车牌区域粘连成连通域，再根据车牌的固有特征，实现功能<sup>[21]</sup>。该方法的具体思路如下图所示：

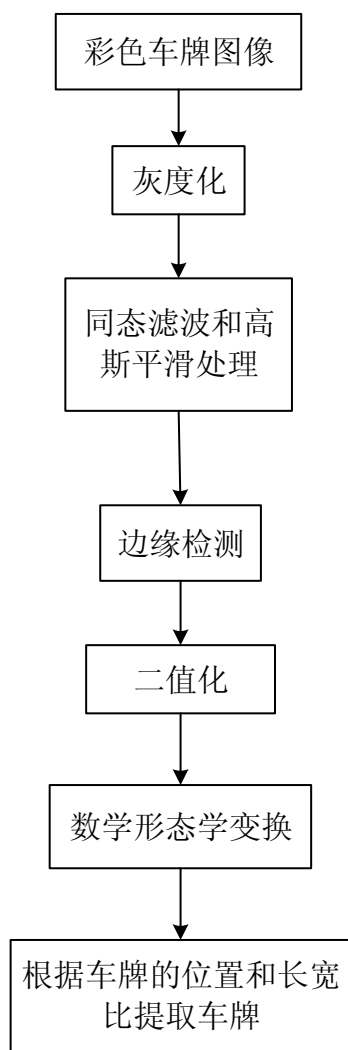


图 3.5 车牌定位算法流程图

### 3.4 车牌倾斜矫正

通常情况下，车牌的形状是一个比例固定的矩形，但是由于拍摄角度和车辆位置的不同，实际拍摄出的车牌往往会产生畸变，这对接下来的字符分割和字符识别产生了一定的影响。通常车牌倾斜了之后，会产生下图中出现的三种情况。

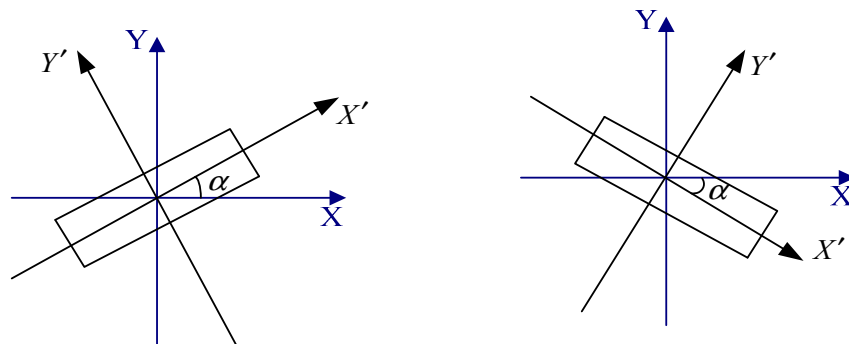


图 3.6 水平倾斜的两种情况

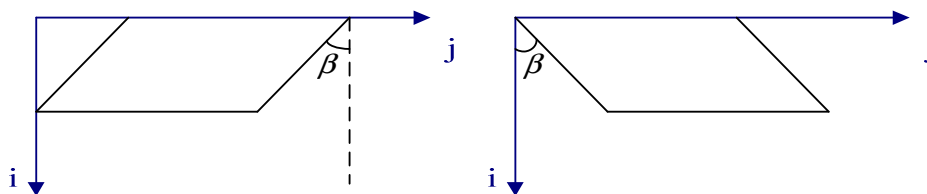


图 3.7 垂直倾斜的两种情况

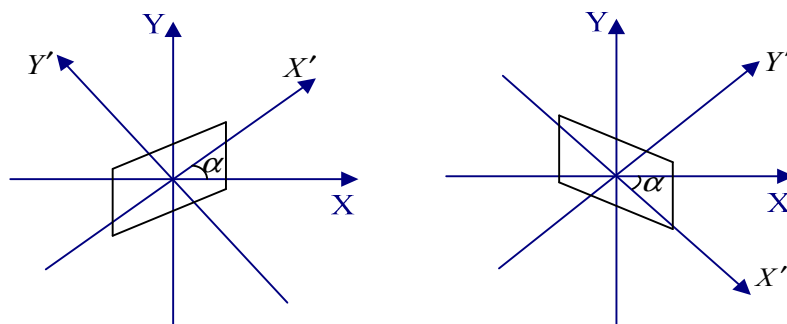


图 3.8 水平垂直倾斜的两种情况

图 3.6 所示就是车牌图像发生水平倾斜时出现的两种位置，此时车牌上的字符几乎没有发生倾斜，车牌的坐标轴与图像的坐标轴间相差的倾斜角度为  $\alpha$ ，求出的值为  $\alpha$ ，主要将图像旋转  $-\alpha$  度就可以矫正好车牌。车牌垂直倾斜的两种情况如图 3.7 所示，此时的倾斜事实上是同一行上的像素的偏移，求出倾斜角度  $\beta$  然后进行矫正。车牌水平垂直倾斜的两种情况如图 3.8 所示，此时车牌图像即发生了水平倾斜又发生了垂直倾斜。这种倾斜情况是最常见的。

在我国研究车牌识别技术时，使用率较高的矫正算法有：Hough 变换法、旋转投影法和 Radon 变换法<sup>[22]</sup>。Hough 变换法是利用车牌的上下边沿与水平夹角的倾斜角度，来实现车牌位置的矫正的。但是在现实情况下，受到噪声和车牌边框不够突出等干扰因素的影响，Hough 变换法应用起来困难重重，所以计算量大而且矫正效果不够理想。为此，应该在矫正车牌前，先对车牌进行二值化和边缘检测处理，使得车牌轮廓分明，以便于接下来的矫正处理。旋转投影法是利用反复旋转车牌图像的方法，将旋转后的图像投影到坐标轴上，通过寻找旋转后的投影最大值，试探性地求出倾斜角度从而实现车牌位置矫正。尽管旋转投影法可以较好地避免干扰，但是实时性不够高，所以本文并未采用这种方法。Radon 变换法是这三种方法中相对较好的一种，这种算法也需要先对二值图像进行边缘检



测,然后再对车牌做形态学处理,接着计算水平边的倾斜角 $\theta$ ,最后利用如下所示的公式进行 Radon 变换。Radon 变换不仅检测了水平倾斜角还检测了垂直倾斜角,然后将车牌旋转某一给定角度从而矫正车牌水平位置,并使用双线插值的方法来实现车牌垂直方向的矫正。

$$\begin{cases} x' = x \cos \theta + y \sin \theta \\ y' = -x \sin \theta + y \cos \theta \end{cases} \dots\dots\dots (3.13)$$

本文采用 Hough 变换法,通过检测上下两条边框来实现矫正车牌位置的效果,图 3.9 就是倾斜车牌位置矫正的算法流程图。

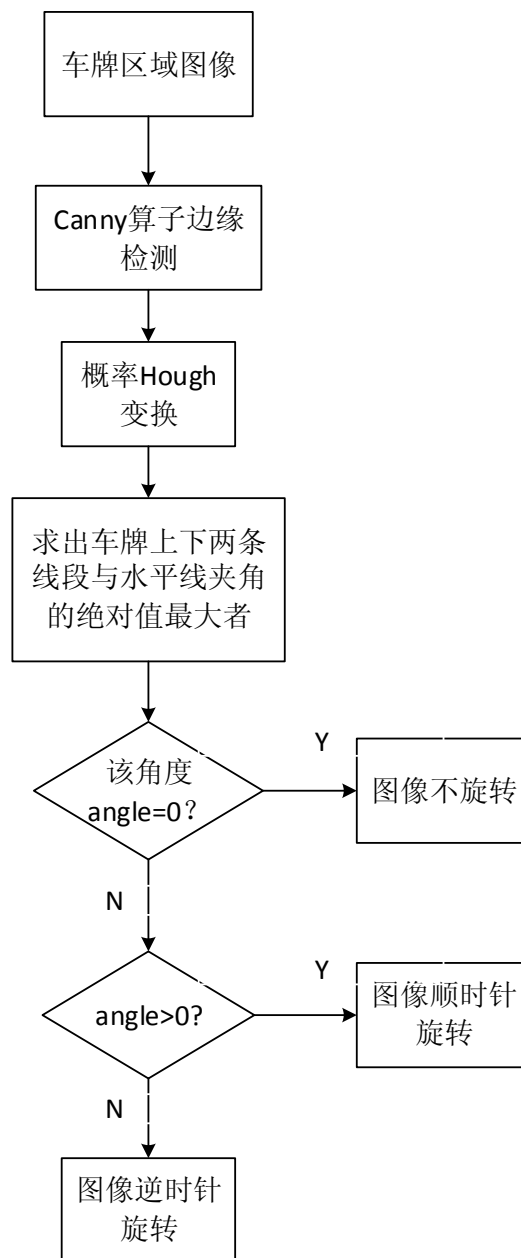


图 3.9 车牌矫正算法流程图

### 3.5 本章小结

本章主要分析了车牌的定位方法,首先涂药对采集好的车牌图像进行图像预处理,在本文中主要利用了图像的灰度化,图像的二值化以及数学形态学方面的方法为之后的图像分割以及图像识别打好了基础。分析了两种图像定位的方法,分别是基于边缘和基于颜色的定位方法并进行了倾斜校正。

## 第四章 车牌分割算法研究

### 4.1 车牌字符的规格

本系统设计的识别软件主要是面向低排量汽车。这种车牌的边框尺寸为 450×150mm，其中包含了 7 个字符。车牌第一个字符是汉字，代表的是地区，在车牌中第二个位置上的字符是大写的英文字母，代表的含义是发证机关，在车牌中第三个位置上的字符到第五个位置上的字符有可能是大写的英文字母，也可能是阿拉伯数字，在车牌中第六个位置上的字符和第七个位置上的字符是阿拉伯数字。根据国家规定，所有车辆的汽车牌照上的字符必须达到 409mm 的长度，而每个独立字符必须达到 45mm 的宽度和 90mm 的高度，第二个位置上的字符与第三个位置上的字符间相距 34mm，而其他字符间相隔 12mm 的距离。所以在进行车牌识别技术的研究时，可以利用这些先验知识来达到车牌字符分割的目的。

### 4.2 字符分割方案分析

车牌的字符分割就是指将车牌号码的 7 个字符从定位出的车牌图像中一个一个地分割并提取出来。字符分割的依据是字符边缘、字符的高宽比、字符间间距等特征信息。这里简要介绍投影法和模板匹配法。投影法包括水平投影法和垂直投影法，通常情况下，使用垂直投影法来进行车牌字符分割。垂直投影法就是将字符在垂直方向进行投影，然后利用在该方向上的投影，字符间和字符内的间隙能取得局部最小值的方法，切割局部最小值附近，就可以实现字符分割。

现在有很多进行图像阈值分割的方法，基于纹理分析的分割方法就是其中的一种。该方法可以根据带测字符的灰度梯度或分布与背景的灰度梯度或分布进行分割，利用基于模式识别的方式种的最大值最小值准则进行图像的函数求解，进而获得最终的分割结果，当字符的灰度大于一定阈值  $a$  时：

$$\sum_{i=0}^a P(x/\text{背景}) = \sum_{i=a}^{\infty} p(x/\text{字符}) \quad \dots\dots\dots (4.1)$$

当字符的灰度小于一定阈值  $a$  时：

$$\sum_{i=0}^a P(x/\text{字符}) = \sum_{i=a}^{\infty} p(x/\text{背景}) \quad \dots\dots\dots (4.2)$$

其中： $p(x/\text{字符})$  代表车牌字符的灰度的分布函数， $p(x/\text{背景})$  代表

车牌背景的灰度的分布函数。水平投影的方法就是对图像从上至下进行逐行扫描，当扫描到的像素点开始出现白色时，记录此时的位置，然后再从下往上扫描，同样地，当遇到第一个白点时，记录位置，从而基本可以确定出图像的高度范围。用同样的方法，左右方向扫描可以确定出图像的宽度范围。这种方法程序简单，计算量小，执行速度快，但是不能很好地解决字符粘连等问题。模板匹配法是根据车牌字符的排列规则，利用已有模板与竖直积分投影曲线进行匹配，计算出匹配位置的最优值，从而实现车牌分割的功能。

### 4.3 边框的去除

在实际操作时，由于客观因素的干预，待识别的车牌图像中字符背景往往比较复杂，包含了很多的干扰信息。当车牌经过二值化之后，仍然会保留一些噪声如边框和铆钉等。在进行车牌字符分割之前需要将车牌上的干扰信息去除，否则势必会影响接下来的字符分割和识别过程。

车牌图像经过二值化后，可以明显地看出车牌边框与车牌字符区域的明暗跳变次数的差异，车牌字符区域黑白跳变频繁，可以利用这个特征来去除车牌的边框。显而易见地，可以通过观察发现字符区域和边框中存在间隙，所以可以搜索并删去间隙，保存字符区域。可以对矫正后的车牌进行行扫描，从车牌的上边框水平向下扫描，当扫描到车牌的  $1/4$  处时，发现车牌的黑白跳变次数大于某一设定阈值时，停止扫描，并删去这  $1/4$  的区域，此区域是车牌上边框。对于左右边框和下边框也可以采用同样的方法来去除。去除上边框的算法流程图如 4.1 所示：

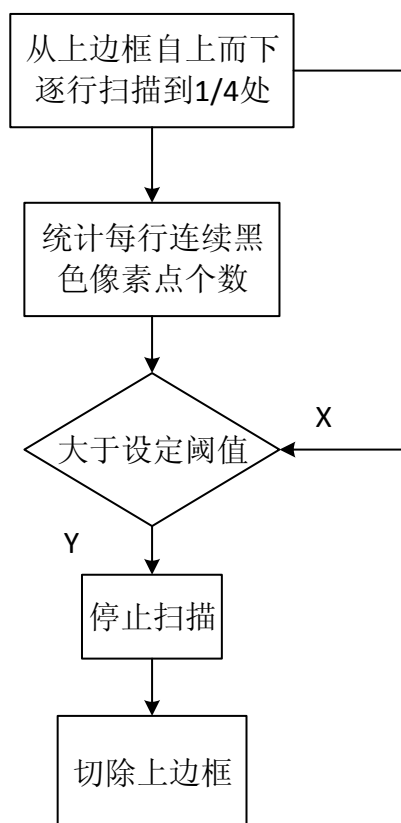


图 4.1 去除上边框算法流程图

去除边框之后的图样如下图 4.2 所示：



图 4.2 去除边框的牌照

#### 4.4 单个字符的分割

经过前面所做的工作已经将原来采集到的图像进行预处理，图像增强，字符定位，并分割，现在需要将每一个单个的字符分割出来，以备后续的字符识别。单个字符的分割主要需要进行整个字符轮廓的分析和分割，字符投影后的效果和字符与字符之间的间距的问题，需要综合考虑。

一般来说车牌的字符中拥有 7 个字符，这 7 个字符包括阿拉伯数字，英文字符和省市的缩写汉字。根据图像的大小，设置一个阈值，检测图像的 X 轴方向，若宽带等于这一阈值，则进行分割，分离出 7 个字符。

$[m,n]=\text{size}(d)$ ,逐排检查有没有白色像素点, 设置 $1 \leq j \leq n-1$ ,  
若图像两边 $s(j)=0$ ,则切割, 去除图像两边多余的部分

切割去图像上下多余的部分

根据图像的大小, 设置一阈值, 检测图像的X轴,  
若宽带等于这一阈值则切割, 分离出七个字符

归一化切割出来的字符图像的大小为 $40 \times 20$ ,  
与模板中字符图像的大小相匹配

图 4.3 字符分割框图

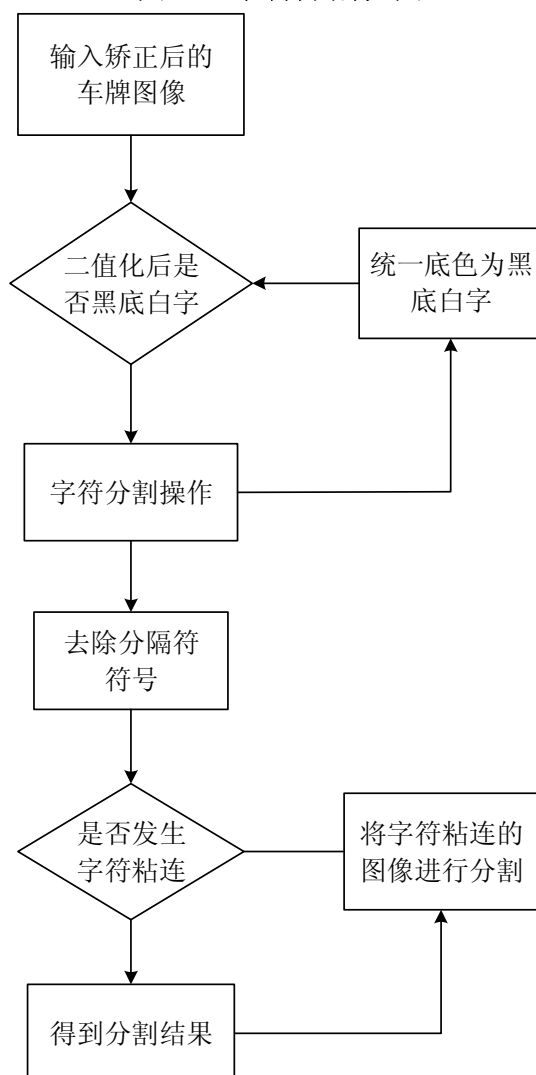


图 4.4 算法流程图

匹配程度由匹配偏移量  $M_T$  来决定，匹配偏移量的计算公式如下：

$$M_T(\omega, a_1) = \sum_{i=1}^6 \sum_{n=b_i}^{a_{i+1}} p(n) \quad \dots \dots \dots (4.3)$$

其中  $p(n)$  为竖直积分投影曲线， $\omega$ ,  $b_i$ ,  $a_{i+1}$  如图 3.7 所示。随着偏移量  $M_T$  的值不断变小，匹配的程度会变得越来越高的。

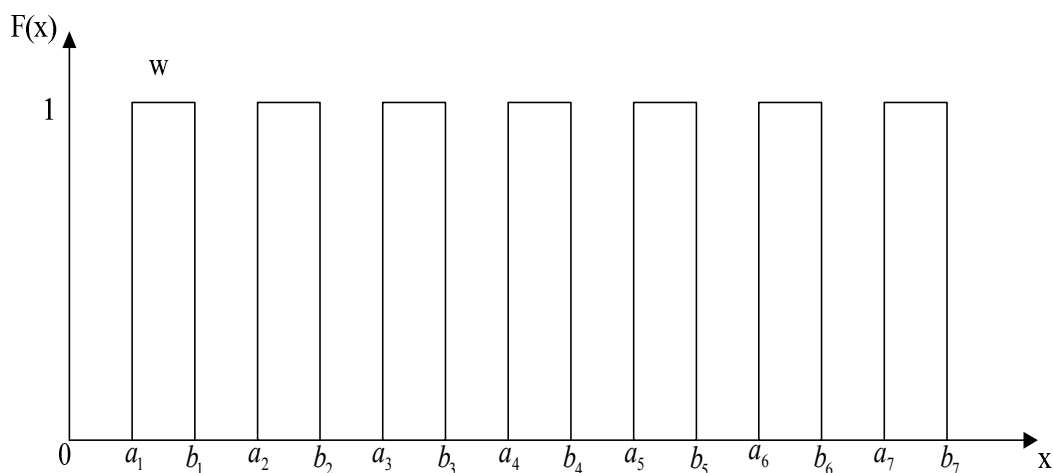


图 4.5 字符分割模板

#### 4.5 字符归一化

当然从车牌中进行分割下来的字符都是差不多大小的，但是一般在分割过程中很容易受到外界因素的干扰使其在分割过程中造成过分割或者缺分割，这就需要在设计过程中进行字符的补齐，使分割下来的字符在长和宽上相同，方便下一步的字符识别，这就是字符的归一化过程。

如果在切割的过程中，出现了过切割的时候，一般使用递推差值的方式进行补齐，本文应用的差值算法如下，其插值公式如式 4-12 所示：

$$f'(x, 2i) = f(x, 2i+1) = f(x, i) \quad i = 0, 1, 2, \dots, 16 \quad \dots \dots \dots (4.4)$$

$$f'(3j, y) = f'(3j+1, y) = f'(3j+2, y) = f(j, y) \quad j = 0, 1, 2, \dots, 8 \quad \dots \dots \dots (4.5)$$

其中， $f(x, y)$  是原始的车牌图像， $f'(x, y)$  是归一化之后的车牌图像。

#### 4.6 本章小结

本章主要介绍了车牌识别系统中的图像分割方面的内容，首先分析了现有中国的车牌的主要规格，然后分别进行了图像的二值化和边框的去除以及字符的归一化等内容。

## 第五章 车牌字符算法研究

### 5.1 引言

车牌的字符识别是整个车牌识别系统的最终环节，也是最为重要的一环，它是将前面字符分割后形成的字符图像识别出真正的文本字符，并给予输出显示。车牌的信息中主要包括了阿拉伯数字，汉字和大写的英文字符。车牌识别的思想主要是模式匹配的思想。首先需要建立一个大量的数据库，将库中元素的模式特征抽取出来，并制定出一定的准则，然后将采集来的图像进行分割后，将其与数据库中的元素进行比对，从而识别出车牌中的字符。阿拉伯数字和英文字符识别较为容易，因为其比划较少，而汉字的比划较多，识别难度较大。主要原因如下：

- 由于车牌受到一定程度的污染或者喷上泥泞的东西，容易造成误识别。
- 由于有些汉字的比划较为复杂，出现粘连等情况，使其识别难度较大。

### 5.2 车牌字符识别常用方法

#### 5.2.1 基于模板匹配的字符识别算法

模板匹配的方法是字符识别中从原理来讲最简单的一个。它的主要做法是将已经采集好并分割出来的待识别字符与已建立的数据库中的元素进行比较匹配，经过逐一比较之后相似度高的作为最终的识别结果。而在建立数据库和进行匹配的过程中，相似度函数是一个最为关键的函数，它是能否正确识别的关键。车牌字符的宽高比，字符间的间隔都需要作为相似度函数的特征进行考虑。相似度函数是表征待测字符与数据库中相似程度的量，将待测字符与数据库中的字符做差，差值越小，说明相似度越好，图 5.1 描述了整个利用模板匹配的方法进行字符识别的整个过程。将之前采集好的车牌图像，首先进行图像预处理，包括图像的灰度化和二值化，然后进行滤波和图像的增强等，最终将其图像分割与其在数据库中的数据进行匹配。可以利用欧式距离的差值函数表示模板匹配的的程度如下公式所示：

$$D = \sqrt{\sum_{x=1}^{col} \sum_{y=1}^{row} (f(i, j) - g(i, j))^2} \dots\dots\dots (5.1)$$

其中  $f(i, j)$  和  $g(i, j)$  分别代表待测字符和数据库中的字符特征函数，当  $D$  较大时，说明匹配效果不是很好。当  $D$  达到最小时，说明匹配度达到最好，可以输出作为最终识别结果。由于灰度图像拥有 256 个灰度级别，所以这种方法得计



算量相对比较大。

为了减小计算量，采取二值化后的图像进行整个车牌的识别过程，此时的匹配度公式如下式所示：

$$D = \sqrt{\sum_{x=1}^{col} \sum_{y=1}^{row} f(i, j) \oplus g(i, j)} \dots \dots \dots (5.2)$$

虽然这种方法有效的缩短了计算的时间，但相对而言，图像的信息量也相对的减少了，所以整个识别效果并不能达到很好。总体来说如果对于图像清晰，车牌中的字符不复杂的话，此种方法具有识别快的优点，但对于相对复杂的车牌，识别度较差。

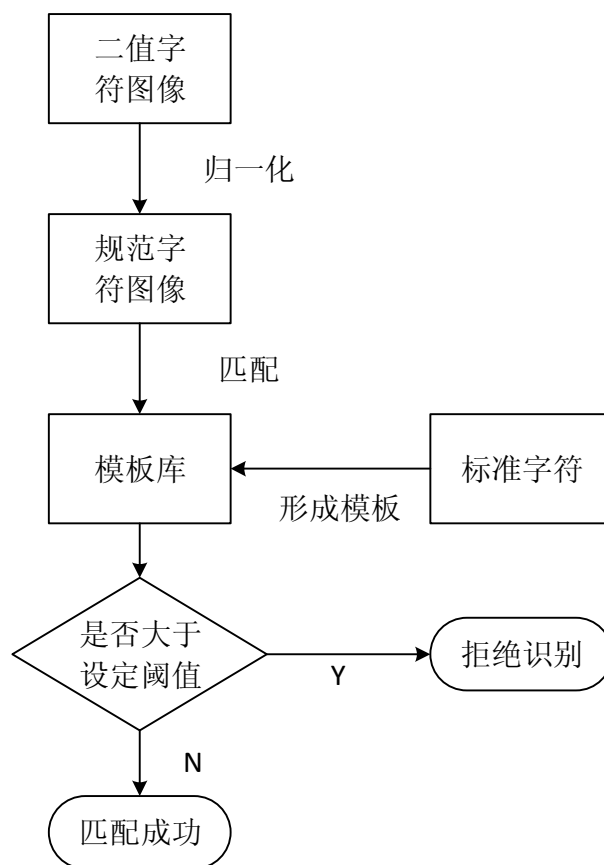


图 5.1 模板匹配法的算法流程图

### 5.2.2 基于支持向量机的字符识别

支持向量机建成 VSM，是一种机器学习方法，它可以通过标签上的数据使用进行学习，可以用来监督学习。当需要一些具有标签性质的东西需要学习的时候，可以训练以往的数据来对标签上的某个性质进行增强。训练 SVM 分类器，简单来说在二维空间里就是求解这样一个线性函数， $g(x)=wx+b$ ， $x$  是输入的特

征， $g(x)$  是输出的分类的类别，已知了输入特征与输出类别，来训练求解  $w$  和  $b$  的最优的值，获得这两个值后也就获得了这个分类的数学模型，接着就可对新样本进行预测。在任一分类之前的第一个任务是训练分类器。整个识别过程需要经过图像预处理然后提取出图像的具体特征，然后对图像中的特征进行学习，形成特征参数存储起来，然后对一幅待测的车牌图像进行决策计算、判决和最终显示识别结果。

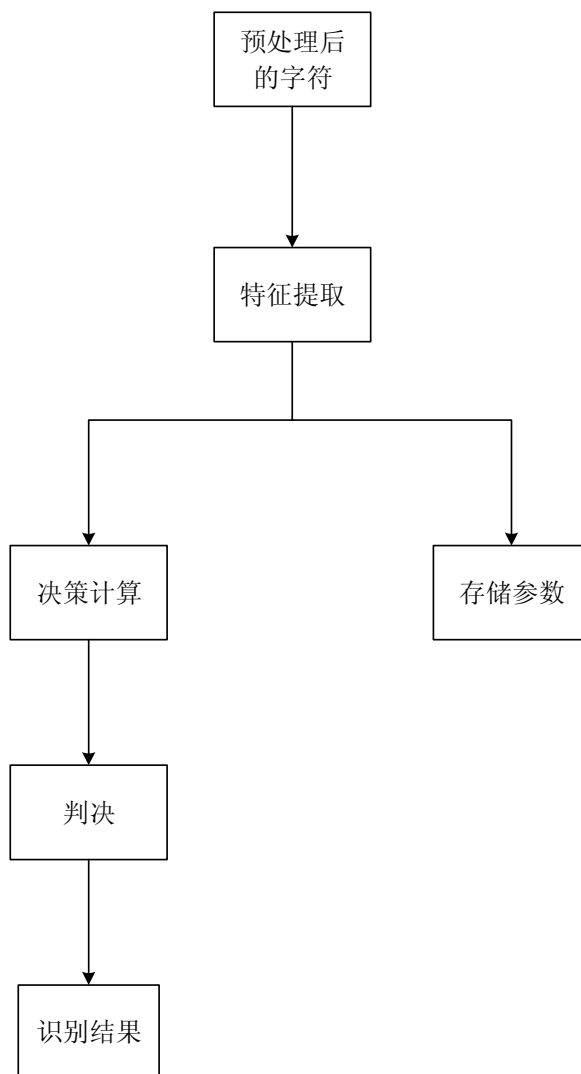


图 5.2 字符训练与识别结果

### 5.2.3 基于人工神经网络的字符识别算法

人工神经网络是现在很常用的一种用于图像识别的方法。本文正是采用的利用神经网络学习的方法去识别车牌的。人工神经网络 (Artificial Neural Networks) 的本质就是用机器语言来试图模拟人类做出判断和决策。神经网络的字符识别算

法可以分为两个过程：学习期和工作期。

具体来说就是将输入设置为字符图片，输出设置为对应的文本字符，反复训练，通过神经网络的学习，最后得到一个网络；然后，进入工作期，将神经网络的输入设置为待识别字符图片，利用网络连接权值不变，得出输出量即正确的文本字符。

人工神经网络算法适应性强，对图像的质量和车牌的位置要求不高，并且拥有执行速度快和并行处理能力强的优势，可以精确识别出车牌字符。唯一的不足之处是，它在学习期需要大量的训练数据。人工神经网络识别过程示意图如图 5.3 所示。在神经网络的四类网络模型（前馈网络、反馈网络、自组织网络和随机型网络）中，反馈神经网络即 BP（Back Propagation）神经网络在实际生活中被大量应用。该网络具有三个优点：（1）非线性映射能力（2）泛化能力（3）容错能力。基于这些优点，本系统将要采用 BP 神经网络进行车牌字符识别。

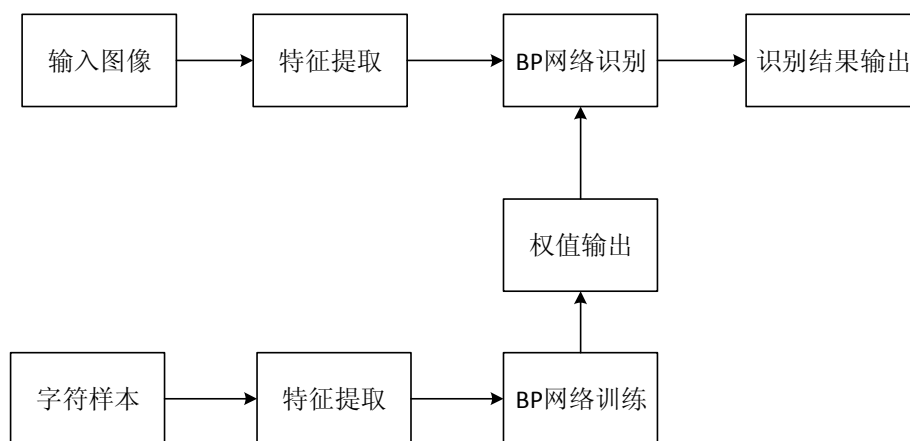


图 5.3 人工神经网络识别过程

### 5.3 BP 神经网络的结构

采用 BP 算法的人工神经网络就是 BP 神经网络，BP 网络也可以称为反向误差传播神经网络，它是一个多层的前向神经网络。BP 神经网络由学习与训练两个步骤完成，其中信号的正向发送与误差的反向发送是学习过程的两个主要阶段<sup>[28]</sup>，训练过程则是一直进行到误差减小到一个要求的范围为止。BP 神经网络的网络结构拓扑图如图 5.4 所示。从图 5.4 可以直观地看出，BP 神经网络是一个多层感知器，可以分为输入层（input layer）、隐含层（hide layer）和输出层（output

layer), 每一层内部的神经元没有连接起来, 而层与层之间的神经元是采用全连接的方式进行连接的。每一层的神经元个数和隐藏层的个数根据问题本身的不同情况, 各有不同。

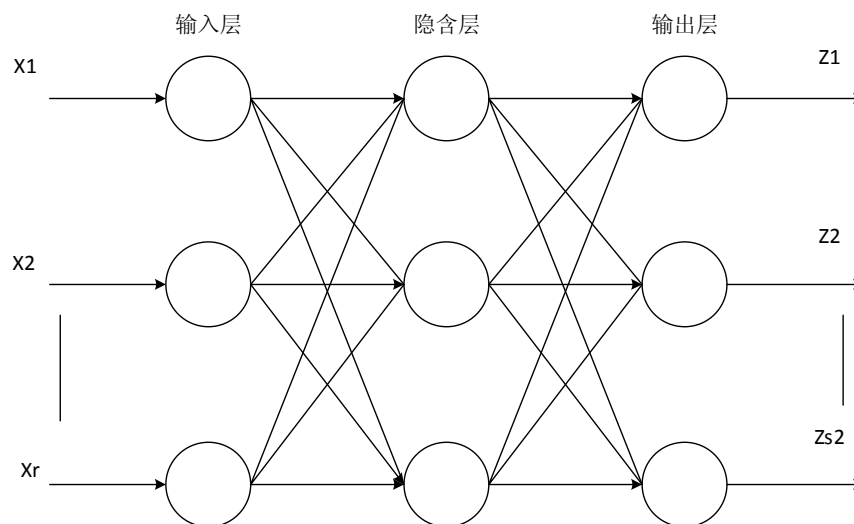


图 5.4 BP 神经网络结构图

#### 5.4 BP 神经网络在车牌识别中的应用

首先要提取出所得字符的特征向量, 然后将其作为输入送入 BP 神经网络, 对其进行反复的训练, 得出相应的权值矩阵, 即达到了将字符分类的效果, 最后数字、汉字的输出即为识别最终结果。图 5.5 就是借助了 BP 神经网络算法进行的字符识别过程, 可以从中看出大体的流程。

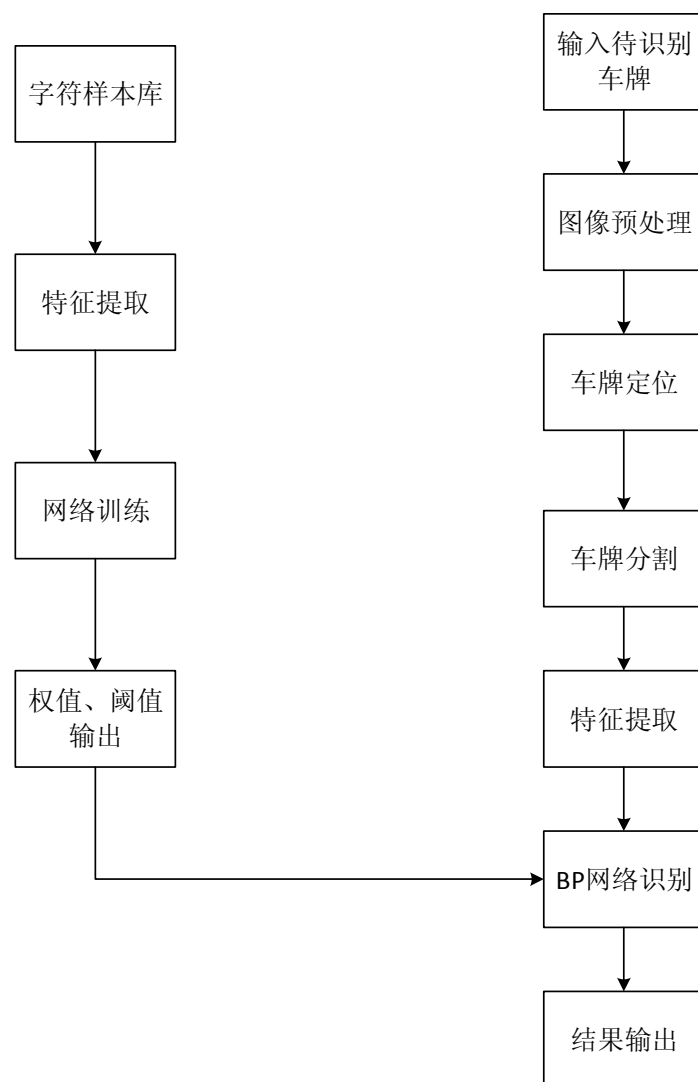


图 5.5 基于 BP 神经网络的车牌字符识别示意图

#### 5.4.1 特征提取

针对不同类型的字符，可以采用不同的特征提取法来提取出字符的特征，如汉字识别器、字母识别器和数字识别器等。由于篇幅的限制，本文仅对字母识别做一定的介绍，并将对 20 点特征点提取法做简要介绍。

首先将字符等分成 16 份，统计出每份白色像素点的比例为 16 个特征向量，示意图如图 5.6 所示。

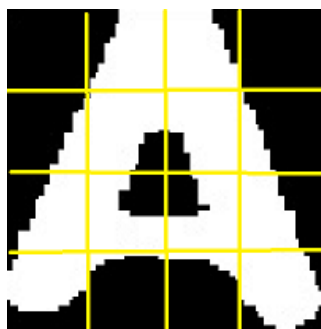


图 5.6 特征提取法

接着统计水平方向中间两列和竖直方向中间两列的白色像素点的比例作为后 4 个特征，最后统计所得的白色像素点比例的个数为第 20 个特征。如图 5.7 所示：

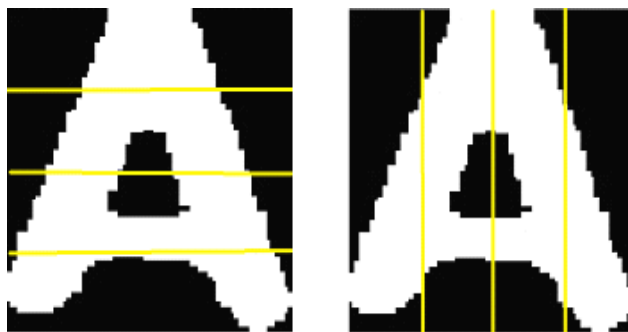


图 5.7 特征示意图

### 5.4.2 BP 神经网络的结构与设计

待识别字符的特征向量的维数对输入节点的神经元个数起到了决定的作用。本实验设计了 4 个三层 BP 神经网络，且每个神经网络都有一个隐含层，能够识别阿拉伯数字、汉字以及大写的英文字母。识别车牌字符中的汉字的 BP 神经网络包含了 20 个输入层和 2 个输出层（“桂”、“苏”，因为只找到这两种样本图片）。用于识别大写英文字母的 BP 神经网络包含 20 个输入层和 24 个输出层（除了 i 和 o）。识别车牌字符中的阿拉伯数字的 BP 神经网络包含有 20 个输入层和 10 个输出层。

### 5.4.3 网络训练

结构和算法设计完之后，就要进行网络训练。经过大量的样本训练就可以得到一个完备的字符识别系统了。经过 MATLAB 仿真之后，结果如下，图 5.8 为对 BP 神经网络的字符样本进行训练的过程示意图，图 5.9 为仿真之后误差的收敛曲线的示意图。

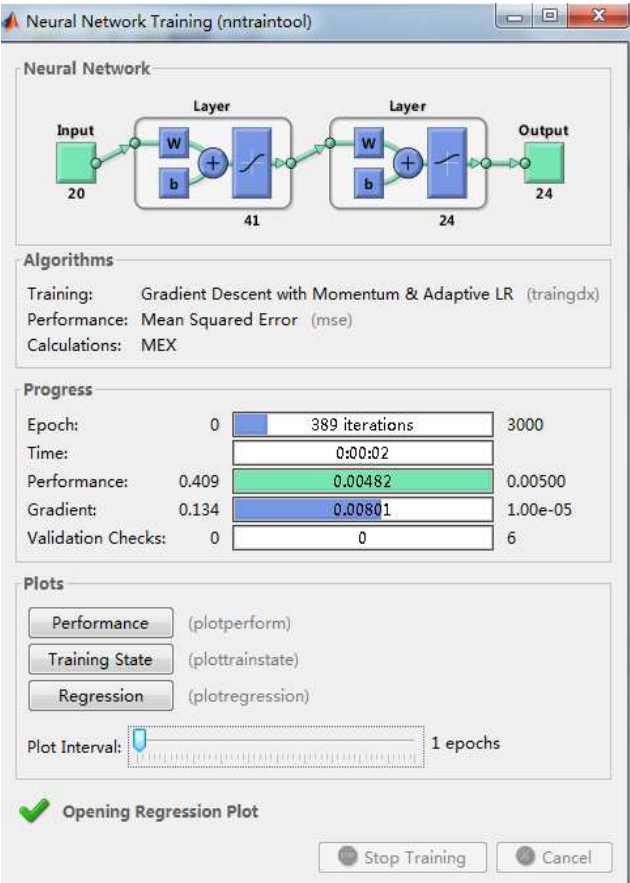


图 5.8 字符样本的训练过程

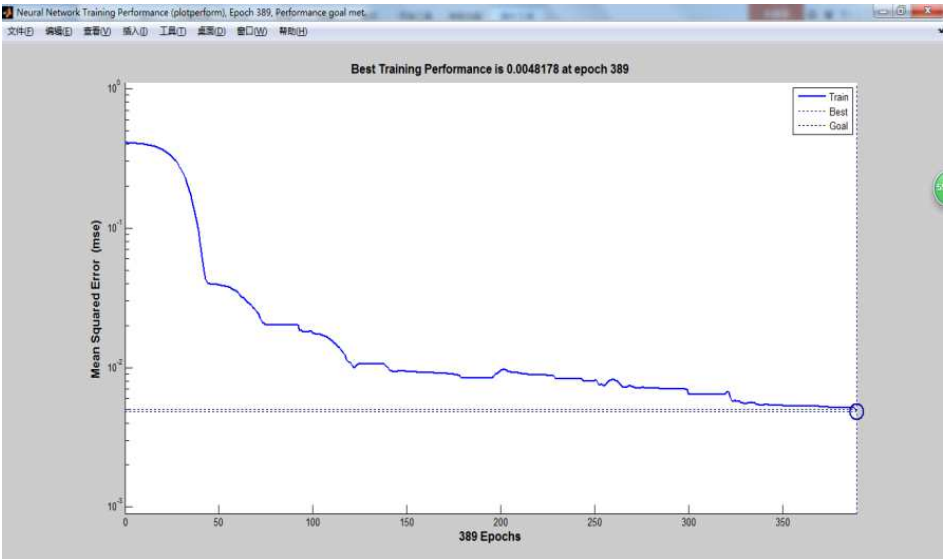


图 5.9 误差收敛情况

### 5.5 实验结果

为了测试 BP 神经网络车牌识别的速率和正确率，随机选取了 60 张车牌图

像进行识别，正确率可以达到 95%，有一定的误识情况，主要是（1）由于车牌图像被污染，（2）图像中含有较高的噪声。具体识别情况汇总如下：

表 5-1 车牌识别情况汇总

| 待识别车牌    | 识别结果     | 错误个数 | 识别时间 | 识别率   |
|----------|----------|------|------|-------|
| 吉 AJ103Q | 吉 AJ103Q | 0    | 0.37 | 100%  |
| 吉 AC3692 | 吉 AC3692 | 0    | 0.3  | 100%  |
| 吉 AM5885 | 吉 AM5881 | 1    | 0.36 | 85.7% |
| 吉 A75299 | 吉 A75299 | 0    | 0.57 | 100%  |
| 吉 AF2830 | 吉 AF2830 | 0    | 0.46 | 100%  |
| 吉 AF2830 | 吉 AF2830 | 0    | 0.53 | 100%  |
| 吉 C92030 | 吉 C72030 | 1    | 0.33 | 85.7% |
| 吉 AV8730 | 吉 AV8730 | 0    | 0.5  | 100%  |
| 吉 AE0886 | 吉 AE0886 | 0    | 0.42 | 100%  |
| 吉 AE6095 | 吉 AE6095 | 0    | 0.39 | 100%  |
| 吉 EFB319 | 吉 EFB319 | 0    | 0.36 | 100%  |
| 吉 EFB318 | 吉 EFB318 | 0    | 0.37 | 100%  |
| 吉 EFB349 | 吉 EFB319 | 1    | 0.43 | 85.7% |
| 吉 A3B099 | 吉 A3B099 | 0    | 0.42 | 100%  |
| 吉 AB6823 | 吉 AY6823 | 1    | 0.35 | 85.7% |
| 吉 AB6823 | 吉 AY6823 | 1    | 0.43 | 85.7% |
| 吉 AMK717 | 吉 AMK717 | 0    | 0.43 | 100%  |
| 吉 ADZ249 | 吉 ADZ219 | 1    | 0.39 | 85.7% |
| 吉 AQ8287 | 吉 AQ8287 | 0    | 0.41 | 100%  |
| 吉 AEE077 | 吉 AEE077 | 0    | 0.29 | 100%  |
| 吉 AMV889 | 吉 AMV889 | 0    | 0.42 | 100%  |
| 吉 AMF347 | 吉 AMF317 | 1    | 0.46 | 85.7% |
| 吉 A78Q22 | 吉 A78Q22 | 0    | 0.45 | 100%  |
| 吉 B85672 | 吉 BB5672 | 1    | 0.43 | 85.7% |



|          |          |   |      |       |
|----------|----------|---|------|-------|
| 吉 A78Q22 | 吉 A78Q22 | 0 | 0.37 | 100%  |
| 吉 AJJ678 | 吉 AGJ678 | 1 | 0.37 | 85.7% |
| 吉 AKM530 | 吉 AAM530 | 1 | 0.43 | 85.7% |
| 吉 AEG802 | 吉 AEG802 | 0 | 0.44 | 100%  |
| 吉 ACC199 | 吉 ACC199 | 0 | 0.4  | 100%  |
| 吉 ACC199 | 吉 ACC199 | 0 | 0.39 | 100%  |
| 吉 AM9678 | 吉 AM9678 | 0 | 0.43 | 100%  |
| 吉 F02235 | 吉 F02235 | 0 | 0.58 | 100%  |
| 吉 A72668 | 吉 A72668 | 0 | 0.4  | 100%  |
| 吉 A87360 | 吉 A87360 | 0 | 0.48 | 100%  |
| 吉 AM5881 | 吉 AM5881 | 0 | 0.36 | 100%  |
| 吉 AQ7956 | 吉 AQ7956 | 0 | 0.44 | 100%  |
| 吉 A33Z35 | 吉 A63Z39 | 1 | 0.39 | 85.7% |
| 吉 A05J23 | 吉 A05J23 | 0 | 0.4  | 100%  |
| 吉 AV5109 | 吉 AV5109 | 0 | 0.42 | 100%  |
| 吉 AB9388 | 吉 AB9388 | 0 | 0.7  | 100%  |
| 吉 ACF410 | 吉 ACF410 | 0 | 0.35 | 100%  |
| 吉 AD3966 | 吉 ALA966 | 2 | 0.5  | 71.4% |
| 吉 A05J23 | 吉 A05J23 | 0 | 0.39 | 100%  |
| 吉 AEE077 | 吉 AEE077 | 0 | 0.38 | 100%  |
| 吉 AK9097 | 吉 AW9097 | 1 | 0.35 | 85.7% |
| 吉 ANG577 | 吉 ANG577 | 0 | 0.39 | 100%  |
| 吉 AKM530 | 吉 AAM530 | 1 | 0.36 | 100%  |
| 吉 AX8665 | 吉 AX8665 | 0 | 0.36 | 85.7% |
| 吉 AUZ266 | 吉 AUT263 | 2 | 0.45 | 71.4% |
| 吉 AHD806 | 吉 AHD806 | 0 | 0.37 | 100%  |
| 吉 K18140 | 吉 K18140 | 0 | 0.29 | 100%  |
| 吉 AUG869 | 吉 ALG869 | 1 | 0.31 | 85.7% |
| 辽 ANQ238 | 辽 ANQ238 | 0 | 0.38 | 100%  |

|          |          |   |      |       |
|----------|----------|---|------|-------|
| 辽 AJ2868 | 辽 AJ2868 | 0 | 0.42 | 100%  |
| 辽 ASG928 | 辽 ASG928 | 0 | 0.36 | 100%  |
| 辽 A2C605 | 辽 ATL605 | 2 | 0.33 | 71.4% |
| 辽 A1V698 | 辽 A1W698 | 1 | 0.35 | 85.7% |
| 辽 AZF605 | 辽 ATL605 | 2 | 0.35 | 71.4% |
| 辽 A8N528 | 辽 A8N528 | 0 | 0.34 | 100%  |
| 辽 AMK717 | 辽 AMK717 | 0 | 0.31 | 100%  |



(a) 第一张车牌识别结果



(b) 第二张车牌识别结果

图 5.10 车牌识别示例

## 第六章 总结与展望

### 6.1 总结

本文主要研究了基于数字图像处理平台的车牌识别系统,对每一个功能模块都进行了深入的研究,包括车牌定位,图像预处理,车牌字符分割和车牌字符识别。首先分析了整个系统的研究意义和背景,对国内外研究现状进行了了解,对系统的硬件和软件系统进行了分类。

本文通过对车牌识别系统中图像预处理、车牌定位、车牌字符分割和字符识别四个关键环节进行研究。搭建了一套从图像采集到图像识别的完整车牌识别系统,并利用 MATLAB GUI 界面完成整个车牌识别的流程和显示。

在图像预处理方面,本文首先将彩色图像转化为黑白图像,然后分别进行了拉伸、均匀化、中值滤波等方面的处理,并进行了二值化。

在车牌定位方面,本文研究了前人根据颜色特征以及纹理特征的车牌定位的方法基础之上,结合两种方法的优势,提高了定位的精度,并有效的降低了定位所需要的时间。车牌定位中使用的算法是边缘检测算法中的 Canny 算子和形态学方法处理图像。车牌定位后图像中还存在着一定的噪声和干扰信息(如车牌边框和铆钉等等),所以需要使用一定的算法对车牌区域的图像进行处理。当图像中的车牌位置发生倾斜时,需要对图像进行矫正处理,文中具体的介绍了 Hough 变换法检测边框来实现矫正车牌位置。

在字符分割方面,仍然在充分分析了前人所提算法的基础上,分别添加了标准归一化,倾斜几何校正和去除边框等方法,很好的进行了字符的分割。

在车牌字符识别方面,在深入研究了几种常用的字符识别的方法基础上,字符识别部分是车牌识别系统中最为核心的环节,本文介绍了三种识别算法。模板匹配法、支持向量机 SVM 以及神经网络算法,其中本文使用了 BP 神经网络算法。这个算法的识别效果还是不错的,但是对一些相似度高的字符识别起来有定的难度。详细介绍了本人使用的基于 BP 神经网络的字符识别方法,并随机选取了 60 张车牌照片进行识别,识别正确率达到 90%。

### 6.2 展望及下一步工作计划

由于时间和本人能力有限,对本课题的研究还不够深入有以下一些问题待以后进行研究。

- (1) 由于在实际应用过程中, 车牌容易受到污染等侵蚀, 可利用模糊图像处理的办法进行车牌识别。
- (2) 在车牌定位方面, 由于车牌的悬挂位置和车牌颜色的多样性, 需进一步提升程序的适应性, 提高定位精度。
- (3) 在车牌图像预处理方面, 考虑多种去噪措施, 使用图像增强的方法, 使其为下一步的车牌分割以及识别打好基础。
- (4) 文中所采用的 BP 网络算法, 虽然能实现字符识别的效果, 但是识别率有待进一步提高, 因为有先类似的字符可能会被混淆造成识别结果出错, 所以可以继续二次识别或者采用多种识别方法相结合。
- (5) 将 MATLAB 向 VC++ 程序改写, 利用 ARM 脱机处理板进行处理, 提高识别正确率的同时, 摆脱 PC 机的笨重。

## 参考文献

- [1]. 赵丹,丁金华.基于 Matlab 的车牌识别[J].大连: 大连大连理工学报, 2008.6, 14 (4): 44~48
- [2]. 冀小平.基于 Matlab 的车牌识别系统研究[J].北京: 电子设计工程,2009.11, 11 (3): 18~21
- [3]. 徐辉.基于 Matlan 实现汽车车牌自动识别系统[M]. 北京: 国防工业出版社, 2010.6
- [4]. 王爱玲, 叶明生等.MATLAB R2007 图像处理技术与应用[M].北京: 电子工业出版社, 2008.1
- [5]. 张德丰.MATLAB 模糊系统设计[M].北京: 国防工业出版社,2009.2
- [6]. 叶晨洲等.车辆牌照字符识别[J].上海: 上海交通大学学报,2000.4~6
- [7] 胡小峰, 赵辉; Visual C++/Matlab 图像处理与识别实用案例精选; 2004 年 5 月;
- [8] 周金萍; Matlab6.5 图形图像处理与应用实例; 2003 年 8 月;
- [9] 郝文化, 田蕾, 董秀芳; Matlab 图形图像处理应用教程; 2004 年 1 月;
- [9] 张引; 基于空间分布的最大类间方差牌照图像二值化算法; 浙江大学学报,2001 年 5 月,第 35 卷,第三期;
- [10] 杨斌; 汽车牌照识别系统相关技术研究实现; 西南交通大学硕士学位论文,2003 年 6 月;
- [11] Feng Wang, Lichun Manb, Bangping Wang, Yijun Xiao, Wei Pan, Xiaochun Lu; Fuzzy-based algorithm for color recognition of license plates[J]; Pattern Recognition Letters, 2008, 29(7): 1007~1020;
- [12] Wenjing Jia, Huaifeng Zhang, Xiangjian He; Region-based license plate detection[J]; Journal of Network and Computer Applications, 2007, 30(4): 1324~1333;
- [13] Mehmet Sabih Aksoy, Gültekin Çağıl, Ahmet Kür, sat Türker; Number-plate recognition using Inductive learning[J]; Robotics and Autonomous Systems, 2000, 33(2~3): 149~153 ;
- [14] Shamik Sural , P. K. Das ; Fuzzy Hough transform and an MLP with fuzzy input/output for character recognition[J]; Fuzzy Sets and Systems, 1999, 105(3):

489~497;

[15] 章毓晋; 图像处理和分析; 清华大学出版社,1999 年,第一版;

[16] 左奇, 史忠科; 一种基于数学形态学的实时车牌图像分割的方法; 中国图像图形学报,2003 年 3 月,第 8 卷(A 版)第 3 期;

[17]Chafik Djalal Kermad,Kacem chehdi; Automatic image segmentation system through iterative edge-region co-operation; Image and Vision Computing[J], 2002(3);

[18]Alineda Rocha Gesualdi, Jose Manoelde Seixas; Character Recognition in Car License Plates Based on Principal Components and Neural Processing;

[19] Xifan Shi, Weizhong Zhao, Yonghang Shen. Automatic License Plate Recognition System Based on Color Image Processing[J]. Springer-Verlag Berlin Heidelberg, 2005: 1159-1168

[20] ZhiWei.Yuan, PanXiaoLou. Vehicle license plate location algorithm [J]. Journal of kunming university of science and technology, 2001, 26 (2)

[21] 贺兴华,周媛媛,王继阳等.MATLAB 图像处理[M].北京:人民邮电出版社,2006:96-100

[22]刘佐濂, 邓荣标, 孔嘉圆.中国科技信息 [J].北京:中国科技报,2005, (23):9-12.

[23]张禹,马驹良,韩笑.车牌识别中的图像提取及分割算法[J].吉林大学学报,2006,44(3):408-411

[24]霍宏涛.数字图像处理[M].北京:机械工业出版社,2003.5

[25]许志影,李晋平.MATLAB 及其在图像处理中的应用[J].计算机与现代化,2004,(4):21-29

[26] 雷英杰. MATLAB 遗传算法工具箱及应用[M].西安:西安电子科技大学出版社,2005.

[27] 张强,王正林.精通 MATLAB 图像处理[M].北京:电子工业出版社,2009.

[28] D. S. Kim and S. I. Chien, Automatic car license plate extraction using modified generalized symmetry transform and image warping[ J ]. in Proc. IEEE Int. Symp. Industrial Elec2tronics, vol. 3, 2001, pp. 2022 - 2027

[29] 袁磊. 基于机器视觉的指针式仪表示值识别研究[D]. 湖北: 武汉理工大学, 2012

[30] 双维. 基于神经网络的车牌识别技术研究[D]. 山西: 中北大学, 2009

- [31] 冯慧娜, 白艳萍, 胡红萍. 一种基于颜色和灰度跳变的车牌定位方法[J]. 测试技术学报, 2009(6): 544—550
- [32] 王娜, 彭青玉, 邓保青. 医学图像背景分割[J]. 中国医学影像技术, 2010(8): 1573—1575
- [33] 李训青. 车型自动识别研究[D]. 北京: 中国科学院自动化研究所, 2003
- [34] 张玉璞, 杨旗, 张旗. 基于计算机视觉的图像多尺度识别方法[J]. 计算机应用, 2015(2): 501—506, 549
- [35] 李志海. 视频序列中车辆检测与跟踪算法的研究与实现[D]. 江苏: 南京邮电大学, 2009
- [36] 冯满堂, 马青玉, 成峰. 基于混合特征的多车牌定位算法[J]. 微计算机信息, 2009(9): 235—239
- [37] 王亮亮. 车牌识别技术中定位算法的研究[D]. 山东: 山东大学, 2012
- [38] 杜晓刚. 车牌识别系统中牌照定位、倾斜校正及字符分割技术的研究[D]. 山西: 中北大学, 2013
- [39] 丁坚. 智能停车场中的车牌识别系统研究[D]. 江苏: 南京航空航天大学, 2008
- [40] Jong-Bae Kim. MSER and SVM-Based Vehicle License Plate Detection and Recognition System[J]. Springer-Verlag Berlin Heidelberg, 2012: 529-535
- [41] Qiu Chengqun. Design of Automobile License Plate Recognition System based on MATLAB and Fuzzy PID[J]. 2013 Fifth Conference on Measuring Technology and Mechatronics Automation, 2000: 438-441
- [42] 严丽. 车牌识别系统的关键技术研究[D]. 南京: 南京理工大学, 2012
- [43] 杨超. 基于 Hadoop 平台的字符识别的研究[D]. 陕西: 西安电子科技大学, 2012
- [44] 洪健. 车牌识别相关技术的研究[D]. 安徽: 中国科学技术大学, 2007
- [45] 叶锋. 基于神经网络的综合集成车牌识别技术的研究及其应用[D]. 安徽: 合肥工业大学, 2002
- [46] 陈思. 一种 BP 神经网络学习率的改进方法[J]. 长春师范学院学报, 2010(4): 25-27



## 致 谢

本文是在导师顾玲嘉老师的悉心指导下完成的，值此即将毕业之际，衷心感谢顾老师在毕业设计上给予过的帮助。在毕业设计的过程中，从选题、立题、设计以及论文的撰写，顾老师悉心教导、实时督促。顾老师一丝不苟的治学态度、渊博的专业知识、执着的求学态度、丰富的科研经验以及平易近人的态度都另我深深地感到钦佩，为我未来的为人求学树立了一个好的榜样。

其次要感谢我的父母，妻子，同学和朋友们，是你们的帮助和支持才能让我顺利完成我的毕业设计。最后还要对百忙之中评阅本人毕业论文的各位专家与教授表达深深的感谢之情，由于本人才疏学浅，恳请各位专家学者批评指正。

## 附件

车牌识别系统 MATLAB 程序

```

%%%%%%%%----用于车牌识别系统----%%%%%%%%
%%%%%%%%----功能模块分为： 车牌定位 图像预处理 字符分割 字符识别---%%%%%%%%
%%%%%%%%----作者： JUNSUI----%%%%%%%%
%%%%%%%%----时间： 20160918----%%%%%%%%
clc;
clear;
%%%%%%%%----车牌定位----%%%%%%%%
I=imread('C:\Users\Administrator\Desktop\IMG_20160920_074745.jpg');
threshold1=0.5;      threshold2=1/20;
threshold3=2.5;      threshold4=[1/7,1/3,1/12,1/6];
I=rgb2gray(im);
[height,width]=size(I);
I_edge=zeros(height,width);
for i=1:width-1
    I_edge(:,i)=abs(I(:,i+1)-I(:,i));
end
I_edge=(255/(max(max(I_edge))-min(min(I_edge))))*(I_edge-min(min(I_edge)));%归一化处理
(0~255)
[I_edge,y1]=select(I_edge,height,width);    BW2 = I_edge;% 经

SE=strel('rectangle',[10,10]);
IM2=imerode(BW2,SE);
IM2=bwareaopen(IM2,20);
IM3=imdilate(IM2,SE); p_h=projection(double(IM3),'h');
if(p_h(1)>0)
    p_h=[0,p_h];
end
p_v=projection(double(IM3),'v');
if(p_v(1)>0)
    p_v=[0,p_v];
end
p_h=double((p_h>5));
p_h=find(((p_h(1:end-1))-p_h(2:end))~=0));
len_h=length(p_h)/2;
(p_v>5));
p_v=find(((p_v(1:end-1))-p_v(2:end))~=0));
len_v=length(p_v)/2;
%%%%%%%%%%%%%%
k=1;

```

```

for i=1:len_h
    for j=1:len_v
        s=IM3(p_h(2*i-1):p_h(2*i),p_v(2*j-1):p_v(2*j));
        if(mean(mean(s))>0.1)
            p{k}=[p_h(2*i-1),p_h(2*i)+1,p_v(2*j-1),p_v(2*j)+1];
            k=k+1;
        end
    end
end
k=k-1;

for i=1:k
    edge_IM3=double(edge(double(IM3(p{i}(1):p{i}(2),p{i}(3):p{i}(4))), 'canny'));
    [x,y]=find(edge_IM3==1);
    p{i}=[p{i}(1)+min(x),p{i}(2)-(p{i}(2)-p{i}(1)+1-max(x)),...
        p{i}(3)+min(y),p{i}(4)-(p{i}(4)-p{i}(3)+1-max(y))];
    p_center{i}=[fix((p{i}(1)+p{i}(2))/2),fix((p{i}(3)+p{i}(4))/2)];
    p_ratio(i)=(p{i}(4)-p{i}(3))/(p{i}(2)-p{i}(1));
end
%%%%%%%%%%
if k>1
    n=0;
    ncount=zeros(1,k);
    for i=1:k-1

        if(abs(p{i}(1)+p{i}(2)-p{i+1}(1)-p{i+1}(2))<=height/30&&abs(p{i+1}(3)-p{i}(4))<=width/15)
            p{i+1}(1)=min(p{i}(1),p{i+1}(1));
            p{i+1}(2)=max(p{i}(2),p{i+1}(2));
            p{i+1}(3)=min(p{i}(3),p{i+1}(3));
            p{i+1}(4)=max(p{i}(4),p{i+1}(4));
            n=n+1;
            ncount(n)=i+1;
        end
    end
    if(n>0)
        d_ncount=ncount(2:n+1)-ncount(1:n);
        index=find(d_ncount~=1);
        m=length(index);
        for i=1:m
            pp{i}=p{ncount(index(i))};

            %pp_center{i}=p_center{ncount(i)};

```

```

        pp_ratio(i)=(pp{i}(4)-pp{i}(3))/(pp{i}(2)-pp{i}(1));
    end
    p=pp;
    p_ratio=pp_ratio;
    clear pp;clear pp_ratio;
end
end
k=length(p)
m=1;T=0.6*max(p_ratio);
    if(p_ratio(i)>=T&p_ratio(i)<20)
        p1{m}=p{i};
        m=m+1;
    end
end
p=p1;clear p1;
k=m-1;
clear edge_IM3;clear x; clear y;    for i=1:k

    index=p{i};
    figure(5);I(index(1)-2:index(2),index(3):index(4));
end
if(k==1)
    imwrite(I(index(1)-2:index(2),index(3):index(4)),'cp.jpg');
end
function [y,y1]=select(ImageData,h,w)
thr=0.5;delta=0.05;
y=(ImageData>=thr*mean(max(ImageData)));
    BW2=bwareaopen(y,10);SE=strel('square',15);
    IM2=imdilate(BW2,SE); IM3=imerode(IM2,SE);
    average=sum(sum(IM3))/(h*w);
while(average<0.03||average>0.08)
    if(average<0.03)
        thr=thr-delta;
    else
        thr=thr+delta;
    end
    y=(ImageData>=thr*mean(max(ImageData)));
    BW2=bwareaopen(y,10);
    IM2=imdilate(BW2,SE);
    IM3=imerode(IM2,SE);
    average=sum(sum(IM3))/(h*w);
end
y1=y;
y=IM3;

```

```

function y=projection(I,s)
if(s=='h')
    y=sum(I);
end
if(s=='v')
    y=sum(I);
end
function e=qiege(d)
[m,n]=size(d);
top=1;bottom=m;left=1;right=n;
while sum(d(top,:))==0 && top<=m
    top=top+1;
end
while sum(d(bottom,:))==0 && bottom>=1
    bottom=bottom-1;
end
while sum(d(:,left))==0 && left<=n
    left=left+1;
end
while sum(d(:,right))==0 && right>=1
    right=right-1;
end
dd=right-left;
hh=bottom-top;
e=imcrop(d,[left top dd hh]);
function [word,result]=getword(d)
word=[];flag=0;y1=8;y2=0.5;
while flag==0
    [m,n]=size(d);
    wide=0;
    while sum(d(:,wide+1))~=0 && wide<=n-2
        wide=wide+1;
    end
    temp=qiege(imcrop(d,[1 1 wide m]));
    [m1,n1]=size(temp);
    if wide<y1 && n1/m1>y2
        d(:,1:wide)=0;
        if sum(sum(d))~=0
            d=qiege(d);
        else word=[];flag=1;
        end
    else
        word=qiege(imcrop(d,[1 1 wide m]));
        d(:,1:wide)=0;
    end
end

```

```

        if sum(sum(d))~=0;
            d=qiege(d);flag=1;
        else d=[];
        end
    end
end
end
%end

    result=d;
function [d]=main(jpg)
close all
clc
web -browser http://www.ilovematlab.cn/thread-23239-1-1.html
I=imread('car1.jpg');
figure(1),imshow(I);
I1=rgb2gray(I);
figure(2),subplot(1,2,1),imshow(I1);
figure(2),subplot(1,2,2),imhist(I1);
I2=edge(I1,'robert',0.15,'both');
figure(3),imshow(I2);t
se=[1;1;1];
I3=imerode(I2,se);
figure(4),imshow(I3);
se=strel('rectangle',[25,25]);
I4=imclose(I3,se);
figure(5),imshow(I4);
I5=bwareaopen(I4,2000);
figure(6),imshow(I5);
[y,x,z]=size(I5);
myI=double(I5);
tic
    Blue_y=zeros(y,1);
    for i=1:y
        for j=1:x
            if(myI(i,j,1)==1)

                Blue_y(i,1)= Blue_y(i,1)+1;
            end
        end
    end
end
[temp MaxY]=max(Blue_y);
PY1=MaxY;
while ((Blue_y(PY1,1)>=5)&&(PY1>1))
    PY1=PY1-1;
end
end

```

```

PY2=MaxY;
while ((Blue_y(PY2,1)>=5)&&(PY2<y))
    PY2=PY2+1;
end
IY=I(PY1:PY2,:,:);
Blue_x=zeros(1,x);
for j=1:x
    for i=PY1:PY2
        if(myI(i,j,1)==1)
            Blue_x(1,j)= Blue_x(1,j)+1;
        end
    end
end
PX1=1;
while ((Blue_x(1,PX1)<3)&&(PX1<x))
    PX1=PX1+1;
end
PX2=x;
while ((Blue_x(1,PX2)<3)&&(PX2>PX1))
    PX2=PX2-1;
end
PX1=PX1-1;
PX2=PX2+1;
dw=I(PY1:PY2-8,PX1:PX2,:);
t=toc;
figure(7),subplot(1,2,1),imshow(IY);
figure(7),subplot(1,2,2),imshow(dw)
imwrite(dw,'dw.jpg');
[filename,filepath]=uigetfile('dw.jpg','输入一个定位裁剪后的车牌图像');
jpg=strcat(filepath,filename);
a=imread('dw.jpg');
b=rgb2gray(a);
imwrite(b,'1.车牌灰度图像.jpg');
figure(8);subplot(3,2,1),imshow(b)
g_max=double(max(max(b)));
g_min=double(min(min(b)));
T=round(g_max-(g_max-g_min)/3);
[m,n]=size(b);
d=(double(b)>=T);
imwrite(d,'2.车牌二值图像.jpg');
figure(8);subplot(3,2,2),imshow(d)
figure(8),subplot(3,2,3),imshow(d)

h=fspecial('average',3);

```

```

d=im2bw(round(filter2(h,d)));
imwrite(d,'4.均值滤波后.jpg');
figure(8),subplot(3,2,4),imshow(d),title('4.均值滤波后')
% se=strel('square',3); % 使用一个 3X3 的正方形结果元素对象对创建的图像进行膨胀
se=eye(2); % eye(n) returns the n-by-n identity matrix 单位矩阵
[m,n]=size(d);
if bwarea(d)/m/n>=0.365
    d=imerode(d,se);
elseif bwarea(d)/m/n<=0.235
    d=imdilate(d,se);
end

d=qiege(d);
[m,n]=size(d);
figure,subplot(2,1,1),imshow(d),title(n)
k1=1;k2=1;s=sum(d);j=1;
while j~n
    while s(j)==0
        j=j+1;
    end
    k1=j;
    while s(j)~=0 && j<=n-1
        j=j+1;
    end
    k2=j-1;
    if k2-k1>=round(n/6.5)
        [val,num]=min(sum(d(:,[k1+5:k2-5])));
        d(:,k1+num+5)=0;
    end
end
end
d=qiege(d);
y1=10;y2=0.25;flag=0;word1=[];
while flag==0
    [m,n]=size(d);
    left=1;wide=0;
    while sum(d(:,wide+1))~=0
        wide=wide+1;
    end
    if wide<y1
        d(:,[1:wide])=0;
        d=qiege(d);
    else
        temp=qiege(imcrop(d,[1 1 wide m]));
        [m,n]=size(temp);

```



```

        all=sum(sum(temp));
        two_thirds=sum(sum(temp([round(m/3):2*round(m/3)],:)));
        if two_thirds/all>y2
            flag=1;word1=temp;    % WORD 1
        end
        d(:,[1:wide])=0;d=qiege(d);
    end
end
[word2,d]=getword(d);
[word3,d]=getword(d);
[word4,d]=getword(d);
[word5,d]=getword(d);
[word6,d]=getword(d);
[word7,d]=getword(d);
subplot(5,7,1),imshow(word1),title('1');
subplot(5,7,2),imshow(word2),title('2');
subplot(5,7,3),imshow(word3),title('3');
subplot(5,7,4),imshow(word4),title('4');
subplot(5,7,5),imshow(word5),title('5');
subplot(5,7,6),imshow(word6),title('6');
subplot(5,7,7),imshow(word7),title('7');
[m,n]=size(word1);
word1=imresize(word1,[40 20]);
word2=imresize(word2,[40 20]);
word3=imresize(word3,[40 20]);
word4=imresize(word4,[40 20]);
word5=imresize(word5,[40 20]);
word6=imresize(word6,[40 20]);
word7=imresize(word7,[40 20]);
subplot(5,7,15),imshow(word1),title('1');
subplot(5,7,16),imshow(word2),title('2');
subplot(5,7,17),imshow(word3),title('3');
subplot(5,7,18),imshow(word4),title('4');
subplot(5,7,19),imshow(word5),title('5');
subplot(5,7,20),imshow(word6),title('6');
subplot(5,7,21),imshow(word7),title('7');
imwrite(word1,'1.jpg');
imwrite(word2,'2.jpg');
imwrite(word3,'3.jpg');
imwrite(word4,'4.jpg');
imwrite(word5,'5.jpg');
imwrite(word6,'6.jpg');
imwrite(word7,'7.jpg');
liccode=char(['0':'9' 'A':'Z' '苏豫陕鲁']);

```

```

SubBw2=zeros(40,20);
l=1;
for l=1:7
    ii=int2str(l);
    t=imread([ii,'.jpg']);
    SegBw2=imresize(t,[40 20],'nearest');
    if l==1
        kmin=37;
        kmax=40;
    elseif l==2
        kmin=11;
        kmax=36;
    else l>=3
        kmin=1;
        kmax=36;
    end
    for i=1:40
        for j=1:20
            SubBw2(i,j)=SegBw2(i,j)-SamBw2(i,j);
        end
    end
    Dmax=0;
    for k1=1:40
        for l1=1:20
            if ( SubBw2(k1,l1) > 0 | SubBw2(k1,l1) <0 )
                Dmax=Dmax+1;
            end
        end
    end
    Error(k2)=Dmax;
end
Error1=Error(kmin:kmax);
MinError=min(Error1);
findc=find(Error1==MinError);
Code(l*2-1)=liccode(findc(1)+kmin-1);
Code(l*2)=' ';
l=l+1;
end
figure(10),imshow(dw),title(['车牌号码:', Code],'Color','b');

```