

中国某某大学中国某某大学

本科生毕业设计

姓 名： 潘传超 学 号： 22120798

学 院： 中国某某大学

专 业： 信息工程

论文题目： 数字仪表读数的自动识别系统

专 题：

指导教师： 华钢 职 称： 教授

2016 年 6 月 徐州

中国某某大学中国某某大学毕业设计任务书

专业年级 信息 12-1 学号 22120798 学生姓名 潘传超

任务下达日期：2016 年 1 月 10 日

毕业设计日期：2016 年 2 月 29 日 至 2016 年 6 月 15 日

毕业设计题目：数字仪表读数的自动识别系统

毕业设计专题题目：

毕业设计主要内容和要求：

本设计主要利用 Matlab 实现数字仪表读数的自动识别系统，要求完成以下工作：

- 1.研究有关图像处理理论。
- 2.研究有关模式识别有关理论。
- 3.设计识别系统结构。
- 4.编制识别程序。

指导教师签字：

郑 重 声 明

本人所呈交的毕业设计，是在导师的指导下，独立进行研究所取得的成果。所有数据、图片资料真实可靠。尽我所知，除文中已经注明引用的内容外，本毕业设计的研究成果不包含他人享有著作权的内容。对本设计所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确的方式标明。本论文属于原创。本毕业设计的知识产权归属于培养单位。

本人签名：_____

日期：_____

中国某某大学中国某某大学毕业设计指导教师评阅书

指导教师评语（①基础理论及基本技能的掌握；②独立解决实际问题的能力；③研究内容的理论依据和技术方法；④取得的主要成果及创新点；⑤工作态度及工作量；⑥总体评价及建议成绩；⑦存在问题；⑧是否同意答辩等）：

成 绩：

指导教师签字：

年 月 日

中国某某大学中国某某大学毕业设计评阅教师评阅书

评阅教师评语（①选题的意义；②基础理论及基本技能的掌握；③综合运用所学知识解决实际问题的能力；④工作量的大小；⑤取得的主要成果及创新点；⑥写作的规范程度；⑦总体评价及建议成绩；⑧存在问题；⑨是否同意答辩等）：

成 绩：

评阅教师签字：

年 月 日

中国某某大学中国某某大学毕业设计答辩及综合成绩

答辩情况					
提出问题	回答问题				
	正确	基本正确	有一般性错误	有原则性错误	没有回答
答辩委员会评语及建议成绩:					
答辩委员会主任签字: 年 月 日					
学院领导小组综合评定成绩:					
学院领导小组负责人: 年 月 日					

摘 要

随着信息技术的高速发展，数字化技术已经渗透到生活中的方方面面，越来越多的仪表采用了数字显示，但是在大部分数字显示仪表中，采用的仅仅是数显输出方式，而没有其它电压、电流、频率的输出，当需要对此类对象进行自动调校或抄表时，必须采用人工录入方式，既费时又费力，亦不能保证人工录入的准确性。因此需要设计数字仪表读数的自动识别系统，便于数字显示仪表信息的自动录入和计算机自动处理。

针对所研究问题本文展开一下几方面的研究与开发

- 1.对数字显示仪表的读数识别技术研究现状进行了分析。
- 2.对数字仪表读数自动识别系统进行了硬件结构与软件功能的总体设计。
- 3.对读数识别中涉及的关键技术研究，包括：图像的预处理、图像感兴趣区域的提取、图像的二值化、字符图像建模与识别。
- 4.对识别软件所涉及的模块进行了程序设计。
- 5.对系统软硬件进行了联合调试。

本课题通过设计一个数字仪表读数的自动识别系统，对仅仅是数显输出方式的数字仪表进行自动识别，而且该系统还支持对识别后的数据进行语音的播报、存储和传输。实验结果表明该系统能够正确的实现数字仪表读数的识别。

关键词：数字仪表； 读数识别； 图像处理； 图像识别； 字符识别；

ABSTRACT

With the rapid development of information technology, digital technology has penetrated into all aspects of life, more and more use of digital display meter, but in the majority of digital display instrument is only used in digital display output voltage and no other, current, frequency output, when the need for such objects automatically adjusts the meter reading or must be used manual entry way, time-consuming and laborious, but also can not guarantee the accuracy of manual entry. Therefore, the need to design a digital meter readings automatic identification system, automatic digital display for easy entry and computer instrumentation information automatically.

For research questions about several aspects of this article to expand research and development

- 1.the status readings for identification of digital display instrument analyzed

1. automatic meter reading digital identification system for the overall design of the hardware architecture and software features

3. study of the key technologies involved in reading recognition. Comprising: extracting preprocessing, image the region of interest of the image, the image binarization, character modeling and image recognition.

4. recognition software modules were involved in program design

- 5.on a joint software and hardware debugging.

This topic through the design of a digital instrument readings automatic identification system, only the digital output of the digital instrument automatically identified, and the system also supports the identification of the data after the voice broadcast, storage and transmission. Experimental results show that the system identification number of the meter readings can be implemented correctly.

Key words: digital instrument; reading identification; image processing; image recognition; character recognition;

目 录

1 绪论	1
1.1 课题背景及意义	1
1.2 国内外研究现状	2
1.3 课题的研究思路	2
1.4 研究内容及论文章节安排	3
2 开发技术的研究	4
2.1 软件系统开发	4
2.2 图像相关的理论知识	6
2.2.1 图像定义	6
2.2.2 数字图像	6
2.2.3 数字图像的发展	7
2.3 数字图像处理的内容简介	7
2.4 本章小结	7
3 系统的总体设计	9
3.1 系统设计目标和原则	9
3.2 总体设计	9
3.3 本章小结	13
4 各模块设计与实现	14
4.1 数字显示仪表的模拟	14
4.2 数字仪表读数图像采集	16
4.3 数字仪表图像预处理	18
4.3.1 数字仪表图像增强	19
4.3.2 数字仪表图像滤波	22
4.3.3 数字仪表图像感兴趣区域的定位	28
4.3.4 数字仪表图像感兴趣区域的提取	33
4.4 数字仪表图像识别处理	36
4.4.1 图像的边缘检测	37
4.4.2 ROI 区域旋转校正	39
4.4.3 ROI 区域图片错切矫正	41
4.4.4 单个字符图像建模	41
4.4.5 数字仪表的识别结果的输出	45

4.5 本章小结	51
5 总结与展望	52
5.1 总结	52
5.2 不足与展望	52
参考文献	54
附录	55
翻译部分	60
英文原文	60
中文翻译	70
致 谢	79

1 绪论

1.1 课题背景及意义

目前数显仪表的应用领域十分广泛，大多利用液晶或者 LED 数码管作为数字显示元件，但是大部分数字显示仪表中采用的仅仅是数显输出方式而没有其它电压、电流、频率的输出，当需要对此类对象进行自动调校或抄表时，必须采用人工录入方式，既费时又费力，亦不能保证人工录入的准确性。因此需要设计数字仪表读数的自动识别系统，便于数字显示仪表的信息自动录入和实现计算机的数据自动处理。

如果能找到一种合适的方法，以最廉价的介入方式，让那些仅仅只有数显而没有其他形式信号输出的仪表也能实现仪表数据的自动调校、抄表或其他形式数据的输出，那么就能避免因通过人工数据录入方式而造成不必要的劳动力浪费。

提出这个问题的同时，我们知道当今科技发展的高度不是短时间内技术的突破，而是长时间的技术沉淀，从一个时代到另一个时代的跃进是一个漫长的过程，老式电子设备向新式智能化产品的更替过程同样是一个漫长的过程，我们不可能说老式电子设备能一次性地全部被新式电子设备完全替换掉。比如在当今的互联网时代，大部分电力提供商希望所有的电力控制柜都能接入到互联网中，于是出现在面前的巨大矛盾就是更换掉所有不具有网络接入功能的过时产品同时与之产生的巨额更换成本，这显然是不可取的做法，或许能做的仅仅是逐步地将设备进行更替，或者找一些中间的联网输送介质，比如通过 485 转以太网 232 转以太网等转换模块进行设备的升级，以实现设备联网管控。

所以在仅仅只有数显输出而没有其他形式信号输出的数显仪表类产品中，需要找到一种起媒介作用的中间介入对象，来实现这类仪表的数据自动调教、自动抄表功能也就变得尤为重要了。那么在这类仅仅只有数显输出而没有其他形式数据输出的仪表中，如何找出通用的仪表数显转成其他形式输出信号的中间介质对象呢？基于此问题，首先要对仪表数显特性进行了分析，仪表类的数显输出特性很明显是将自身的电信号转换成数显方式表达的图像信号。也就是需要找到一种能对

仪表图像进行识别并能将仪表的图像信号转换成其他形式的输出信号，实现仅仅只有数据显示为没有其他形式数据输出的该类仪表自动调校或自动抄表。

由此本课题提出了一种数字仪表读数的自动识别系统，作为仅仅只有数显而没有其他形式信号输出的仪表类对象进行二次数据传输的中间介质，实现这部分仅仅只有数显输出而没有电流、电压或频率输出的仪表进行其他形式数据二次传输，从而方便其他设备对该类仪表进行自动调校或抄表。

1.2 国内外研究现状

自从 1929 年光学字符识别的提出，经过近一个世界的发展现在其应用的产品已经纷繁众多，比如扫识别技术，人脸识别技术等先进的光学图像识别技术^[1]。数字自动识别系统作为光学字符识别中的简单技术，在日常生活中已经存在广泛的应用，比如常见的快递单号数字识别，该识别打破了传统的条形码扫描的局限。又如食品、酒瓶等数字编号的识别，又如交通监测系统中的车牌号识别等^[1]。

在国内主要还是运用在快递单号和车牌号等的识别中，在国外一般数字的自动识别系统在工业生产线上的应用是比较常见，例如生产车间的包装工号识别，产品对应编号的识别。然而在基于数字仪表的自动识别系统，其应用在国内外还是比较少有的，并非是识别技术的不成熟，而是由于传统的人工调校和人工抄表思想比较根深，对数字仪表从手动到自动的调校和录入方式的适应需要一段适应的时间，就像当年非智能手机到智能手机的普及一样，用户的增加是需要一个渐变的过程。

总之国内外数字仪表读数的自动识别系统开发技术还是比较成熟的，但是具体针对仪表类读数的自动识别软件设计还是比较少见。

1.3 课题的研究思路

本课题通过设计一个数字仪表读数的自动识别系统，对仪表的图像数据自动识别并对数据进行存储，便于自动采集仪表数据，从而节省劳动成本。本课题设计的数字仪表读数的自动识别系统能很方便的对仪表数据进行识别和录入。

本课题的研究思路首先经过三部分的字符图像识别处理，再对识别出的数据进行存取等操作，研究思路如图 1-1 所示。先进行含字符的图像采集，用以获取有

用的字符信息图像，再进行图像的处理，图像处理内容是图像增强、图像的滤波、图像的特征区域提取和字符图像的分割，最后对分割的有用字符图像进行识别和数据存储、数据分析和数据处理。

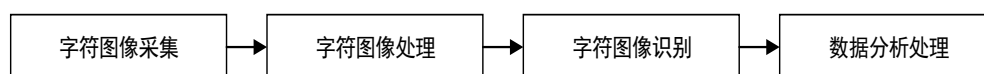


图 1-1 课题研究思路

1.4 研究内容及论文章节安排

本设计主要运用图像增强算法、图像的滤波算法、图像边缘检测算法、图像旋转算法、图像错切算法、图像感兴趣区域定位算法、图像分割算法、图像字符识别等算法，并利用 Matlab 的强大数据分析功能，对采集到的图像进行图像处理、识别、数据分析、存储。本文的章节安排如下：

第一章，绪论部分主要介绍课题的研究背景及意义、图像模式识别的国内外发展现状、课题的研究思路以及课题的研究内容及论文章节安排。

第二章，主要介绍系统的开发技术，包括软件系统开发、图像相关的理论知识及数字图像处理的内容。

第三章，主要是系统的总体设计部分，简述系统设计的目标与原则，对系统进行总体描述和功能分析。

第四章，数据仪表读数的识别，根据系统功能的实现过程，本节主要分为数字显示仪表的模拟，数字仪表读数图像采集，数字仪表图像处理和数字仪表图像的识别。

第五章，总结与展望，对系统已完成的部分加以总结，提出不足以及可以继续扩展的部分。对更高性能、全面的设计加以展望。

2 开发技术的研究

2.1 软件系统开发

1. 开发硬件环境

本课题研究的是数字仪表读数的自动识别系统，需要对被识别的图像进行图像处理，比如图像的预处理，图像的噪声处理和图像的分割识别处理等。由于每一幅图片的数据量比较大，因此对处理器的性能要求较高，在产品中一般用数字图像处理芯片 DSP 或者用 FPGA 等高速的处理芯片，由于笔者的时间和精力有限，所以笔者将省去电路搭建的过程，直接在笔记本电脑上对仪表图像数据进行处理。

由于对处理单元的性能要求较高，所以选择的计算机性能将会影响图像的识别速度。本系统开发的计算机平台：CPU 主频 2.4GHz，硬盘 500GB，内存 4G。

2. 辅助设备支持



(a) USB 摄像头



(b)数显表头

图 2-1 摄像头和模拟表头

本软件系统经过外接 USB 摄像头，对含字符数据图像进行采集，摄像头如图 2-1(a)所示。本实验中的字符识别图像的识别对象是 51 单片机控制的数显表头，通过其模拟含字符的图像，由此对该模拟对象进行数字图像识别，模拟硬件设备如图 2-1(b)所示。

3. 软件开发环境

基于图像的主流算法开发软件有 Matlab 和 Opencv 等，其中 Matlab 更多的用于对算法的研究和学习，而 Opencv 则更多的被用于商业软件的开发^[2]。

OpenCV 支持多种操作系统, 比如 Linux、Windows、Mac OS, 该软件还支持 Python、Ruby、Matlab 等语言的开发, 拥有众多图像处理的相关算法库, 现在其大部分的算法库都采用的是 C++ 语言, 进一步提高了软件执行的效率, 该软件是视觉图像处理领域中具有影响力的开源软件^[3]。又由于其代码具有很好的可移植性, 所以基于 OpenCV 开发出的算法也能较为容易地被移植到其他处理器中, 比如 ARM 和 DSP 或者 FPGA 或者其他更加高级的 GPU 图像处理芯片中。

Matlab 是由美国的 MathWorks 公司出品的商业数学软件, 主要由 Matlab 和 Simulink 两部分组成, 由于其强大的矩阵运算处理能力, 一般用于算法的研究, 数据的分析处理等 Matlab 是由两个词的组合, matrix 和 laboratory 即矩阵和实验室的意思组成。其最开始是由美国新墨西哥大学计算机科学系主任 Cleve Moler 为了减轻学生编程的负担, 用 FORTRAN 编写了最早的 MATLAB。Moler、Little、Steve Bangert 在 1984 共同合作成立了 MathWorks 公司, 并成功的把 MATLAB 推向了市场, 20 世纪 90 年代 Matlab 成为了国际上的标准数据处理软件^[4]。

Matlab 是一款跨平台的数据处理软件支持 Linux 操作系统、Windows 操作系统、苹果电脑操作系统, 拥有众多的数据处理、数据分析、电路仿真等工具箱。Matlab 集成了 M 文件开发的运行环境, 包括 Matlab 桌面、命令行窗口和脚本编辑、函数编辑、GUI 界面、仿真操作界面等操作界面, 而且配备了丰富并且全面的帮助文档, 提供大量的函数操作例程, 能很快的帮助初学者入手学习。Matlab 包括了大量的数学、图像、航天、生物、医疗等函数库。软件内包含大量的算法库, 算法库中又有大量的算法函数, 比如神经网络、小波、傅里叶变换、快速傅里叶变化、拉普拉斯变换、三角函数、图像拟合、中值滤波、Sobel 算子、Roberts 算子、Prewitt 算等丰富的函数库^[5]。Matlab 支持较多的开发语言, 也有很多应用程序的接口(API), 能方便的使用 C、VB、Fortran 等高级语言进行脚本文件的编写^[5]。重要的是 Matlab 语言语法简单便于上手。

本课题是数字仪表读数的自动识别系统, 主要是基于仪表读数识别进行的学习和研究, 并不涉及软件应用的商业化开发, 为了便于学习和研究笔者选择的软件平台是 Matlab。

Matlab 具有强大的数据处理功能，内部集成大量的数据处理快速实现的算法，具有很好的学习和研究价值，比如简单的数据拟合，可以根据你提供的离散数据，和数据拟合的阶次，完成数据拟合并返回数据拟合曲线方程的系数，在一般的嵌入式控制数据分析中将会给我们带来很大的方便，代码根据 Matlab 自带的函数能很容易的实现，简单的数据拟合代码如下

```
a=[1.68,2.1,2.5,2.854,2.966,3.0];  
b=[0.2,0.22,0.34,0.57,0.6,0.46];  
plot(a,b,'g','markersize',25);  
hold on ;  
p3=polyfit(x,y,2);  
plot(x,y);
```

Matlab 也具有强大的图像处理功能，能很方便的处理二维和三维的图像数据。本设计就是基于 Matlab 强大的图像数据处理功能设计的数字仪表读数的自动识别系统。

2.2 图像相关的理论知识

2.2.1 图像定义

图像是各种图形图案的总称，图像是客观对象相似性的一种描述总称，既是信息表达的方式也是信息的载体，常用于表达对象的特征。据统计人们 75%的信息都来源于视觉图像。图片信息的载体来源有纸质图案、液晶显示屏和投影仪等。图像按其表示形式可分为模拟图和数字图。模拟图通过某种信号（光、电...）的强弱来表示，而数字图像是通过计算机存储的二进制形式来表示，本文涉及图片均指数字图片。

2.2.2 数字图像

数字图像是一个二维的图像矩阵，其基本单元是像素，它可以看作是几何或者光学的校准器。数字图像又称数码图像或者数位图像，是二维图像用有限数字值像素的一种表示方式，数字图像处理内容就是研究他们的变换算法。图像的表

现形式各式各样，有可见的图像也有不可见的图像，有真实的图像也有非真实的虚拟图像，也有模拟图像和数字图像等，总之图像的存在形式多种多样。

2.2.3 数字图像的发展

图像在信息的获取、传输和表达中起着重要的作用，因此数字图像处理技术已成为信息科学技术、计算机科学技术、工程、生物、航天、医疗等技术领域的重要研究内容^[6]。



图 2-2 第一张数字图片



图 2-3 第一台数字扫描仪

图 2-2 是从扫描、存储、和重新经数字化像素显示在老式的电脑上的效果图，图 2-3 是数字图片出现在 20 世纪 60 年代 Bartlane 的数字传真机器。图像处理技术通过太空计划的推进、医学的研究，以及 Jet Propulsion Laboratory、MIT, Bell Labs 和 The University of Maryland 等知名研究机构的涉足，使得图像处理技术发展迅速。如今的数字图像相关技术已经广泛的应用于生活中，比如交通监管系统中车牌号的识别，防盗设备或者签到机的指纹识别和其他它的密码机器上的虹膜识别等。

2.3 数字图像处理的内容简介

数字图像处理的目的是为了便于对图像数据进行分析和对图像信息进行改进以及为了方便计算机的存储、传输的现实，因此图像处理也称为计算机图像处理。处理的主要内容有图像的增强，图像的降噪，图像的各种变换，和感兴趣区域的提取等。

2.4 本章小结

本章主要对数字仪表读数的自动识别系统开发背景及理论知识进行简单介绍。主要介绍系统的开发环境和数字图像相关知识。本软件开发是利用 **Matlab** 语言自行设计的多种图像识别算法，对含字符的图像数据进行了识别处理。本部分重点介绍了数字图像相关发展，为下面介绍软件中图像识别算法的引入做好准备。

3 系统的总体设计

3.1 系统设计目标和原则

在进行软件系统设计前先对软件系统进行了需求的分析，对设计方案进行了全方位的讨论、综合，并整理了为实现数字仪表自动识别系统的思路，对整体的设计流程进行了更有效率的布局。本设计的目的是以最低廉的方式，最有效率的运算分析，搭建具有核心技术的设计软件。因此设计时需要遵守以下的原则

1.低成本

仪表模式识别相关产品的推行比较缓慢不够普及还有一方面原因可能是产品性能的不稳定，另一方面可能是成本的不低廉。不过笔者认为占主要原因的可能是产品的成本比较高，这极大的影响了仪表自动识别系统相关产品的普及。因此本设计利用更低的设计成本，更易于推广的系统运行方式，满足大部分需要数字仪表自动识别系统相关领域的需求。

2.可移植性

在使用中对软件性能的不断完善，并对软件进行功能的调整，才能使软件更加完善。因此本软件在设计阶段就留有升级的空间便于日后升级调试，每个算法单独组成函数体，增强算法处理的兼容性和可移植性。

3.简单实用

本系统设计应适用于大部分的数字仪表的识别，界面友好，操作简单，不需要图像相关的理论知识，只需要点击识别按钮，便可以对仪表数据进行识别分析。无需刻意学习，上手相对容易。

4.稳定耐用

稳定性是衡量一个系统性能的关键因素，所以在软件调试过程中建立了多种噪声模型，对仪表识别进行了误差系数的统计，并对误差进行了矫正，最后在相对噪声环境下误差几乎可以忽略。

3.2 总体设计

数字识别算法有很多种，比较常见的有：统计模式识别、结构模式识别、模糊

模式识别和人工神经网络识别。本文采用的是 Matlab GUI 界面进行软件设置，主要通过图像的获取、图像的增强、图像的定位、图像的分割、图像的特征值识别等步骤完成。

首先通过摄像头捕获数字仪表的图像数据，将捕获的图片经过图像的滤波、变换等修正手段，对图片做初步处理，图片经直方图统计、对比度增强等相关算法处理，将环境背景噪声尽量抑制掉，再利用所有像素在不同光照强度下灰度统计特性的不同提取合适的阈值并对图片二值化，再利用数码管显像的特点定位出图像的感兴趣区域，接下来再对图像的感兴趣区域进行分割提取出图像的感兴趣区域，并对感兴趣区域进一步的处理，最后是对图像的模式识别处理。

其中图像的识别有多种方法，有的是基于模板的匹配模型，此模型在算法上具有算法简单特点，但运算量较大并且具有很多的不稳定因素，还需要配置较大的模板数据库，对设备的硬件要求比较高，不适合本课题的研究，该方法在运用中资源浪费比较严重，并且对使用的条件比较严格，对背景噪声影响较为严重，所以在本课题设计中相对其他识别方案而言不具有应用价值。

因此本文根据统计学特征提出了两种更方便的识别算法，一般数码管，或者 LCD 屏的数显方式比较固定，本设计主要针对数字显示的字符识别处理。本文第一种识别算法采用数码管分段显示原理，对 8 段式数码管的 a,b,c,d,e,f,g,h 段分别识别，再对识别结果进行数字的匹配，比如数字 1 则 8 段识别中为 b,c 段为点亮，其他段为 led 熄灭的状态。当然为了减少运算量，将数码管分割成 6 个区域，对 6 个区域的识别结果，统计出 a,b,c,d,e,f,g,h 段的结果，达到数字的识别。具体的识别方法在第四章有详细的介绍。第二种识别算法基础是对数字建立显示模型，分析模型中数字成像的特征。对于数字‘1’，根据其宽和高的比例关系，很容易将其与其他数字区分开来，对于其他数据我们只需要逐一的按其形态学原理进行分析，具体算法及实现过程在第四章中会进行详细的介绍。最后通过串口将数据发给下位机，下位机接收到指令后将执行相关功能。

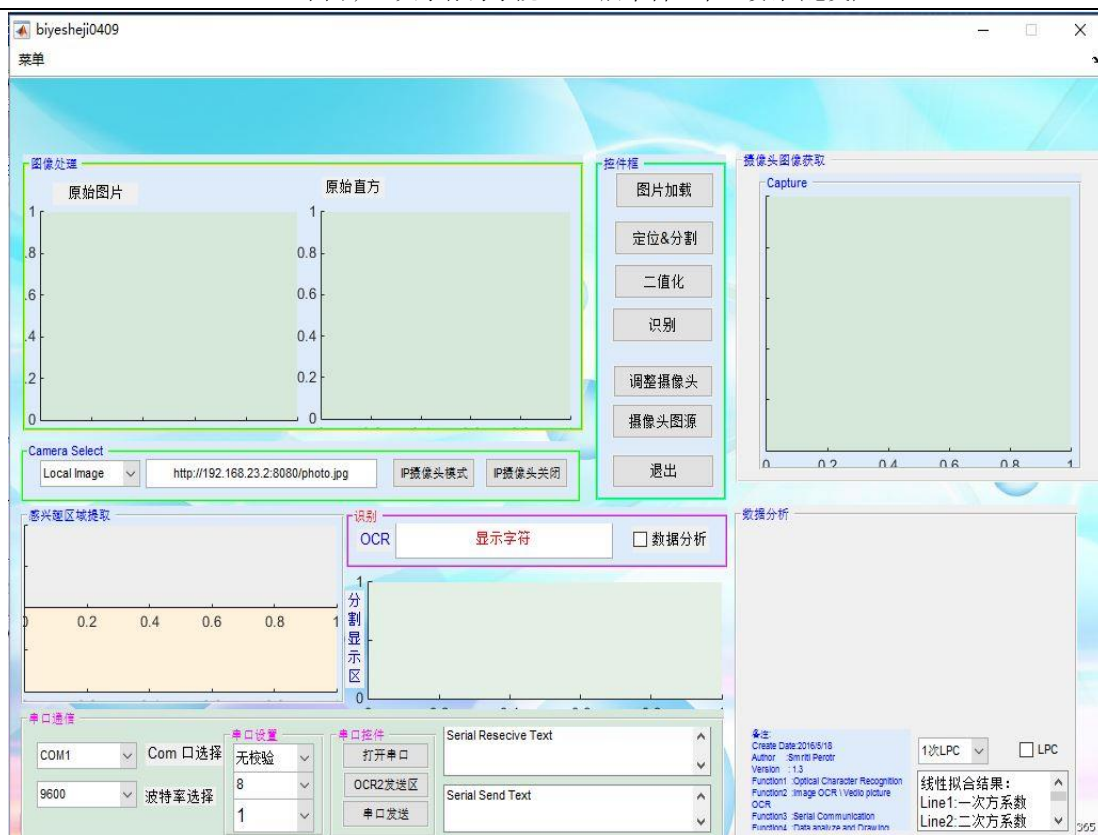


图 3-1 UI 设计

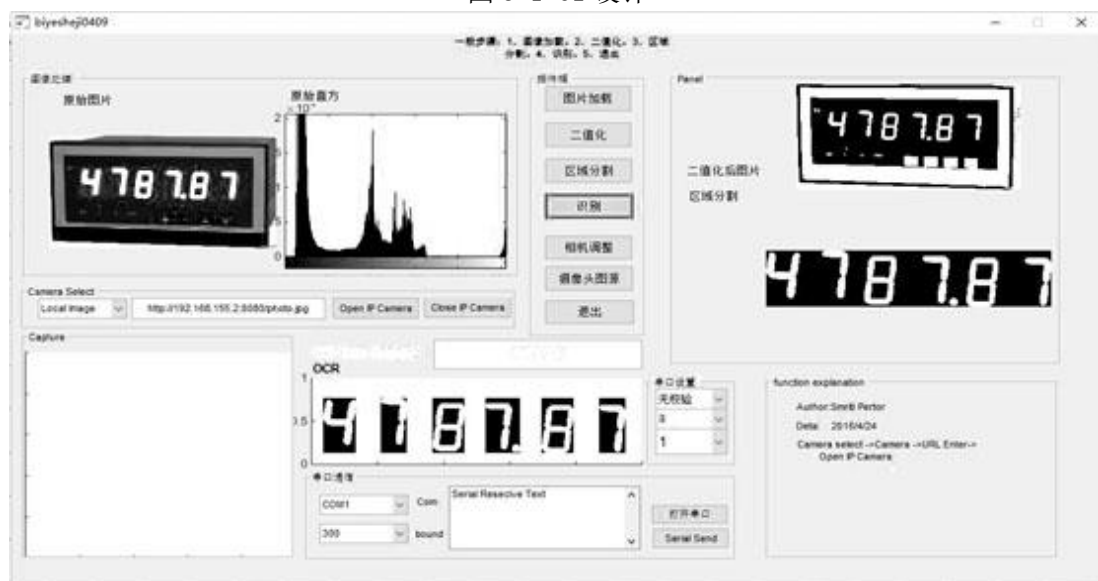


图 3-2 识别效果图

软件界面如图 3-1 所示，整个设计概念图如图 3-2 所示，软件识别结果如图 3-3 所示。

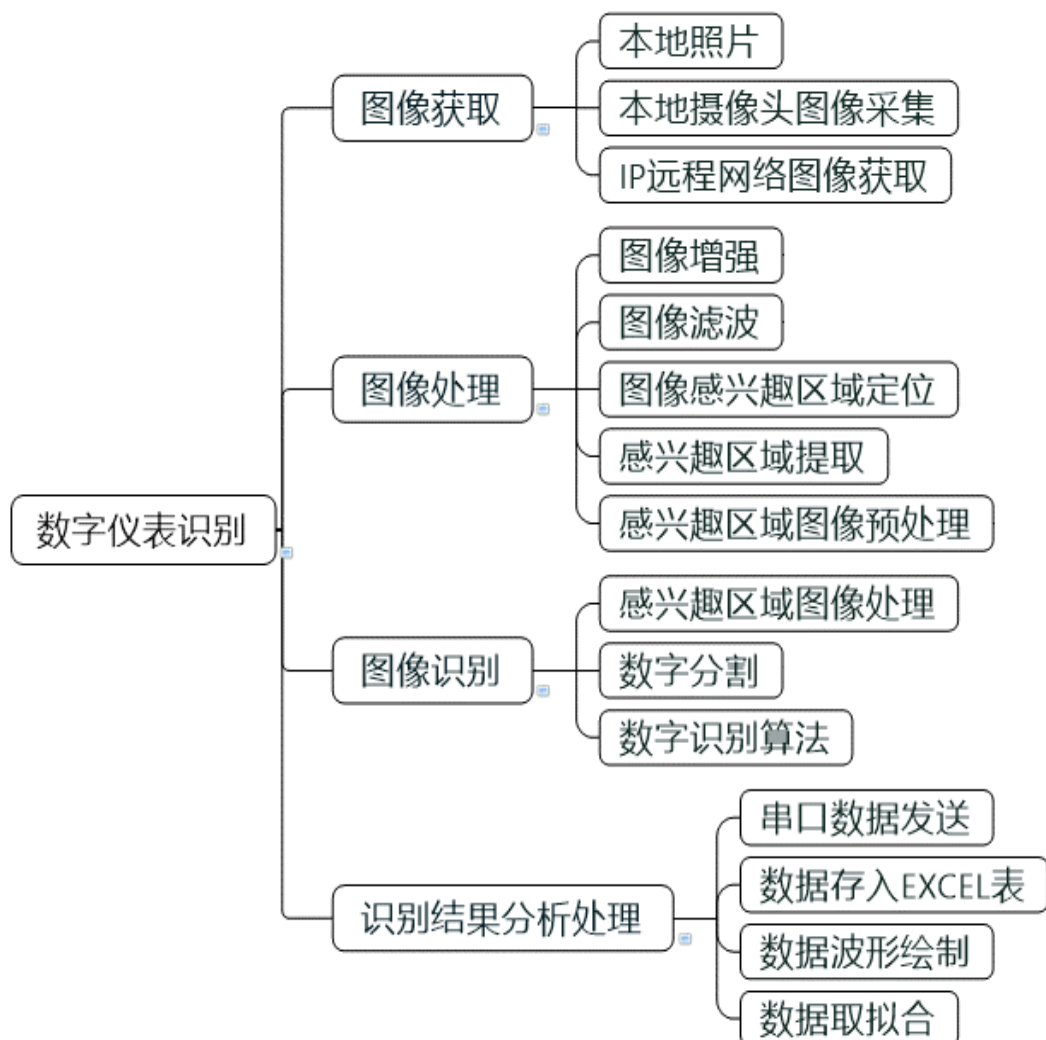


图 3-3 整体概念图

软件的最终目的是对图像上包含的字符数据进行识别，为了便于看清演示结果，对于本地的图像我们进行了各部分算法的单独实现并呈现结果，一般我们先加载图像，其次区域定位、图像的二值化、图像的区域分割、字符分割、图像的识别、最后进行数据处理，数据的处理包括数据的显示，数据的传输，数据的离散值分析，以及方程式的提取等。系统设计的流图如图 3-4 所示。

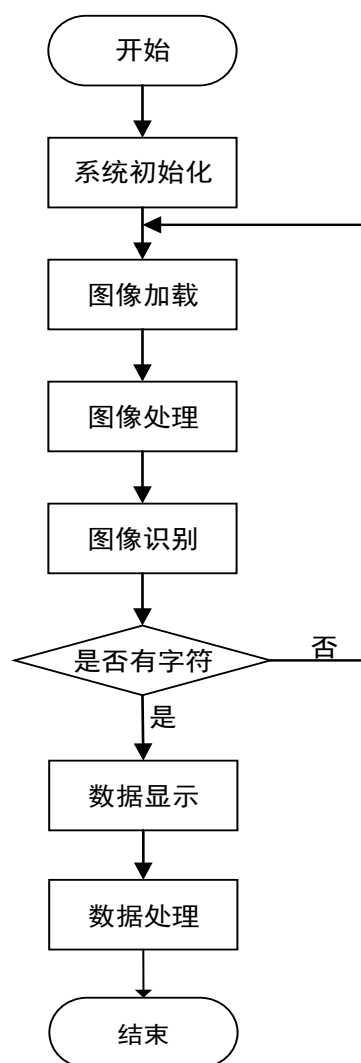


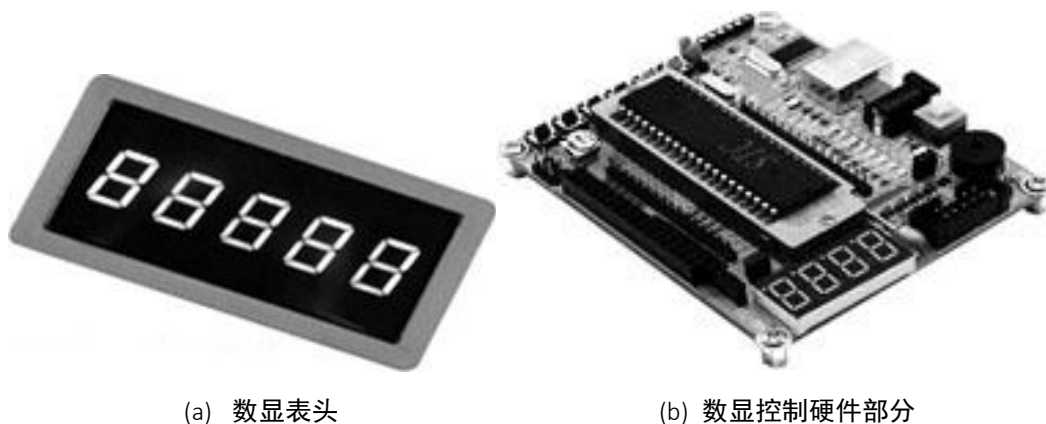
图 3-4 系统设计流程

3.3 本章小结

本章系统的总体设计是对系统进行的构思，提出设计方案，验证方案可行性的关键。明确设计目标在软件设计中会对全局把握更加得当，是设计思路的整体体现，使设计有征对性，使设计方案实现更加顺利，能起到事半功倍的效果。本章主要介绍系统的总体设计，首先对市场需求进行分析和对产品进行的定位，明确系统设计的目标以及原则，对整个构思进行总体的描述以及分析。对本设计涉及的关键算法进行简要的思路罗列，明确本设计软件的工作任务。

4 各模块设计与实现

4.1 数字显示仪表的模拟



(a) 数显表头

(b) 数显控制硬件部分

图 4-1 模拟设备

本课题的数字仪表读数的自动识别系统的数字显示部分是用 51 单片机系统与 LED 数字显示表头模拟组合而成（如图 4-1(a)图所示）。数字显示表头具有以下特点

1. 5 位高亮红色数码管显示屏模块有四个螺丝孔安装方便。
2. 内部使用 5 个贴片 74HC595 驱动, 数码管每段均带电阻限流保证亮度均匀。
3. 静态驱动, 在数字不变的情况下, 单片机不用一直刷新, 非动态扫描, 节约单片机系统资源。
4. 接口简单 VCC 正极, GND 负极, SDI 数据, 移位时钟 SCLK, 锁存 LOAD, 仅需 3 根线与单片机接口节约单片机 IO。
5. 有数据输出方便级联, 例如 4 个数码管模块（不论几位）可以一个连一个, 最前面的表头接单片机即可。

控制表头进行数字显示的执行设备是一款简单的 51 单片机学习板：集成蜂鸣器、数码管、LED、EEPROM、液晶、温度、红外、独立按键等众多功能。该学习板电路如图 4-1(b)图所示, 在实验中我们只用到了学习板上单片机引脚引出的端口, 从而实现了数字表头模拟显示的控制。控制表头模拟数字仪表显示的程序流程图如图 4-2 所示, 完整的代码见附录, 表头部分显示代码如下。

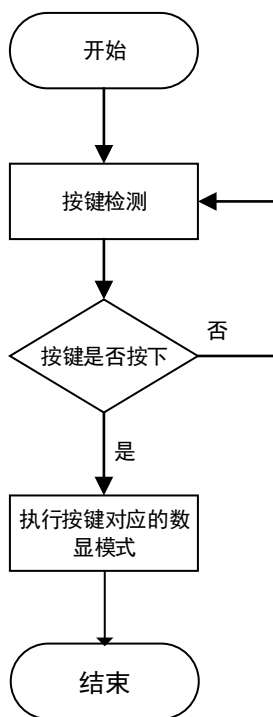


图 4-2 表头显示的程序流程图

```

sbit  LOAD=P2^2;
sbit  CLK=P2^1;
sbit  SDK=P2^0;
void SPI_595(unsigned char out_data)
{
    unsigned char i,temp;
    for(i=0;i<8;i++)
    {
        CLK=0;
        temp=out_data&0x80;
        if(temp==0x80)
            SDK=1;
        else SDK=0;
        out_data=out_data<<1;
        CLK=1;
    }
}
void show (void)//显示代码
{
    unsigned char i;
    for(i=0;i<20;i++)

```

```

{
    LOAD=1;
    LOAD=0;
    delay10ms(500);
    SPI_595(sjfx[i][4]);
    SPI_595(sjfx[i][3]);
    SPI_595(sjfx[i][2]);
    SPI_595(sjfx[i][1]);
    SPI_595(sjfx[i][0]);
}
for(i=20;i>0;i--)
{
    LOAD=1;
    LOAD=0;
    delay10ms(500);
    SPI_595(sjfx[i][4]);
    SPI_595(sjfx[i][3]);
    SPI_595(sjfx[i][2]);
    SPI_595(sjfx[i][1]);
    SPI_595(sjfx[i][0]);
}
delay10ms(500);
}

```

4.2 数字仪表读数图像采集

在本设计中,图像获取方式有三种,第一种方式是选择本地照片,对本地照片上的数字将进行模式识别,第二种方式是用本地摄像头采集数字显示设备的图像,对图像进行处理识别。第三种方式是通过 IP 地址获取网络上的图片并对图片上的数字进行识别,但是这三部分的图像获取后的最终形式都是一幅 RGB 真彩色图像,所以三种图像获取的最终形式是一样的。



图 4-3 数字图像

本地图片是指预先存储到设备上的含有用的字符信息的数字仪表图像，可以是 jpg、png、bmp 等彩色图像如图 4-3 所示，本地图片的读取并显示的代码如下。

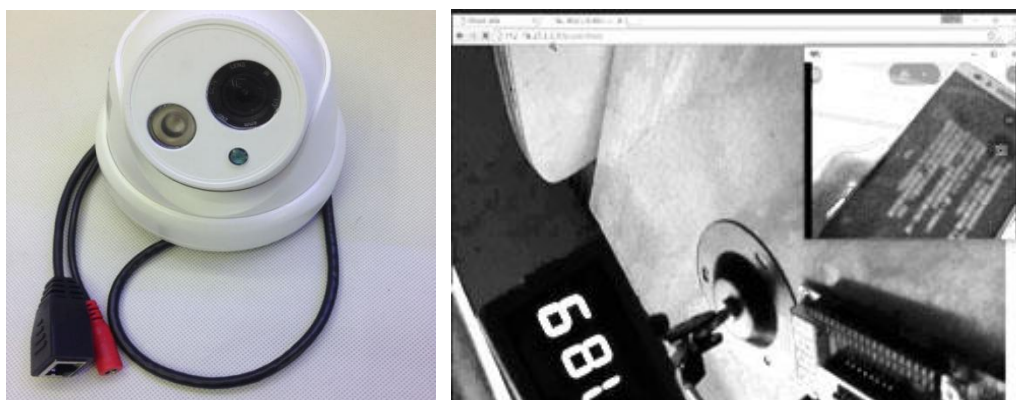
```
[filename pathname]=uigetfile({'*.png'; '*.jpg'; '*.jpeg'; '*..*'}, '选择图片');
handles.path=[pathname filename];
if ~ischar(handles.path)
    disp('没正确选择文件请从新选择! ');
    return ;
end
handles.image1=imread(handles.path);
    %读取图像文件
z=handles.image1;
axes(handles.axes1);
    %图片显示
imshow(handles.image1);
```



图 4-4 图像数字化

摄像头图像获取是通过摄像头采集图像，如图 4-4 所示。一般模拟摄像头采集图像是通过数字采集卡将模拟的感光元件获取的模拟图像信息转成数字图像信息，最终存储到计算机中，不过现在已经有数字的感光传感器件能直接输出数字图片的感光信息。摄像头图像获取和显示的代码如下。

```
obj=webcam('USB2.0 PC CAMERA');
    %摄像头选择
VediorgbImage = snapshot(obj); % Acquire a single image.
axes(handles.axes6);
imshow(VediorgbImage);
```



(a) 网络摄像头

(b) 网络摄像头图像采集

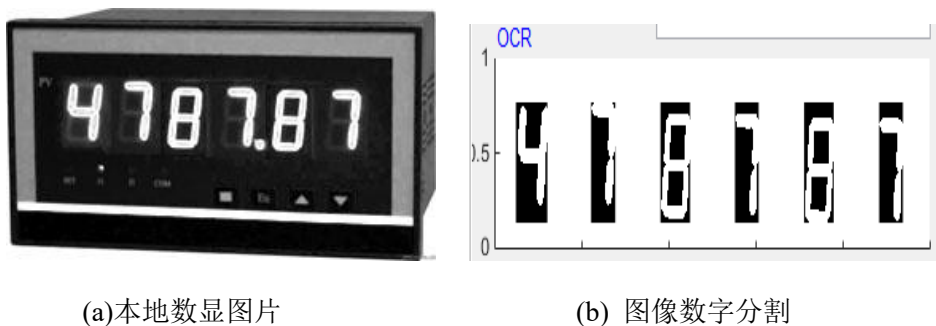
图 4-5 IP 摄像头图像获取

IP 摄像头图像的获取一般需要有 IP 摄像头（如图 4-5 (a)），也就是将一般的摄像头图片实时上传到指定的 IP 地址上，效果如图 4-5(b)所示是软件上读取的网络摄像头图像，其实也就是通过直接访问指定的图像 IP 地址，下载指定 IP 上的 RGB 图像。具体的图像采集和显示的代码如下。

```
IPURL=get(handles.IPText,'string');
IPim=imread(IPURL);
imshow(IPim);
```

4.3 数字仪表图像预处理

图像处理是整个识别过程的关键技术，本节的图像处理分为图像的增强处理、图像的滤波处理、图像的感兴趣区域的定位和感兴趣区域的提取。每个图像处理的过程都是图像识别过程实现的关键，所以本设计对每个涉及细节都进行了认真的思考。本节对每个处理部分都是先给出实现效果，通过有结果再做解释的方式，解释实现过程。经本节图像处理前后的图像效果如图 4-6 所示。



(a) 本地数显图片

(b) 图像数字分割

图 4-6 图像处理前后效果

图 4-6 (a)为原始图像, 图 4-6 (b)为图像处理后的效果图。我们所感兴趣的区域只是数字显示的部分, 所以仪表壳体和背景对于我们的数字图像处理都是一种干扰, 所以我们第一步将是去除干扰。在对图像进行识别前的图像处理, 是整个图像识别的关键。将仪表和背景图像进行去除, 使感兴趣区域更加突出, 为下一步作识别做好了充分准备。



图 4-7 图像处理的步骤

图 4-7 为图像处理的步骤,在设计过程中,笔者对算法进行了多次优化,也体现出了笔者为实现更有效的算法而不断的尝试的精神,软件设计也进行了多次算法改版,图 4-8 为图像处理概念图。

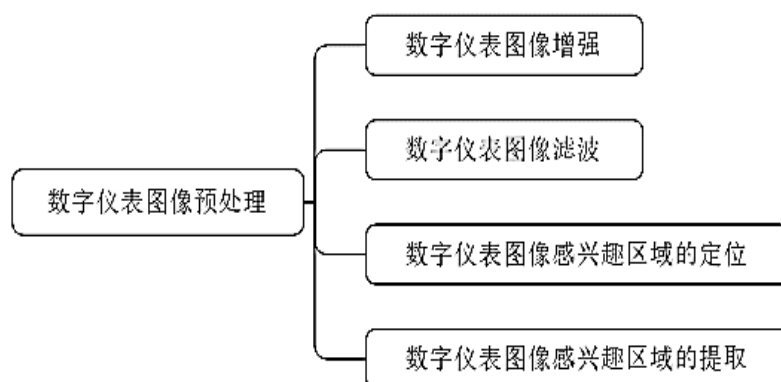


图 4-8 图像预处理的概念图

4.3.1 数字仪表图像增强

1. 仪表图像增强的原因

数字仪表图像增强的目的是改善图像的视觉效果展现和突出图像的兴趣区域有针对性地扩大图像感兴趣区域和凸显与图像背景噪声的差异, 从而改善图像质量和丰富图像信息以及改善图像判读和识别效果等, 使其满足某些特殊的图像分析需要。

2. 仪表图像增强处理步骤

由于图像的增强有空域增强和变换域增强, 变换域增强的一般的处理方式是通过对傅里叶变换、离散余弦变换、沃尔什变换、拉普拉斯变换或是小波变换等变换

算法,将图像从空域变换成相应的变换域,可以完成比较复杂的图像处理^[8]。但是目前变换域的图像处理方式,其图像处理的运算速度还很受限制于硬件电路,虽然现在也有很多快速运算的算法,但是还是改变不了运算速度慢的特点,然而空域增强具有算法简单,运算量也就存在二维平面的处理,也可以对图像进行小规模的空域滤波处理,运算的算法结构也不太复杂,易于上手,并且处理简单的图像已经足够了,一般空域变换有邻域处理算法和点处理算法。

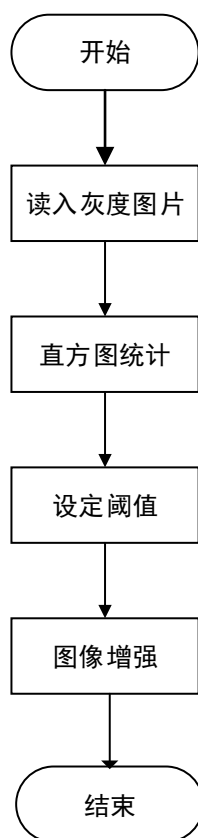


图 4-9 灰度图像增强实现代码

邻域处理算法一般就是对像素的邻域进行处理的方法,如均值滤波、拉普拉斯算子运算、平滑算子运算和卷积运算。点处理算法一般是指对图像像素进行逐一处理的方法。经过多种算法的尝试及验证,本课题设计中选用了空域处理算法,通过图像直方图进行统计,并求出图像的阈值,通过图像的阈值对图像进行增强处理。

本设计中使用的方案是通过直方图均衡化处理图像,改善图像因为灯光的不均匀特性造成的图像的模糊现象并进行了图像的修正,使图像对比度增强。但处理

的效果不明显。经过尝试寻找到了另一种图像像素分布统计算法，通过对灰度直方图的像素统计，求出每个像素对应的像素出现概率，寻找合适的阈值，在试验中我们取得是 15%，经过图像像素分布统计算法处理后获得增强的图像，图像像素分布统计具体的算法见附录。图像增强算法处理过程第一步是像素统计、第二步是统计像素出现的概率。第一步像素统计，实现流程如图 4-9 所示，算法如下

%对三维图像变换成灰度图像，经过像素统计过程求出每个像素出现的次数。

```
if length(size(picture)) == 3
    picture = rgb2gray(picture);
end
%%对灰度图像进行像素统计
N=zeros(1,256);%定义一维的零矩阵
[m,n] = size(picture);
%获取picture的行和列大小
for i=1:m
    for j=1:n
        a = picture(i,j);
        N(a+1) = N(a+1)+1;    %像素个数统计
    end
end
```

第二步统计像素出现的概率，算法如下

```
P = zeros(1,256);    %定义一个一行 256 列的一维矩阵
for i=1:length(N)
    P(i) = N(i)/sum(N);    %对1-256的像素出现的概率进行统计
end
DelImage = picture;    %对新的变量进行赋值
for k=1:255
    if P(k)>rate    % 像素出现的概率,可以取值为1.5%
        for i=1:m
            for j=1:n
                if k == DelImage(i,j)
                    DelImage(i,j) = DelImage(i,j) + addpix;
                end
            end
        end
    end
end
end
```

3. 图像增强前后的效果

本软件中使用的图像增强方式是空域的统计分析方法通过寻找合适的阈值对图像进行增强，图像增强前后的效果如图 4-10 所示。

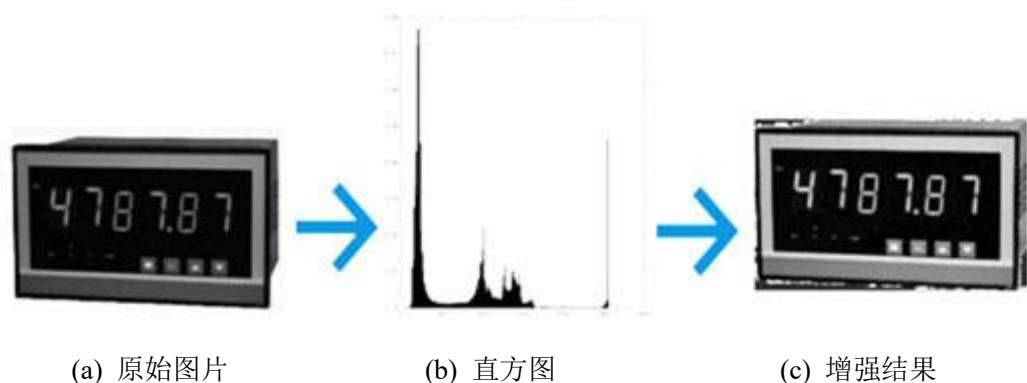


图 4-10 图像增强的前后效果

4.3.2 数字仪表图像滤波

常见的空域滤波器有均值滤波器、统计滤波器、和自适应滤波器。均值滤波器分为算数均值滤波器、几何均值滤波器、谐波均值滤波器和逆谐波均值滤波器。统计滤波器分为中值滤波器、最大或最小值滤波器和中点滤波器^[9]。

1. 图像进行滤波处理的意义

图像会因为成像设备、取像环境的不理想，导致成像的不理想，也就是常见的图像噪声污染，常见的图像噪声为电子噪声和光学噪声。

电子噪声是设备内部的阻性器件随着电子热运动而造成的，是一种简单的噪声。这类噪声能够在硬件上进行滤除，所以一般在电路设计中就通过硬件电路给滤除掉了。在图像中分布的该类噪声因其具有高斯函数形状的直方图分布以及平坦的功率谱，所以一般常用零均值高斯白噪声作为其噪声模型^[10]。

图像上的噪声一般表现为引起较强视觉效果的孤立像素点或像素块，具体的噪声一般与其研究的对象相关，是一种扰乱图像观测结果的不利于因素^[11]。对于数字图像的对象，噪声表现为极大或极小的像素点，当这些极值加在真实的图像上时对图像造成亮暗的噪声干扰，降低了图像的质量并影响图像的复原、分割和特征值提取等图像处理操作^[12]。所以在设计中图像的滤波处理也显得十分重要，它可用其 RMS 值（标准差）来完全表征。

光电子噪声是由于光的统计本质和光学图像传感器的光电转换过程引起的,在强弱不同的光线情况下,其对图像采集的影响是十分突出的,对于这种噪声常用具有泊松分布趋势更易描述的高斯分布,而标准差等于均值的平方根,这意味着噪声的幅度是与信号有关的^[13]。噪声的特点是叠加性和分布大小不规则性、噪声和图像之间的不规则性。图像的叠加性是指图像的串联传输系统中图像的噪声分布具有叠加性。噪声的分布和大小具有不规则性体现在噪声出现的位置和幅度是随机的。噪声和图像之间具有不规则性是由于摄像机拍摄时的颤动和其他原因会使得图像中的随机噪声变得不规整。常见的噪声有椒盐噪声、高斯噪声、泊松噪声等概率噪声等^[14]。

2. 图像滤波过程

高斯噪声的定义是如果一个噪声的幅度分布服从高斯分布,但是它的功率谱密度又是均匀分布的,那么我们称它为高斯白噪声^[15]。高斯白噪声是二阶矩不相关,一阶矩为常数的一种噪声,是指先后信号在时间上的相关性。其中热噪声和散粒噪声都属于高斯白噪声。

椒盐噪声是由图像传感器,传输信道,解码处理等产生的黑白相间的亮暗点噪声^[9]。椒盐噪声往往由图像切割引起,常用的椒盐噪声去除算法是中值滤波算法,椒盐噪声是指两种噪声,一种是盐噪声(salt noise),另一种是胡椒噪声(pepper noise)。盐=白色,椒=黑色。盐噪声是高灰度噪声,椒噪声属于低灰度噪声。一般两种噪声同时出现,呈现在图像上就是黑白杂点^[16]。

高斯噪声是指噪声服从高斯分布(即某个强度的噪声点个数越多,离这个强度越远噪声点个数越少),图像的高斯噪声是一种加性噪声即噪声直接加到原图像上,因此可以用线性滤波器滤除^[17]。椒盐噪声的特点是噪声分布在图像上类似在图像上撒上椒盐由此得名,这类噪声在图像上表现为白点和黑点的噪声分布,如电视里的雪花噪声等。椒盐噪声一般不用线性滤波器进行滤除,因为椒盐噪声一般被看成是一种逻辑噪声常采用中值滤波对噪声进行滤除。

中值滤波顾名思义是对所选区域求取中值,用中值代替其中心的像素值,也就是中值滤波窗矩阵中的中心像素的值,他是一种非线性的图像平滑滤波,对脉冲干

扰级的椒盐噪声有很好的抑制作用，在抑制噪声的同时能减少因为滤波造成的边缘模糊^[18]。

中值滤波可以过滤尖脉冲，滤波后的数据不会因滤波而发生显著的区别，其保留了原图像的变化趋势，也去除了尖峰脉冲对图像造成的影响，便于图像的进一步处理、分析。中值滤波的例子如下。

对于一维信号[100 90 80 40 70 94 93]，对其进行从小到大排序结果为[40 70 80 90 94 93 100]，对排序后的一维信号求得中间像素值为 90，也就是其灰度序列的中值像素为 90。用中值代替规定位置上的像素值也就是信号的中值滤波后的像素值。对于这个例子，进行中值滤波的结果是用中值 90 代替[100 90 80 40 70 94 93]中的信号序列中心位置的值，如果次序列中存在更大的脉冲噪声或更小的像素突变，就可以很方便的将其滤除，具体的中值滤波算法是对一幅图像的像素矩阵，取目标区域像素，以某大小窗口作为一个矩阵窗口，比如默认的 3×3 大小的窗口，或者别的 5×5 大小的窗口等，对窗口内的像素进行排序后取中间一个值作为目标像素的新的灰度值。在二维灰度图像上从左上角向右下角滑动，如此就完成对 $m \times n$ 行的二维灰度图像进行了中值滤波处理。

3. 图像滤波处理前后效果对比

对于椒盐噪声处理算法常见的有中值滤波算法、邻域平均法和维纳滤波等算法，根据以下比较发现，中值滤波算法在含椒盐噪声的图像中具有很好的滤波效果，同时使图像边缘尽可能的减少干扰损失。以下代码段为模拟椒盐噪声，向图像中加入椒盐噪声。

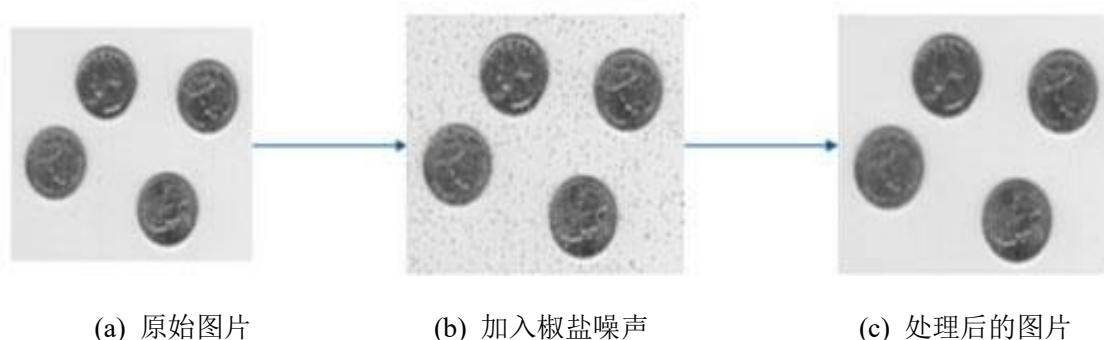


图 4-11 中值滤波前后效果

中值滤波处理作用前后的效果如图 4-11 所示。以 Matlab 自带的硬币图片为例，实现的流程图如图 4-12 所示，实现的代码如下

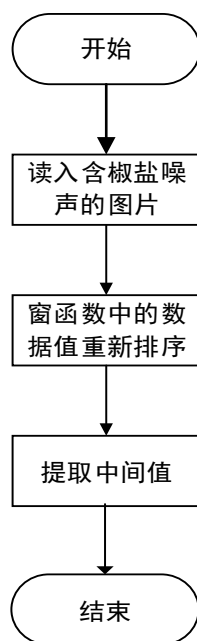


图 4-12 中值滤波实现流程

```

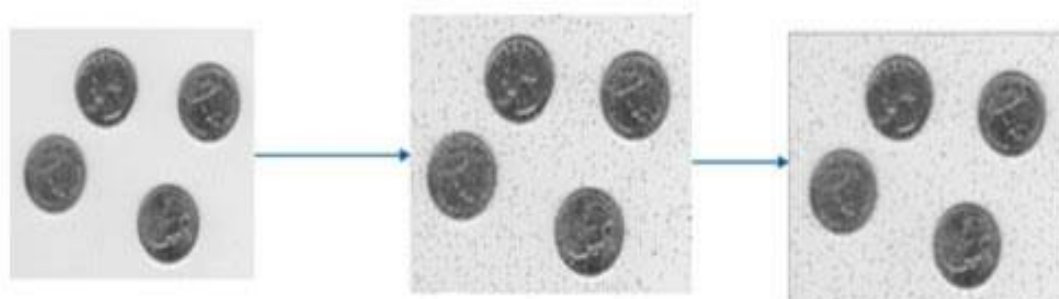
I = imread('eight.tif'); %读取硬币图片
J = imnoise(I,'salt & pepper',0.02); %加入椒盐噪声
  
```

具体的中值算法可以用冒泡排序法求得窗矩阵的中值，简单的中值滤波处理

代码如下

```

K=medfilt2(J,[3 3]); %中值滤波函数
figure, imshow(K)
figure, imshow(I)
figure, imshow(J)
  
```



(a) 原始图像

(b) 加入盐噪声

(c) 处理后的图片

图 4-13 邻域平均法滤波前后效果

邻域平均法滤波的作用效果如图 4-13 所示，代码实现如下。

```
K=medfilt2(J,[3 3]);
figure, imshow(K)
K1=filter2(fspecial('average',3),J);    %3 x 3 的模板平滑滤波
figure,imshow(uint8(K1))
```

维纳滤波：设计的维纳滤波器实质是求解维纳-霍夫方程的解，是寻求最小方差误差下的数字信号传递函数的表达式。假设输入的不含噪声信号为 $k(n)$ ，噪声信号 $x(n)$ ，其中 $f(t)=k(t)+x(t)$ ，则输入信号 $f(n)$ 是有 $f(t)$ 和取样脉冲函数 $h(t)$ 在时域上卷积而得到。

$$\text{即 } f(n) = f(t) \otimes f(t) \text{ 或者 } f(n) = \int_0^{+\infty} h(\tau) \bullet f(n-\tau) d\tau \quad (4-1)$$

所以求的噪声的误差如式 4-2

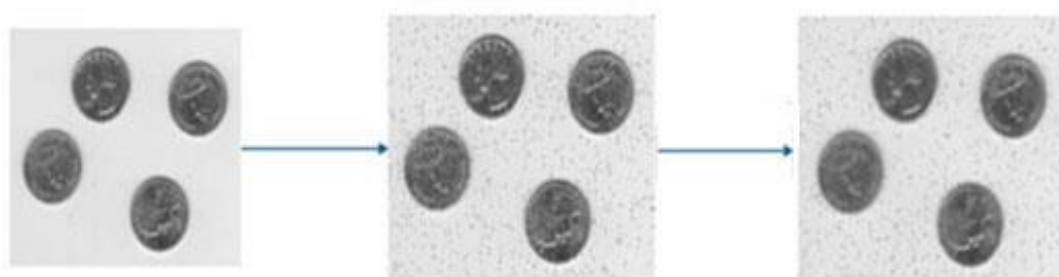
$$\Delta w(n) = f(n) - x(n) \quad (4-2)$$

所以方差为

$$\overline{\Delta w_{(n)}^2} = f(n) - x(n) \quad (4-3)$$

利用数学方法求最小均方误差，求得线性滤波器的冲击响应 $hopt(\tau)$ 可得方程

$$hopt(\tau) = R_{Xy}(\tau) - \int_0^{\infty} R_{xy}(\tau - \sigma) h_{xy}(\sigma) d\sigma; \tau \geq 0 \quad (4-4)$$



(a) 原始图像

(b) 加入椒盐噪声

(c) 处理后的图片

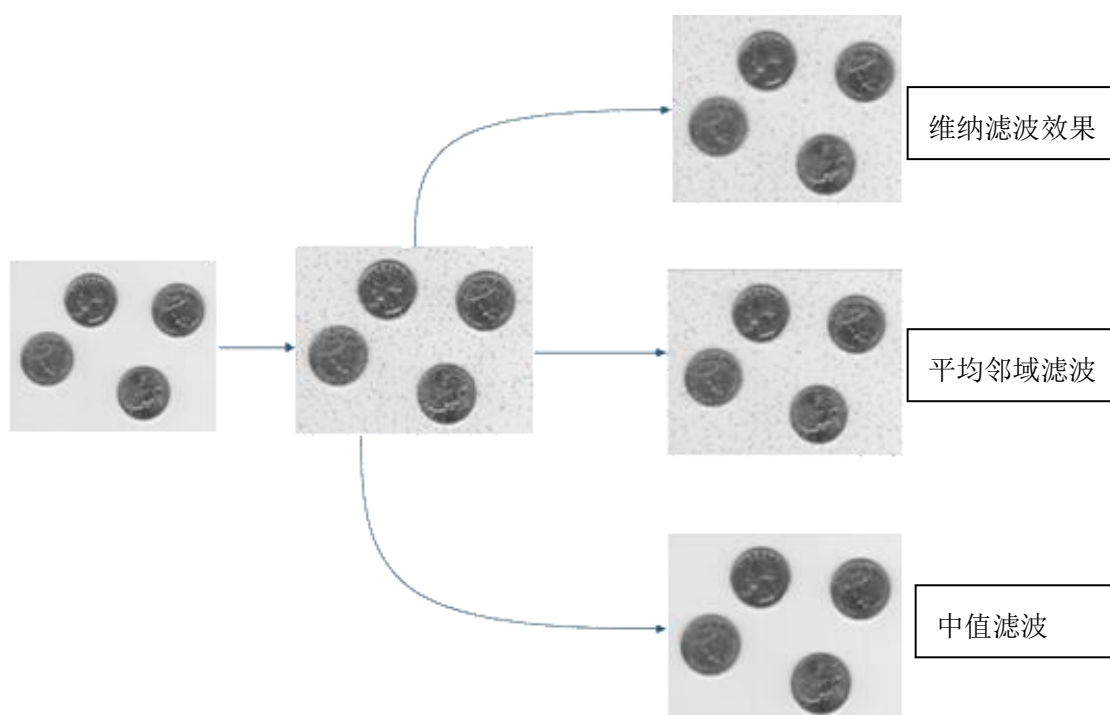
图 4-14 维纳滤波器作用的前后效果

求解维纳-滤波方程可以得到最佳滤波器的冲击响应 $hopt(t)$ 作用效果如图 4-14 所示。维纳滤波处理代码如下。

```
K2=wiener2(J,[5 5]);
figure,imshow(uint8(K2))
```

综上实验结果比较显然中值滤波在处理椒盐噪声中具有较好的处理效果，如图 4-15 所示，所以高斯噪声可以用中值滤波或者维纳滤波算法滤除，效果图如图 4-16 所示，高斯噪声滤除的实现代码如下。

```
I = imread('eight.tif');
G=imnoise(I,'gaussian',0.01);%添加均值为 0 方差为 0.01 的高斯噪声
K1=medfilt2(G,[6 6]); %中值滤波
figure,imshow(K1);
K2=wiener2(G,[6 6]); %维纳滤波
figure,imshow(K2)
figure,imshow(K1)
```



(a) 原始图像

(b) 加入椒盐噪声

(c) 处理效果

图 4-15 三种滤波效果对比图

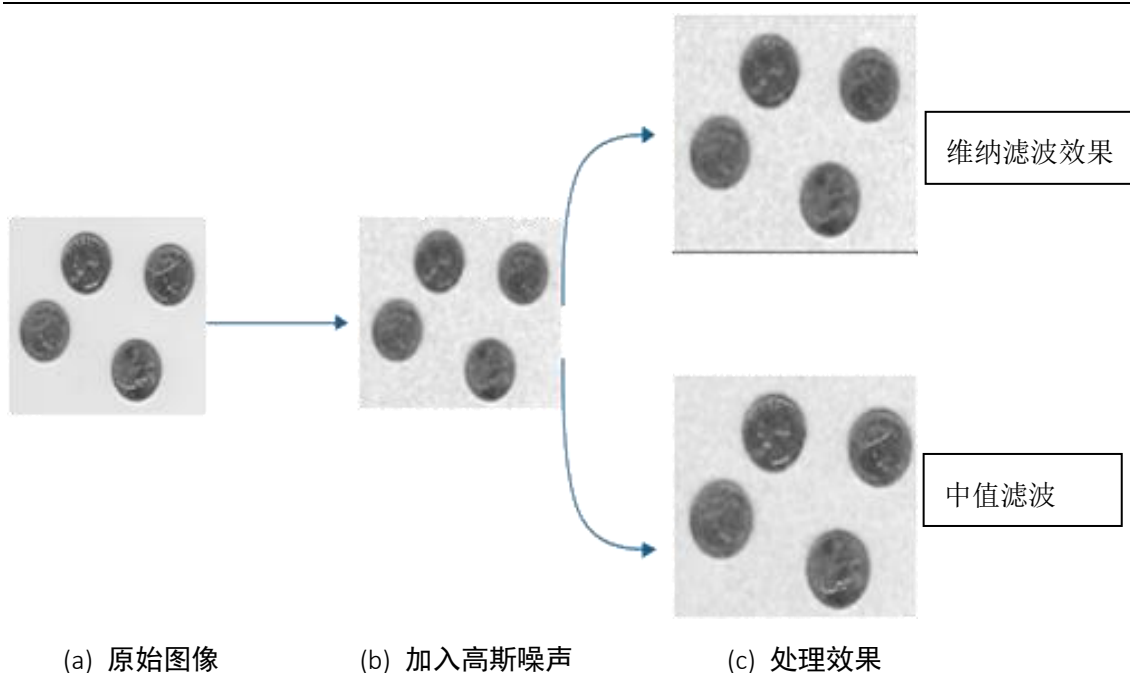


图 4-16 高斯噪声的两种滤波效果图

4.3.3 数字仪表图像感兴趣区域的定位

对图像的感兴趣区域定位的前后定位效果如图 4-17 所示。

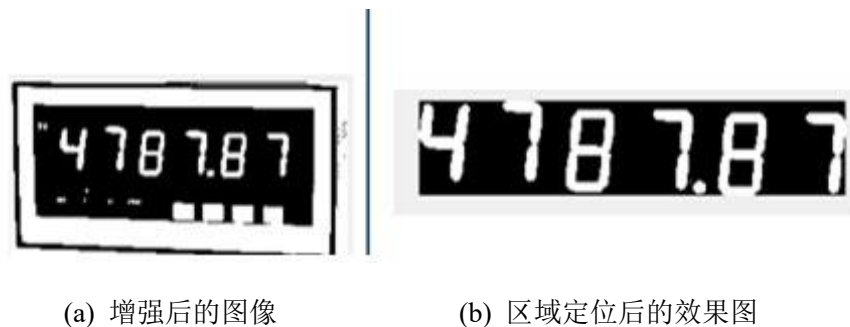


图 4-17 感兴趣区域定位

1. 为什么要进行区域定位

对待识别体进行区域定位是感兴趣区域提取能否成功的关键，良好的区域定位算法能帮助实现区域的准确提取，进一步减少图像处理的运算量和更大程度上的减少图像背景噪声的干扰。

2. 特征区域定位的步骤

本课题所研究的数字仪表读数自动识别系统在确定最终的定位方案前提出了三种感兴趣区域定位的实现方案，并且三种实现方案都得到了实验的验证，确定了

方案的可行性，但是通过对方案的评比，最终选择的是第三种定位方案。

第一种方法是通过人机交互的方式，指定图像上的感兴趣区域。用户通过鼠标选取界面元素图像区域后会生成一个与原图像大小相同的二值图像。指定图像的感兴趣区域是通过 Matlab 自带的函数 `roipoly` 函数实现，或者 `roicolor` 函数对 RGB 或灰度图像的感兴趣区域进行区域的指定。该方案具有处理简单，环境适应性强，不足之处在于，对于每次摄像头的位置改变都需要人为的指定感兴趣区域，对自动化设备的智能化大打折扣，所以该方法不具有广泛的应用价值。

第二种方法是通过预先设定的感兴趣区域位置及大小，与第一种方法不同之处在于，第一种方法是人机交互，第二种方法是内部固化设置了的。通过预先设置的区域大小，通过调节摄像头让数显数码管刚好并且充分的进入图像预定显示区域，这样可以省略掉人机交互的环节，但是却增加了摄像头调节环节，这种方案的优点是，具有低的算法处理复杂度，但是每次移动摄像头后都需要耗费大量的时间对一开始的摄像机对数显部分的捕捉时间，不具有良好的智能设备交互特性。因此该方法不具有广泛的适应价值为此我提出了第三种使用法案。

第三种方案首先针对市场上大多数的数显方式，大多选择的是红色、蓝色、或者黄色的数显数码管。如图 4-18 所示。常见的数码管分别为红色、蓝色和黄色三种。



(a) 红色的数显方式

(b) 蓝色的数显方式

(c) 黄色的数显方式

图 4-18 常见的数码管颜色

所以第三种算法实现方案是基于颜色模型的感兴趣区域定位，首先我们针对市面上使用最广的红色数码管进行感兴趣区域定位。

我们选取图像 RGB 颜色模型的红色通道，对红色通道进行定位，这样在很大程度上能减少复杂的自动定位处理。定位原理是根据三基色的原理，用基色光单位来表示光的量^[11]。在 RGB 颜色空间对任意色光 F 都可以用 R 、 G 、 B 三色不同分量的相加混合而成， $F=r[R]+g[G]+b[B]$ ，如图 4-120 所示，RGB 的另一种描述方

式如图 4-17 所示。

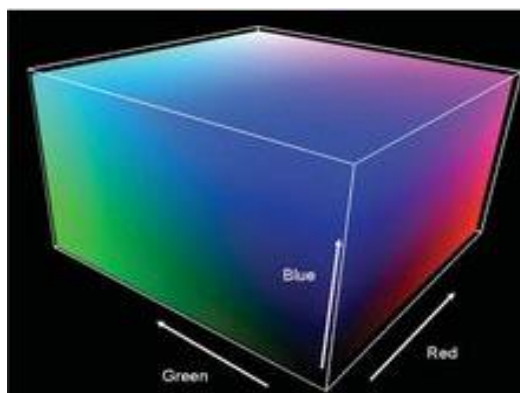
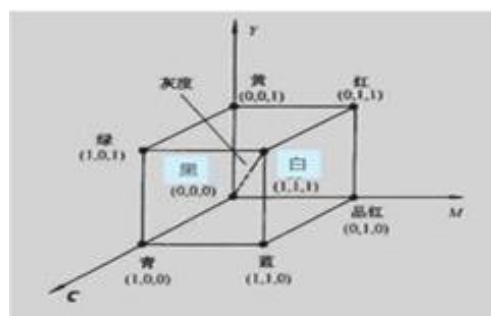


图 4-19 RGB 模型



(注) 彩色图像黑色(0,0,0), 白色(1,1,1)

图 4-20 三基色模型

如图 4-21 的拾色器, 不同的颜色其 RGB 颜色分量是不同的, 当选择颜色为白色时, 我们可以看到 R 分量为 256, G 分量为 256, B 分量为 256。

如图 4-22 所示, 当选择的颜色为灰黑色时, 能从拾色器上看出 R 分量颜色的值为 0, G 分量的颜色的值为 0, B 分量的颜色值为 0。再如上图选中的区域, 当 $R = 138, G = 10, B = 0$ 时表示的是当前的颜色值。

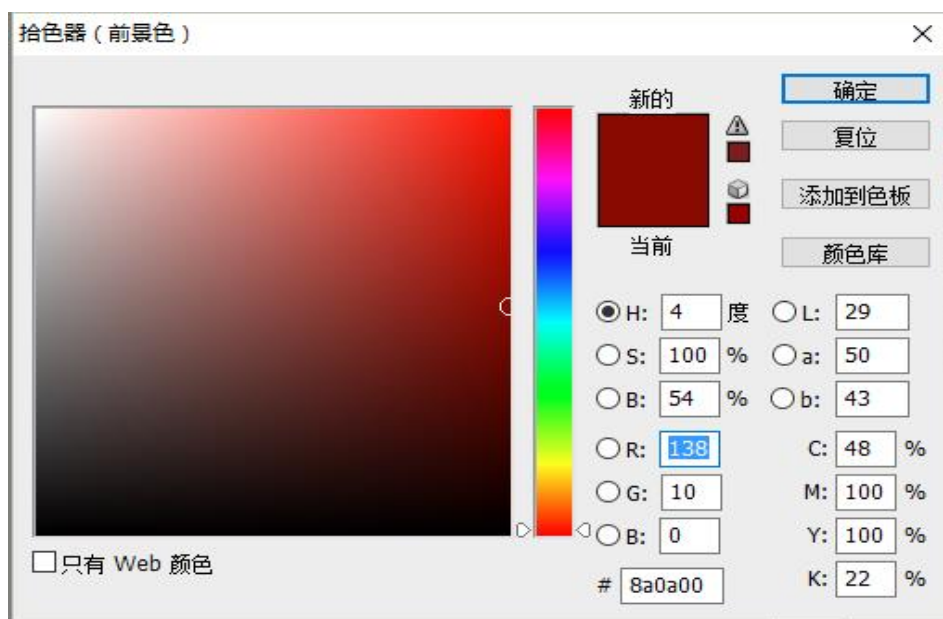


图 4-21 拾色器



图 4-22 仪表数字的颜色值提取

对于本设计中的数码管，我们可以清楚的发现当拾色吸管取数字上的颜色值时红色分量十分明显，而蓝色和绿色分量则相对较弱，经实验得到，我们在进行感兴趣区域定位时可以直接选择红色通道的颜色作为定位的坐标，当然在利用红色区域定位的同时我们还要对所选区域进行排查，排除环境的干扰区域，方法也较为简单，应为红色区域外部是由褐色的屏幕背景作为分割的，与其他环境的背景还是有很大的差别的。

在利用 R 通道的颜色模型进行区域定位时我们需要对红色通道进行阈值的选取，经大量的实验测试得到当各通道的值选取为：

$R_{min}=180; R_{max}=256; G_{min}=10; G_{max}=100; B_{min}=40; B_{max}=200;$

能很好的定位到感兴趣区域，再通过区域定位函数：`pic=rgb2roicollect (z,Rmin , Rmax , Gmin , Gmax , Bmin , Bmax)`;就可以将感兴趣的区域在原始的图像中分割出来。`rgb2roicollect` 算法的思路比较简单，实现的流程图如图 4-23 所示，就是单纯的先统计再判断。具体实现代码如下。

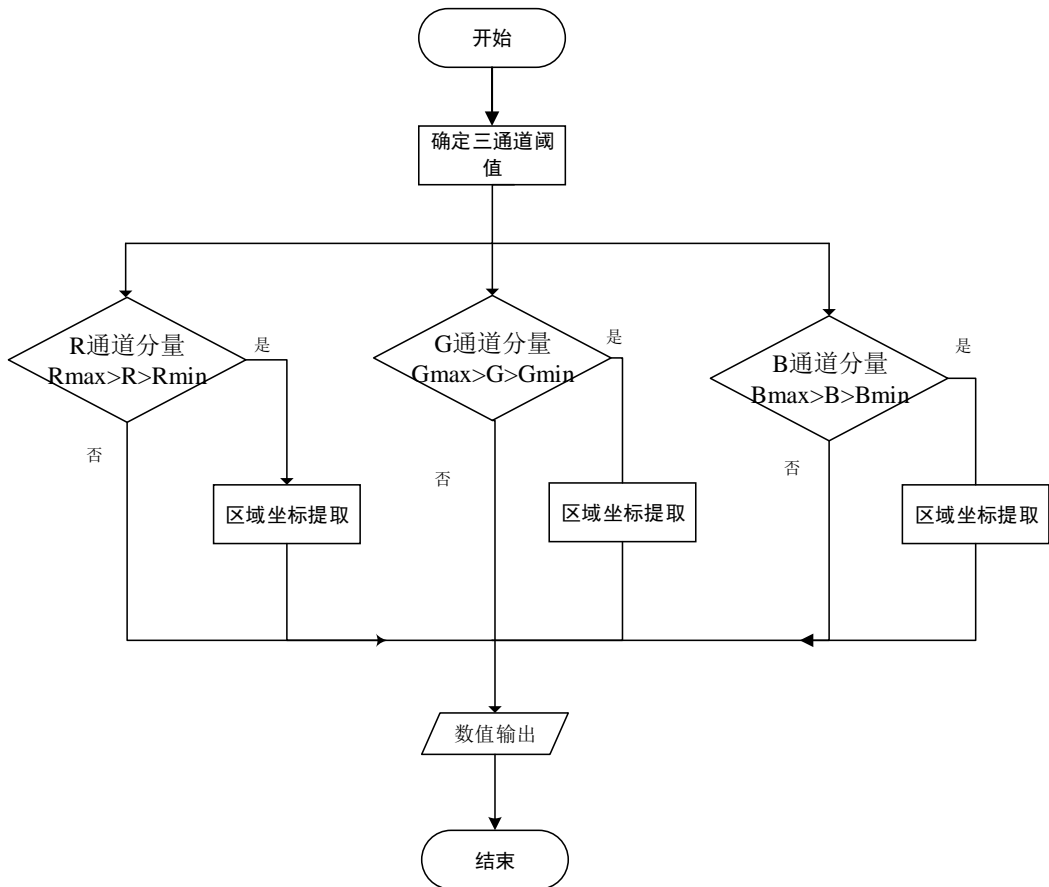


图 4-23 感兴趣区域定位的实现流程

```

%% 首先进行每行的像素统计%% 每行像素统计
Blue_y = zeros(y,1); %建立 0 矩阵
for i=1:y
    for j=1:x
        if((myI(i,j,1) <= Rmax) && (myI(i,j,1) >= Rmin) && ((myI(i,j,2) <= Gmax)
        && (myI(i,j,2) >= Gmin)) && ((myI(i,j,3) <= Bmax) && (myI(i,j,3) >= Bmin)))
            Blue_y(i,1) = Blue_y(i,1) + 1; %计算每一行的蓝色像素点
        end
    end
end
[temp MaxY] = max(Blue_y); %temp = Blue_y 最大值, MaxY 最大值所在位置列
%% x\y 映射求取扫描上下左右边界位置
PY1 = MaxY;
while ((Blue_y(PY1,1) >= 5) && (PY1 > 1)) %左边
    PY1 = PY1-1;
end
PY2 = MaxY;

```

```

while ((Blue_y(PY2,1) >= 5 ) && (PY2 < y)) %右边
    PY2 = PY2 + 1;
end
IY = RGBpicture(PY1:PY2, :, :); %提取左右两边的数字号行的区域
%% 每列像素统计
Blue_x = zeros(1,x);
for j = 1:x
    for i = PY1:PY2
        if((myI(i,j,1) <= Rmax) && (myI(i,j,1) >= Rmin) && ((myI(i,j,2) <=
Gmax) && (myI(i,j,2) >= Gmin)) && ((myI(i,j,3) <= Bmax) && (myI(i,j,3) >=
Bmin)))
            Blue_x(1,j) = Blue_x(1,j) + 1;
        end
    end
end
%% 寻找上下边界
PX1 = 1;
while ((Blue_x(1,PX1) < 3) && (PX1 < x)) %从上往下扫描找上边界
    PX1 = PX1 + 1;
end
PX2 = x;
while ((Blue_x(1,PX2) < 3) && (PX2 > PX1))%从下往上扫描找下边界
    PX2 = PX2 - 1;
end
PX1 = PX1 - 2;
if PX1 < 1
    PX1 = 1;
end
PX2 = PX2 + 2;
if PX2 > x
    PX2 = x;
end
Plate = RGBpicture(PY1:PY2,PX1:PX2,:);

```

4.3.4 数字仪表图像感兴趣区域的提取

随着计算机图像技术的快速发展，图像处理技术已被广泛应用于众多领域，比如指纹识别技术、车牌识别技术、文字识别技术、纸币识别技术等。

在整个图像处理过程中，图像分割技术是图像识别，处理的基础，也是关键^[19]。图像分割的作用是将反应图像真是情况的有用区域，与识别的干扰区域分割

开来。好的分割应具备以下特性^[20]

1. 分割出来的有用区域，对某种特质而言具有相似性。
2. 相邻区域对分割所依据的性质有明显的差异。
3. 区域边界是明确的。

图像感兴趣区域分割出来的目的是把图像空间按要求分割成不同的感兴趣块，使感兴趣区域凸显出来，例如文字识别中的个别文字分割，识别车牌号中的车牌与背景的分割，车牌号和车牌背景的分割等^[11]。

图像不同区域之间像素存在着不连续性，而相同区间的像素却存在着相似性，根据不同灰度，不同的颜色，不同的像素亮度、梯度等不连续性，对图像区域进行分割时一般较为常用的方法有区域生长法、四叉树分解的分割算法、平均灰度的分割算法、基于统计特性的区域分割算法。

区域生长法是一种古老的算法，区域生长图像分割算法提出时间比较早，最早是 Levine 等人提出的，为一种串行的分割算法。四叉树分割算法执行的效率比基于区域生长算法的执行效率要高，更能凸显图像的基础细节。在分割结束能够获得目标的大小，边缘和个数信息。

本设计中使用的是阈值分割方式，具有算法简单，效率高的特点，我们直接通过阈值的方式，对个通道像素所在感兴趣区域进行检索，求出感兴趣区域的左上、左下、右上、右下的各个坐标，最后根据坐标位置将感兴趣区域提取出来。

1. 感兴趣区域图像的处理

对提取出来的感兴趣区域，我们一般先将其二值化，减少后面图像识别的计算量，我们对感兴趣区域灰度变换后，通过灰度直方图统计图像的像素分布情况，计算每个像素出现的概率。

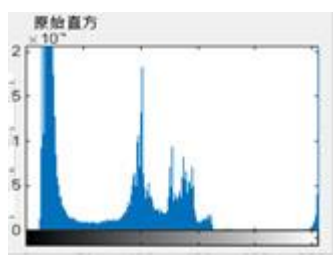


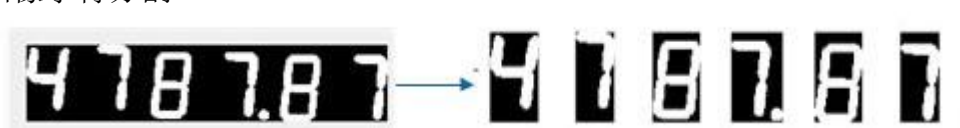
图 4-24 图像直方图统计

通过与出现的概率情况进行自适应的二值化处理。RGB 到灰度图像的变换通过 Matlab 自带的 `rgb2gray` 函数转换得到。得到的图像经过直方图统计如图 4-24 所示。二值化后的图像如图 4-25 所示。



图 4-25 分割图像二值化

2. 图像字符分割



(a) 感兴趣区域图像

(b) 数字分割后的图像

图 4-26 字符的分割

图像的分割常用方法有基于边界的分割方法和自适应阈值分割方法。基于阈值的分割效果如图 4-26 所示。

基于边界的分割方法就是假设图像区域的边界存在，基于区域的分割方法，假设相同子区域具有相同或相近的属性，而不同区域的像素之间属性差别比较大。

图像分割的基本方法一般遵循以下原则。

1. 从简到难逐级分割。从图像的整体出发，分离主要目标，然后再考虑图像的细节。

2. 控制图像的背景环境，降低场景的复杂程度。

3. 把焦点放在感兴趣的区域上，减少不需要的图像区域背景干扰。

自适应阈值分割就是将图像分割成不同的块对每块分割后的的子图像分别计算其局部阈值，再利用形态学原理，对图像边缘进行提取。形态学边缘提取的具体方式，假设原始图像为 A ，数学形态结构元素为 B 。在 B 结构元素下进行腐蚀，得到 C 的结果。用原图像 A 减去 C ，得到边缘 D 的结果。

根据对多种字符分隔的不同算法验证中得出结论分利用统计学方法进行字符

的分割具有算法简单, 运算量少等特点, 于是将该算法引入了字符分割, 通过实验得到了良好的效果。效果如图 4-22 所示。分割的过程首先是对带分割的图像建立图像的数学模型再对图像的特点进行分析, 由感兴趣区域的特点容易看出, 每个字符之间的间隙比较大, 远远大于自身存在的不连续点的间隙, 所以我们只需要确认每个字符间的间隙大于 $1/4$ 字符所占的宽度。于是我们首先从左到右依次统计图像中不连续的列像素点位置, 最后根据统计的列像素位置将图像中的每个字符独立的分割出来。具体算法如下:

```
%% 字符识别算法
%从含众多的字符图中分割出一个个的字符
[~,n]=size(image1);
Ymatrix = zeros(1,n);
for I = 1:n
    Ymatrix(i) = sum(image1(:,i));    %求每列的和
end
lenYmatrix = length(Ymatrix);
Yflag = zeros(1,lenYmatrix);
for i = 1:lenYmatrix-2    %对左端的字符边界采样
    if(Ymatrix(i) > 3) && (Ymatrix(i+1) > 3) && (Ymatrix(i + 2) > 3)
        Yflag(i)=1; %保存右边第一个字符和第二个字符的分界位置位于第
        几列
    end
end
% plot(1:length(Yflag) , Yflag);
```

最后利用前面统计到的每个字符边界的坐标进行图像的裁剪。裁剪的过程就是赋值的过程, 具体的程序代码见附录。

4.4 数字仪表图像识别处理

在图像的识别过程中我们需要先对图像进行边缘检测, 以便于寻找图像中字符与垂直方向的旋转偏差角度, 以及便于寻找图像关于 x 方向或 y 方向的错切角度。对图片修正后, 我们再引入 8 段式数码管数显原理, 对数码管的每个段位进行检测, 通过八段数码管显示的编码, 识别不同的数值。

因此要实现图像特征值的识别, 需要先进行边缘检测、求出旋转角度偏差, 其次再求出错切角度的偏差, 最后通过八段式数显模型求出每个图像的数字。

4.4.1 图像的边缘检测

为了将感兴趣区域图像与图像背景相区分开来,以便更容易地对图像旋转变换的角度进行调整,以及错切变换的调整,需要得到图像的边缘。图像边缘检测就是检测图像的边缘的点,然后根据得到的边缘点连成线,最后连成的线就可以将感兴趣区域图像与图像背景区分开来,所以图像处理中的边缘检测相当重要。边缘检测是根据图像灰度变换显著区域点的检测,因此一般利用像素各点求解一阶或者二阶等多阶导数,从而得到边界点上的像素突变点^[14]。由边界灰度变化的不同情况可以将边缘突变类型分为阶梯状的边缘突变、脉冲状的边缘突变屋顶状的边缘突变三种。对于阶梯状边缘突变,图像的边缘点对应一阶导数的极大值或极小值点,对应二阶导数的拐点位置^[21]。

在数字图像处理中,各图像的像素点均为离散的像素点,所以一般使用的是序列的差分方程来替代连续信号的微积分运算^[15]。图像的一阶导数具有明显的方向性,适合对特定边缘的图像进行检测,不具有普遍的应用价值。在一阶常微分算法的求解中,我们一般定义图像的梯度为梯度算子,也是常用的一种运算方式。

一般图像在灰度变化率最大的方向就是图像的梯度,他能很好地反映图像在边缘处的变化情况。边缘检测算子是利用图像边缘的突变性质来检测边缘的,主要分为两种类型,一种是以一阶导数为基础的边缘检测算子,通过计算图像的梯度值来检测图像边缘,常用的一阶导数模板有 Roberts、Prewitt 和 Sobel 算子^[16]。一种是以二阶导数为基础的边缘检测算子,通过寻求二阶导数中的过零点来检测边缘,常见的二阶导数模板有 Laplacian 算子、LOG 算子、Canny 算子^[22]。

图像处理中导数常用于检测图像边缘像素值的不连续性,一阶导数在图像中产生较厚的边缘,二阶导数具有反应图像细节的功能。例如:对于细的边缘,孤立的点、以及噪声都很敏感。二阶导数在灰度过度区域会产生双边,利用二阶导数的符号可以确定灰度的过渡是从亮到暗还是从暗到亮^[23]。

为了得到满意的边沿检测结果,人们常用几种不同的算子将检测出来的边缘图片进行叠加。边缘检测中的不连续性,在进行边缘检测时,有时检测出的却是

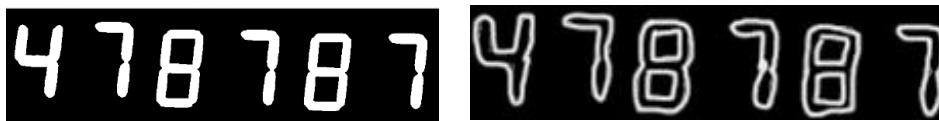
孤立点,有时检测出的是噪声,有时是有用的边缘,所以我们有必要将离散的边缘点进行连接。为了得到图相闭合的边缘线轮廓,去除噪声点,常见的连接方法包括局部处理及 Hough 变换等方法^[24]。基于 Roberts 算子、Sobel 算子和 Prewitt 算子的边缘检测算法比较如下。

Roberts 算子是一阶微分算子 Roberts 利用的是局部差分算子,他的一般表示式子如下。

$$g(x, y) = \left[\sqrt{f(x, y) - f(x+1, y+1)} \right] 2 + \left[\sqrt{f(x+1, y) - f(x, y+1)} \right] 2 \quad (4-5)$$

其中 $f(x, y)$ 、 $f(x+1, y)$ 、 $f(x, y+1)$ 和 $f(x+1, y+1)$ 分别为矩阵 4 个领域的坐标,且是具有整数像素坐标的输入图像;其中的平方根运算使得该处理类似于人类视觉系统中发生的过程^[25]。

Roberts 算子都是模板为 2x2 的矩阵 $\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$ 或者 $\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$, 图像中的每一个点都用他们中的一个进行卷积。通过 Roberts 算子前后效果如图 4-27 所示



(a) 二值化后的图像

(b) 边缘识别后的效果

图 4-27 Roberts 算子作用前后效果

实现代码如下。

```
I=imread('myGray.png'); %读取灰度图像如上图所示。
BW1=edge(I,'roberts');
[BW1,thresh1]=edge(I,'roberts');
figure,imshow(I); %显示原始图像。
figure;imshow(BW1); %显示 Roberts 算子处理的图像如上图所示。
```

Sobel 算子是一种一阶的微分算子,是一个 3x3 的矩阵,通过与图像之间 x 方向和 y 方向的卷积,分别得出各自方向上的亮度近似差分值,例如原始图像为

I, Sobel 卷积算子为 $G_x = \begin{pmatrix} -1 & 0 & +1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$ 以及 $G_y = \begin{pmatrix} +1 & 2 & +1 \\ -2 & 0 & 2 \\ -1 & -2 & -1 \end{pmatrix}$ 分别为横向和纵向灰度卷积算子。

$$G_x = (-1)*f(x-1, y-1) + 1*f(x+1, y-1) + (-2)*f(x-1, y) + 2*f(x+1, y) + (-1)*f(x-1, y+1) + 1*f(x+1, y+1) \quad (4-6)$$

$$G_y = [f(x-1, y-1) + 2f(x, y-1) + f(x+1, y-1)] - [f(x-1, y+1) + 2*f(x, y+1) + f(x+1, y+1)] \quad (4-7)$$

一般取 $|G| = |G_x| + |G_y|$ 如果 G 大于某一阈值则认为是检测到边缘。

Sobel 算子是根据图像的像素点上下、左右相邻点灰度的加权差，在边缘处达到极值这一现象检测边缘^[19]。对噪声具有平滑作用，提供较为精确的边缘方向信息，边缘定位精度不够高。当对精度要求不是很高时，是一种较为常用的边缘检测方法。

Prewitt 算子全称是 Prewitt Operator，Prewitt 提供两个方向的检测分别是水平方向和垂直方向的边缘检测。

$$E_x = \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix}$$

在垂直方向的算子矩阵为：

$$\text{对水平方向的边缘检测算子模板为: } E_y = \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$

本课题的数字仪表读数自动识别系统中图像的边缘检测使用的是Sobel算子的垂直边缘的检测，`BW = edge(image1,'sobel','vertical');`对图像的垂直边缘的检测，能为后面图像的旋转和错切角度的计算减少工作量。

4.4.2 ROI 区域旋转校正

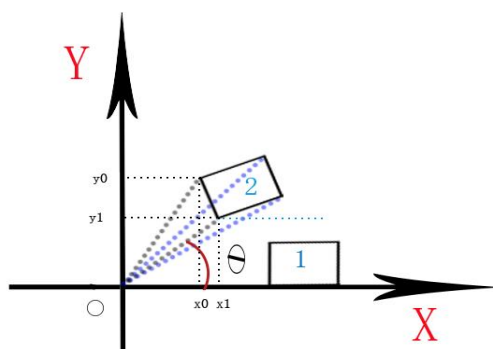
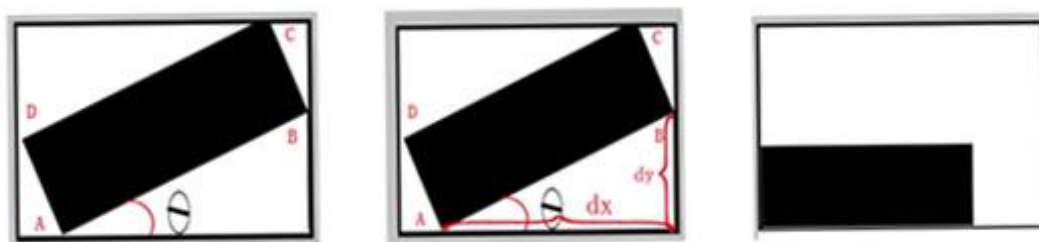


图 4-28 图像旋转变换

对图像感兴趣区域进行旋转校正的前后效果如图 4-28 所示，旋转变换，图像由理想状态的 1 位置旋转变换为 2 的位置，其旋转角度为 Θ ，其旋转变换角度能通过三角形边与角的关系容易地求得。

设旋转前图像的四个顶点坐标分别为 $A_0(x_0, y_0)$; $B_0(x_1, y_1)$; $C_0(x_2, y_2)$; $D_0(x_3, y_3)$; 旋转后图像的四个顶点坐标分别为: $A_1(x_0', y_0')$; $B_1(x_1', y_1')$; $C_1(x_2', y_2')$; $D_1(x_3', y_3')$; 欲求 Θ ，先求 dx 、 dy ，由图 4-29 可以求得 $dx = x_1' - x_0'$; $dy = y_1' - y_0'$; $\Theta = \arctan(dy/dx)$; 由此求得旋转角度 Θ 。



(a) 旋转修复前的图像

(b) 根据算法修复后的图像

图 4-29 旋转变换前后的效果图

利用 Matlab 中的旋转变换函数: `imrotate()`; 将图像旋转 $-\Theta$ 角度大小，则图像旋转变化矫正，流程图如图 4-30 所示，旋转的角度计算具体实现程序见附录。

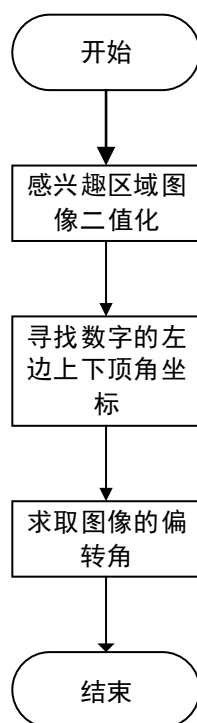


图 4-30 旋转角度求取程序流程图

4.4.3 ROI 区域图片错切矫正

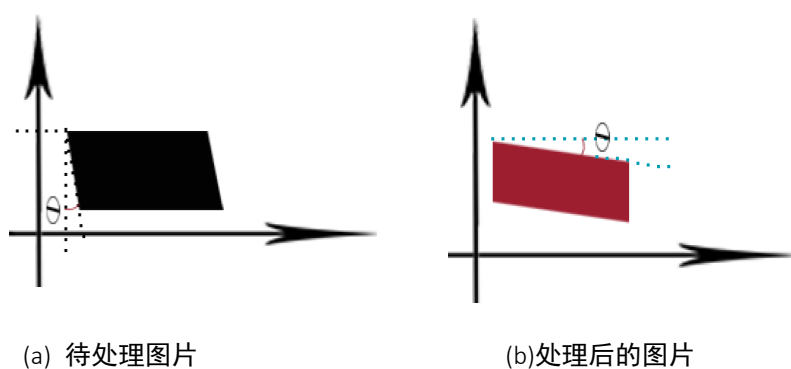


图 4-31 图像的错切变换

常见的两种错切变换如图 4-31 所示，图 4-26(a)是垂直错切变换，如图 4-31(b)是垂直错切变换，由坐标关系求得 Θ 角度， Θ 也是利用三角形边角关系求得，再利用 Matlab 的自带函数，`imtransform()` 函数的仿射变换得到理想的图片，具体实现程序见附录。

4.4.4 单个字符图像建模

字符识别的第一种实现方式，首先我们对每个分割后的字符建立数学模型，

再次对图像上下左右的非有用的区域进行裁剪，再对图像进行处理。在八段式的数码管显示中，将数码管分成六个区间如图 4-32 所示，分别命名为一区、二区、三区、四区、五区、六区。

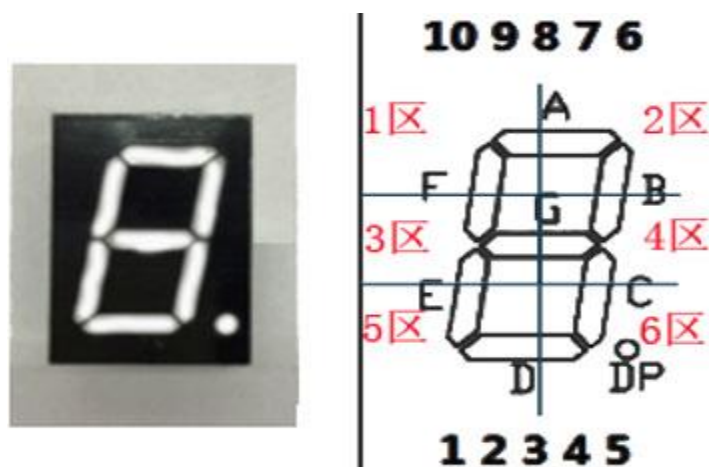


图 4-32 数码管的空间模型

对每个区间分别进行水平方向和垂直方向的像素点进行检测，比如第一个区间如果数显结果为数字 8，理论上是 A、B、C、D、E、F、G 这 7 个段位都为非 0 像素值存在区域^[26]。那么在第一个区间水平和垂直方向我们都能检测到非零像素的存在。

水平像素统计，最后通过 `smg8prow(K)` 函数来统计，如果水平方向存在非零像素那么我们就将第 `K` 个区间 (`k` 属于 1-6 号区间的一个) 其置为 1 否则对 `smg8prow(K)` 的第 `k` 个区间标志位置为 0。实现的流程图如图，代码如下。

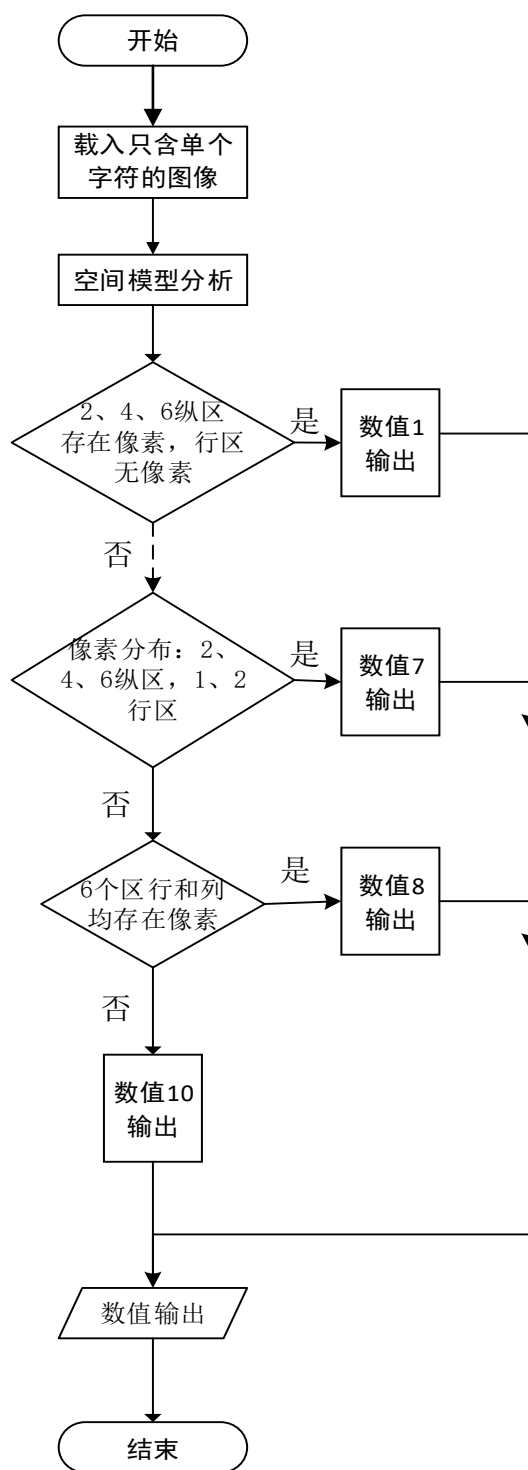


图 4-33 数字识别

```

for row=3:m
    flagrow(K,row)=sum(pictone(row,:,K)); %对行求和
    %480x640
  
```

```

switch choseone
case 1
    if flagrow(K,row)>24 && flagrow(K,row-1)>24 &&
        flagrow(K,row-2)>24    %阈值需要在试验中进行调整
        smg8prow(K)=1;
    end
case 2
    if flagrow(K,row)>20 && flagrow(K,row-1)>20 &&
        flagrow(K,row-2)>20    %阈值需要在试验中进行调整
        smg8prow(K)=1;
    end
case 3
    if flagrow(K,row)>7 && flagrow(K,row-1)>7 &&
        flagrow(K,row-2)>7    %阈值需要在试验中进行调整
        smg8prow(K)=1;
    end
otherwise
    disp('other value');
end
end

```

垂直方向的像素统计算法如下，同样的统计方式只是下面是进行垂直方向的像素统计，只是将统计的结果存入标志变量 smg8pcolum(K)中，部分的实现代码如下

```

flagcolum(K,1) = sum(pictone(:,1,K)); %对列求和
flagcolum(K,2) = sum(pictone(:,2,K)); %对列求和
for colum = 3:n
    flagcolum(K,colum) = sum(pictone(:,colum,K)); %对列求和
    switch choseone
    case 1          %1024x
        if flagcolum(K,colum) > 24 && flagcolum(K,colum-1) >
            24 && flagcolum(K,colum-2) > 24
            %阈值需要在试验中进行调整
            smg8pcolum(K)=1;
        end
    case 2          %512x
        if flagcolum(K,colum) > 20 && flagcolum(K,colum-1) >
            20 && flagcolum(K,colum-2) > 20
            %阈值需要在试验中进行调整
            smg8pcolum(K) = 1;
        end
    end
end

```

```

end
case 3      %480
    if flagcolumn(K,column) > 7  &&  flagcolumn(K,column-1) > 7
        &&  flagcolumn(K,column-2) > 7
        %阈值需要在试验中进行调整
        smg8pcolumn(K) = 1;
    end
otherwise
    disp('other value');
end
end
end

```

将一个完整的字符图像分成六个子对象部分的代码如下

```

pictone=zeros(row,column,6);
pictone(:,1)=picture(1:row,1:column);      %1
pictone(:,2)=picture(1:row,column+1:n);      %2
pictone(:,3)=picture(row+1:2*row,1:column);  %3
pictone(:,4)=picture(row+1:2*row,column+1:n); %4
pictone(:,5)=picture(2*row+1:m,1:column);    %5
pictone(:,6)=picture(2*row+1:m,column+1:n);  %6

```

4.4.5 数字仪表的识别结果的输出

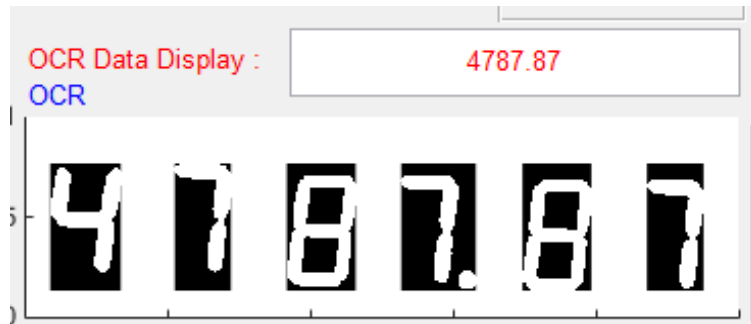


图 4-34 识别结果图

图像识别的效果图如图 4-34 所示，实现其效果的部分识别代码如下

```

%根据分割出的每个字符
%识别出对应字符的数字
% smg8pcolumn 为自己编写的函数体
a = smg8prow(1) + smg8prow(2)-1; %如果 a 段显示则 a=2 其他亦如此
b = smg8pcolumn(2) + smg8pcolumn(4) - 1;
c = smg8pcolumn(4) + smg8pcolumn(6) - 1;
d = smg8prow(5) + smg8prow(6) - 1;
e = smg8pcolumn(3) + smg8pcolumn(5) - 1;
f = smg8pcolumn(1) + smg8pcolumn(3) - 1;

```



```

g = smg8prow(3) + smg8prow(4) - 1;
numeric = [a b c d e f g];
for i = 1:7
    if numeric(i) < 0
        numeric(i) = 0;
    end
end
numeric0 = [1 1 1 1 1 1 0];numeric1 = [0 1 1 0 0 0 0];numeric2 = [1 0 0 1 1 0 1];
numeric3 = [1 1 1 1 0 0 1];numeric4 = [0 1 1 0 0 1 1];numeric5 = [1 0 1 1 0 1 1];
numeric6 = [1 0 1 1 1 1 1];numeric7 = [1 1 1 0 0 0 0];numeric8 = [1 1 1 1 1 1 1];
numeric9 = [1 1 1 0 0 1 1];
if (sum(numeric0 == numeric)) == 7
    num = 0; %求与 numeric 相等的位数，全相等则为 7（7 位都相等）
elseif (sum(numeric1 == numeric)) == 7
    num=1;
elseif (sum(numeric2 == numeric)) == 7
    num=2;
elseif (sum(numeric3 == numeric)) == 7
    num=3;
elseif (sum(numeric4 == numeric)) == 7
    num=4;
elseif (sum(numeric5 == numeric)) == 7
    num=5;
elseif (sum(numeric6 == numeric)) == 7
    num=6;
elseif (sum(numeric7 == numeric)) == 7
    num=7;
elseif (sum(numeric8 == numeric)) == 7
    num=8;
elseif (sum(numeric9 == numeric)) == 7
    num = 9;
else
    num = 10;
end

```

在本课题中数字仪表的字符识别的第二种实现方式是本方式实在第一种实现方式的基础上进行的算法改进。首先同样需要对单个的数字进行形态模型，数学模型分析。

我们根据 0-9 的十个数字各自的特点很容易将其区分开，比如数字‘1’我们

可以直接根据其宽和高的比例,从 0-9 中分辨出数字 1。实现流程图如图 4-35,具体的算法分实现下面代码所示

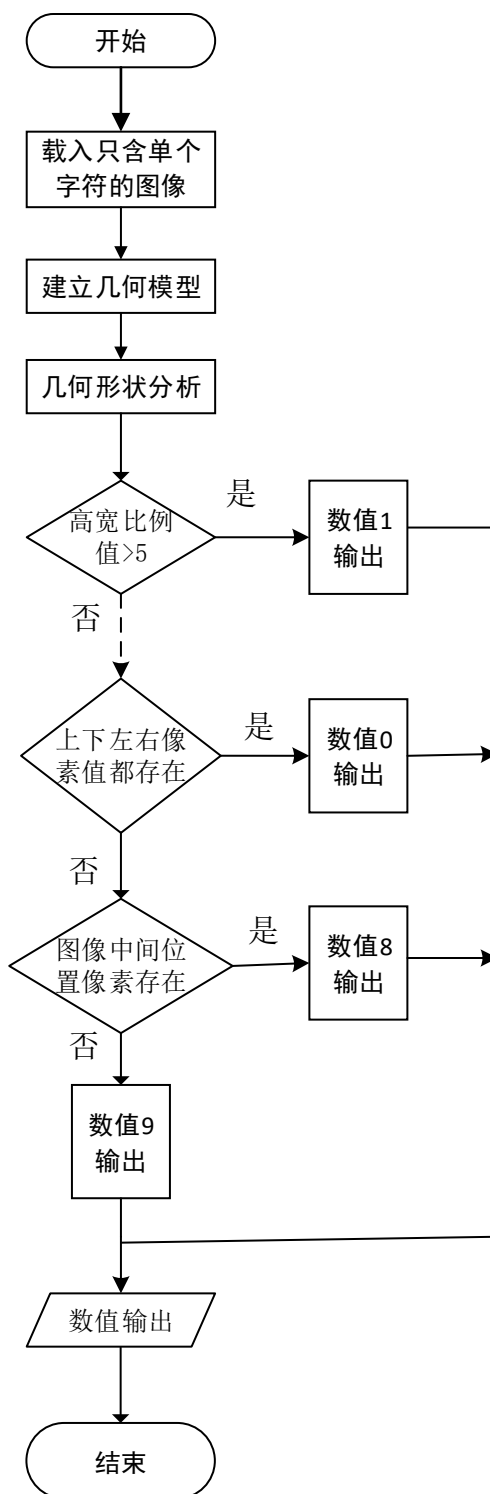


图 4-35 第二种数值识别算法程序流程图

```

%% 每张分离出的单个字符 pictx (为黑白图片)
% 这个地方的 hen 可以不用默认 40 就足够了, 算法是 hen 等于每个字符数字
% 的横或者竖的宽度, 不是长度
function [num]=IPImC2numORC(pictx,hen)
hen=40;
% hen 是指每一竖线的宽度乘以 2 的结果比如数字 7 的竖线的宽度像素个数,
% 当然 1 除外。
[m,n]=size(pictx);
if m/n>2
    num=1;
else
    %% 如果不是 1 则按其他算法识别
    picty=imresize(pictx,[180 120]); %对待识别图片固定大小尺寸,
    %检测中间 2 部分是否只有一条横线(4、7)、中间两条直线(9);
    % 把图片分成 3 部分—1—|—2—|—3—
    %           —1—|—2—|—3—
    %检测方法横着行扫描。
    pictt=picty(:,41:80); %行扫描数据
    j=0; %j 计算行数有多少个, 定义变量 j 的目的是计算并存储区间
    % '2' 的各列上出现横笔的个数。
    for i=1:180
        numtemp(i)= sum(pictt(i,:));
        if numtemp(i)>34
            K(i+1)=1; %K 用来统计在 41~80 列的区间上, 行上的像
            %素之和
        else
            K(i+1)=0;
        end
        if K(i)-K(i+1)==1 %上一行减下一行。
            j=j+1; %统计出现横笔的个数
        end
    end
end

h=sum(K(1:50)); %判断上半部分的分布情况, 如果上半部分有横笔则表
%示为 7 否则为 4, 一次类推
if j==1 && h==0 %表示是 (4 或 7) 接下来判断是 4 还是 7
    num=4;
elseif j==1 && h~=0
    num=7;
elseif j==1

```

```

num=0;
else    %说明 j=3 表示为三行上中下都为 2、5、8、3、6 这几个中的一
        %个
        %% 算法为：分别判断竖线分布情况
        %把数字分成 6 各部分取上面的 2-3 区间 row(31:61)之间的数据，
        %判断竖线的条数，竖线为 2 条则表示成 shuUpflg=2;
        %一条则为 shuUpflg=1;接下来判断 zuoyouUp=1;    %在左边，上部
        %分为竖线，等等，更具 0-9 数字的特征判断不同的数据
%        1_|
%        2_|
%        3_|
%        4_|
%        5_|
%        6_|
shuUp=picty(31:60,:);
shuDown=picty(91:120,:);
count=30;
dav=zeros(30,1);
for i=1:count
    dav(i)=sum(shuUp(i,:));    %统计
    if dav(i)>hen%横着求和
        shuUpflg=2; %
    else
        shuUpflg=1;
    end
end
for i=1:count
    if sum(shuDown(i,:))>hen%横着求和
        shuDownflg=2;
    else
        shuDownflg=1;
    end
end
%数据判断
if shuUpflg==2 && shuDownflg==2    %8
    num=8;
elseif shuUpflg==1 && shuDownflg==2    %6
    num=6;
else    %2、5、3
    zuoyouUp=0;

```

```

for i=1:120
    temp=sum(shuUp(:,i)); %对上半边竖线检测
    if temp>25 %为 1 的行至少 25 行
        break;
    end
end
if i<40
    zuoyouUp=1; %在左边，上部分为竖线
else
    zuoyouUp=2; %zuoyouUp=2 表示竖线在上边的右边位置
end
%下半部分
zuoyouDown=0;
for i=1:120
    temp=sum(shuDown(:,i)); %对上半边竖线检测
    if temp>25 %为 1 的行至少 25 行
        break;
    end
end
if i<40
    zuoyouDown=1; %在左边，上部分为竖线
else
    zuoyouDown=2;
end
if zuoyouUp==1 && zuoyouDown==2
    num=5;
elseif zuoyouUp==2 && zuoyouDown==2
    num=3;
elseif zuoyouUp==2 && zuoyouDown==1
    num=2;
else
    num=9;
end
if shuUpflg==2 && shuDownflg==1 && zuoyouDown==2
    num=9;
end
end
end
end
end

```

最后对两种识别方式加以对比，对比数据如下(在一般环境下的测试结果)

表 4.1 测试结果统计

50 次统计	0-9 数字的识别平均运算时间(t)	误差率(仪表上的数字)
方式一	0.067	0.001
方式二	0.046	0.015

从上面表中能很明显的看出，第二种识别的算法比第一种识别的算法运行的效率要高，但是运行的精度却几乎不变。

4.5 本章小结

本章是实现本课题的重要章节。在这一章中对实现本软件的算法进行详细的介绍，对本设计的处理步骤进行了逐一的介绍，本章中每个步骤都是严格按照程序执行的顺序进行安排，每个步骤都详细写明了处理的动机即为什么要这个过程，以及这个过程所产生的结果。

5 总结与展望

5.1 总结

本课题完成了数字仪表图像模式识别的相关工作，并呈现出比较好的结果。根据国内外图像模式识别产业的发展现状，对目前市场进行分析，了解产业推行的相关问题，本课题致力于设计出低成本，高用途的数字仪表模式识别软件。本系统实现由 3 部分组成，第一部分是图像获取，获取有用的数据信息，第二部分是图像处理，对获取的含有用的图像信息进行图像处理，第三部分是图像的识别，对处理后的图像进行图像的处理。最后提高部分对识别出的数据进行了数据处理，包括串口数据的发送和离散数据的波形显示，求解离散数据的线性拟合方程式等。

本课题以 Matlab 开发环境为设计平台，主要是对图像的相关算法进行研究，实现了数字仪表模式识别的各部分功能。

5.2 不足与展望

本课题实现了数字仪表的模式识别，对图像字符进行了准确的识别处理。但本设计未能将该套算法移植到嵌入式便携移动设备中。如直接移植到 DSP 或者 FPG 或者其他运行速度更快的处理其中，而不单单是电脑的软件上。

由于时间、个人能力、实验条件的限制，仅完成了当前总体设计。在今后的学习过程中将会继续本设计相关算法进行学习，不断提升相关知识运用能力，并不断完善本设计。主要从以下几个方面入手

- 1.实现便携式仪表的算法移植，让本设计算法及功能在便携式设备上运行。便于更多场合上的应用。

- 2.通过便携设备上的控制端口，实现数据传输功能，比如 RS485、RS232、TCP/IP 功能，方便衍生出不同应用环境下的数据交互，协同处理功能，结合实现设备的更加智能。

- 3.在便携的设备上实现本课题设计的识别算法，对不方便人工的仪表读数识别的环境中进行仪表读数的自动识别并回传数据。比如在工厂的液压监测环节中，先

通过本设计算法移植的便携产品接口将识别的数据发送给主控设备，对识别的数据进行检测，如果大于设置过压阈值，方便压力的自动调节，形成自动控制回路，进而避免了人工检测不必要的人力资源浪费。

参考文献

- [1] 楼彬彬. 基于模式分解的医学图像增强方法的研究与实现[D]. 东北大学, 2013-06-01.
- [2] 尚金奎. 基于粗糙集理论的图像分割方法研究[D]. 东北大学, 2008-06-01.
- [3] 赵丽娜. 液晶显示器缺陷自动识别系统研究与应用[D]. 苏州大学, 2010-11-01.
- [4] 石伟栋. 精确制导中目标识别与跟踪的应用研究[D]. 北京大学, 2011-03-01.
- [5] 易平, 陈福生, 邹红艳. 图像处理中消除噪声的方法[J]. 计算机应用与软件, 2003-05-12.
- [6] 赵丽娜. 液晶显示器缺陷自动识别系统研究与应用[D]. 苏州大学, 2010-11-01.
- [7] 张宁. 基于摄像方式的二维条码识别算法的研究[D]. 南京大学, 2013-03-01.
- [8] 刘玉芳. 随机散射表面扫描白光干涉信号的蒙特卡罗模拟[D]. 南昌, 2014-06-01.
- [9] 朱丽娟; 赵青兰. 基于结构相似性的全参考图像质量评价[J]. 信息通信, 2013-02-15.
- [10] 赵嘉兴. 图像处理基本概念[J]. 博客频道《网络 (<http://blog.csdn.net>)》.
- [11] 曹建波. 智能网络视频监控系统中目标跟踪技术的研究[D]. 福州, 2006-01-01.
- [12] 蔡立平. 车牌识别系统中的车牌定位算法研究[J]. 科技信息, 2009-01-15.
- [13] 张群会; 李贵敏; 蔺宝华; 韩波. 基于最大熵-方差模型的图像分割方法[J]. 计算机技术与发展, 2011-06-10.
- [14] 邢青青; 罗新斌. 基于小波去噪和机器视觉的边缘检测[J]. 有色金属, 2011-05-15.
- [15] 岳斌. 立体图像压缩算法研究[D]. 天津大学, 2008-05-01.
- [16] 徐献灵; 林奕水. 图像边缘检测算法比较与分析[J]. 自动化与信息工程, 2007-09-15.
- [17] 张伟. 简牍图像中文字修复的研究与应用[D]. 成都理工大学, 2008-04-01.
- [18] 杨帆. 万能工具显微镜的图像法测量研究[J]. 科技广场, 2013-02-28.
- [19] 叶茂锹. 彩色图像的文本信息提取研究[D]. 东华大学, 2009-01-01.
- [21] 邢青青; 罗新斌. 基于小波去噪和机器视觉的边缘检测[J]. 有色金属, 2011-05-15.
- [22] Recognizing Text-Based Traffic Signs [D]. IEEE, 2012-04-21.
- [23] A New Approach for License Plate Detection and Localization-13 [D]. IEEE, 2015-10-02.
- [24] Car License Plate Detection-4[J]. IEEE, 2015-10-03.
- [25] Vehicle Plate Extraction and Recognition using-7.[J], IEEE, 2015.
- [26] Automatic number plate recognition system for vehicle-4[J]. IEEE, 2016.
- [27] Recognize text using optical character recognition - MATLAB ocr [J]. MathWorks .
- [28] 张德丰. 数字图像处理[M]. 2014.

附录

```

#include <REG51.H>

#define Num_Point      0x80

    ///加小数点的方法:
Tab[x]|Num_Point
unsigned char code Tab[] =
{0xc0,0xcf,0xa4,0xb0,0x99,0x92,0x82,
0xf8,0x80,0x90,0x88,0x83,0xc6,0xa1,0x8
6,0x8e,0xff,0x7f}; //0 1 2 3 4 5 6 7 8
9 A B C D F 灭 . // 共阳数码管//h g f
e d c b a
//h g f e d c b a
// 高位->低位

sbit  LOAD=P2^2;
sbit  CLK=P2^1;
sbit  SDK=P2^0;
sbit  S4=P2^7;
sbit  S3=P2^6;
sbit  S2=P2^5;
sbit  S1=P2^4;

unsigned char flag=0;
void show1(void);
void show2(void);
void show3(void);

void delay10ms(unsigned char temp)
{
    unsigned char i, j, t;

    t=temp;

    while(t!=0)

        {
            for(i=20;i>0;i--)

                for(j=248;j>0;j--)

                    {

                        t--;

                    }

        }

}

void SPI_595(unsigned char out_data)
{
    unsigned char i, temp;

    for(i=0;i<8;i++)

        {
            CLK=0;

            temp=out_data&0x80;

            if(temp==0x80)

                SDK=1;

            else SDK=0;

            out_data=out_data<<1;

            CLK=1;

        }

}

```

```
char flag1=0;

char key_value(void)
{
    if(S4==0)
    {
        delay10ms(1);
        if(S4==0) flag1=2;
    }
    else if(S3==0)
    {
        delay10ms(1);
        if(S3==0) flag1=1;
    }
    else if(S2==0)
    {
        delay10ms(1);
        if(S2==0) flag1=3;
    }
    else if(S1==0)
    {
        delay10ms(1);
        if(S1==0) flag1=4;
    }
    return flag1;
}

void main(void)
{
    delay10ms(2);
    while(S4==1&&S3==1&&S2==1)
    {
        delay10ms(1);
    }
    if(S2==0)
    {
        flag=3;
    }
    else if(S3==0)
    {
        flag=1;
    }
    else if(S4==0)
    {
        flag=2;
    }
    while(1)
    {
        flag=key_value();
        if(flag==1) show1();
        else if(flag==2)
            show2();
        else if
            (flag==3) show3();
    }
```


function	((Blue_y(PY1,1)>=5)&&(PY1>1)) % $\times \phi \pm \beta$
[Plate]=rgb2roicollect(ROIPicture,Rmin	PY1=PY1-1;
,Rmax,Gmin,Gmax,Bmin,Bmax)	end
[y,x,z]=size(ROIPicture);	PY2=MaxY;
myI=double(ROIPicture);	while ((Blue_y(PY2,1)>=5)&&(PY2<y)) %
tic	PY2=PY2+1;
	end
Blue_y=zeros(y,1); %½ ¨ Á¢0¾0Œó	IY=ROIPicture(PY1:PY2, :, :);
for i=1:y	Blue_x=zeros(1,x);
for j=1:x	for j=1:x
%	for i=PY1:PY2
if((myI(i,j,1)<=30)&&(myI(i,j,2)<=62)	%
&&(myI(i,j,2)>=51))&&(myI(i,j,3)<=142	if((myI(i,j,1)<=30)&&(myI(i,j,2)<=62)
)&&(myI(i,j,3)>=119)))	&&(myI(i,j,2)>=51))&&(myI(i,j,3)<=142
if((myI(i,j,1)<=Rmax)&&(myI(i,j,1)>=Rm)&&(myI(i,j,3)>=119)))
in)&&(myI(i,j,2)<=Gmax)&&(myI(i,j,2)>	
=Gmin))&&(myI(i,j,3)<=Bmax)&&(myI(i,j	if((myI(i,j,1)<=Rmax)&&(myI(i,j,1)>=Rm
,3)>=Bmin)))	in)&&(myI(i,j,2)<=Gmax)&&(myI(i,j,2)>
Blue_y(i,1)=	=Gmin))&&(myI(i,j,3)<=Bmax)&&(myI(i,j
Blue_y(i,1)+1;	,3)>=Bmin)))
end	Blue_x(1,j)=
end	Blue_x(1,j)+1;
end	end
[temp MaxY]=max(Blue_y);	end
x\yÓ³ ÉäÇóÈ;É ¨ ¨ ÆëĬĬĬ×óŒŒ±β½Ç Î»ŒĬ	end
PY1=MaxY;	PX1=1;
while	while ((Blue_x(1,PX1)<3)&&(PX1<x))

```
PX1=PX1+1;

end

PX2=x;

while ((Blue_x(1,PX2)<3)&&(PX2>PX1))

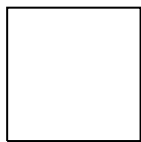
    PX2=PX2-1;

end

PX1=PX1-2;
if PX1<1
    PX1=1;
end
PX2=PX2+2;
if PX2>x
    PX2=x;
end
Plate=RGBpicture(PY1:PY2,PX1:PX2
,:);
t=toc;
% figure,imshow(RGBpicture);
%
figure,plot(Blue_y);figure,plot(
Blue_x);figure,imshow(IY);
% figure,imshow(Plate);
```

翻译部分

英文原文



Recognition of Text in Wine Label images

Abstract:

In this paper, an automatic recognition system for Wine label images is described. The system includes detection and extraction of text for the recognition for Wine label images. It deals with impediments caused by different font styles and font sizes, as well as illumination changes and noise effects. Firstly, the text region is extracted by an edge-histogram, and the text is binarized by clustering. Secondly, the extracted text is divided into individual characters, which are recognized by using the Multi-Layer Perceptron. A shape-based statistical feature is adopted and the recognition results are generated for each character. The system has been implemented in a mobile phone and is demonstrated to show an acceptable performance.

Key Words: Text detection, text segmentation, recognition

1 INTRODUCTION

Natural scene text usually provides semantic information, and is often the crucial key to understanding the image content. These days, with the recent explosion of handheld imaging devices, research on document image analysis entered a new era where breakthroughs are required^{[1], [2]}.

The main challenge of natural scene text understanding is to design a system as versatile as possible to handle all variability in daily life, meaning variable targets with unknown layout, scene text, several character fonts and sizes and variability in imaging conditions with uneven lighting, shadowing and aliasing[3]. Although many commercial OCR systems have good recognition capabilities on high quality scanned documents under well controlled environments, much higher error rates are common for character recognition in real world situations, when the input images do not satisfy the enforced

constraints.

The interest of this paper is in the automatic detection and recognition of texts in Wine label images for mobile applications. The application scenario is as follows: a camera is used to capture an image, or a sequence of images containing the text of interest.

There have been techniques previously developed for the detection and recognition of written text, targeting applications such as document image processing [4], [5], content-based image/video indexing [6], [7], assistance for visually impaired persons [8], text restoration [9], [10], and text recognition in natural scene images [11], [12]. In order to obtain information in an arbitrary image or a scene captured using a camera, the recognition system automatically recognizes characters of various conditions in the scene, and then provides information about the location of text within the captured image. The text regions, detected in the previous step, can be fed into the recognition step for the classification. This work is related to existing research in the recognition of text on special objects such as car license plates [13]. While the early methods required manual selection of the text area [14], [15], recent attempts have moved toward automatic detection and recognition of text in natural scenes, for mobile system applications [11], [15].

Aside from the challenges for the recognition of text in natural scenes, the restrictions on execution time or the overall complexity of the algorithm also need to be enforced in real-time environments, as in mobile systems or PDAs (Personal Digital Assistants). In order to overcome these problems, more robust features are required for the recognition. This paper presents an automatic recognition system for Wine label images, where the text usually represents a Wine name. Efficient methods for text detection, recognition, with their robustness features, are given in the following sections.

2 TEXT DETECTION

Texts in Wine label images are affected by changes of lights, orientation, where the orientation is decided by the viewing angle of the camera. In this paper, it is assumed that the text region is located around the center line. In this section, a hierarchical detection

framework is presented, including the computation of the edge histogram and the text extraction using fuzzy clustering and connected component analysis.

2.1 Detection of the candidate text region

Although the intensity of pixels in a digital image are important features for text detection, it is not robust in dealing with the variations in the lighting. On the other hand, the edge component is less sensitive to light changes and therefore is more dependable for identifying the text detection. The canny edge detector is applied to the gray-scale image to obtain the edges of the input image. In order to detect the candidates for the region containing the text, the horizontal profile of the edges are computed. In this paper, it is assumed that texts are aligned horizontally, but the proposed approach can also be extended to vertically aligned texts. Fig. 1 shows edge profiles

in the horizontal and vertical directions. By scanning vertically, to the upper or lower ends of the image, starting from the center line of the image, valleys in the histogram are detected, which lead to the identification of the candidate regions with the horizontal profile, as shown in Fig. 1(b) with a blue box. Only the region detected from the horizontal profile is used for the vertical detection of the candidate region, as shown in Fig. 1(c).

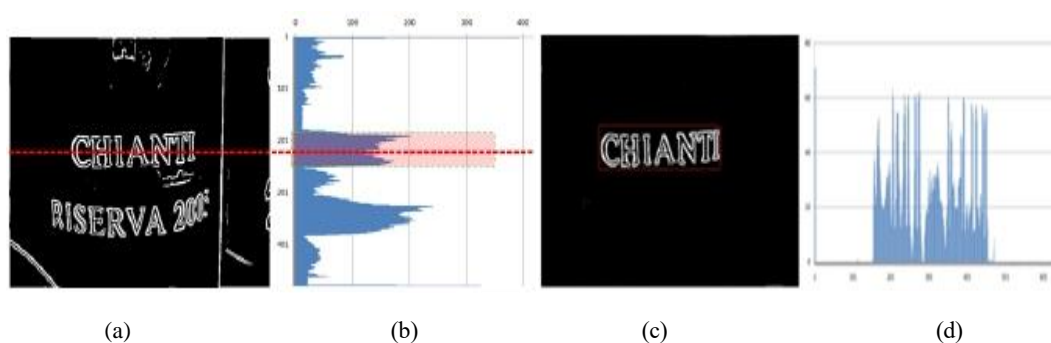


Fig. 1: Detection of the candidate text region with edge profiles:

(a) Detected edges (b) Horizontal edge profile (c) Text region obtained by from the horizontal profile (d) Vertical edge profile.

2.2 Text Image binarization

After the candidate text region is detected, the detected text is binarized. Although the candidate text region is detected, the text in the region may still contain light variations

along the text itself. To deal with light changes in the image, fuzzy c-means (FCM) clustering is used for the text segmentation as it allows one piece of data to belong to two or more clusters and reduces the likelihood of missing correct matches due to the effects of the noise. It is an unsupervised approach based on the minimization of an objective function that has been used in the image segmentation^[16].

Fig. 2 shows the binarized result for a sample of text. If text segmentation is performed for the entire candidate region, as shown in Fig. 2(a), the digitized result for the first character is not good because of uneven illumination and usually segmentation in local regions can improve the results. Therefore, individual character regions are separated by detecting gaps in the vertical edge profiles of the candidate sub-regions. If a value in the vertical edge profile is less than mean/K for some constant K , we separate the candidate region into local regions for the individual characters. Then, each region enclosed by a red box as shown in Fig. 2(b) has FCM clustering applied to it for the text segmentation. The results are shown in Fig. 2(c).



Fig. 2: Segmentation and binarization by the proposed method: (a) Segmentation results, (b) Separated image by vertical profile, (c) Segmentation results by local features.

2.3 Character Segmentation

The objective of the character segmentation with the text image is to bind up the connected components into a sequence of characters^[17]. Given a binarized image of the text region, the first step is to collect the information of every connected component,

namely the width and height of its bounding rectangle, and number of constituent pixels, etc.



Fig. 3: Examples of character segmentation

The second step is for noise removal; if a connected component C_i , whose width, height and number of pixels are w_i , h_i , and n_i , respectively, satisfies the following condition, it is regarded as noise and is removed from further processing:

$$\{ w_i/h_i > 10 \text{ OR } w_i/h_i < 1/10 \} \text{ AND } \{ n_i/(w_i * h_i) < 0.5 \} \quad (1)$$

3 TEXT RECOGNITION

A recognition system for the characters captured from Wine Label images is more difficult than that needed for normal document analysis. For a traditional document analysis task, a scanner with a stable embedded lighting system is used to obtain high quality images. In the case of Wine Label images, however, the noise ratio is much higher because the images are captured by a camera under outdoor conditions with various lighting situations. In addition, there is a wide variety of font styles and font sizes.

For the recognition of characters, two different types of approaches have been developed – structural and statistical. The structural approaches attempt to split the characters into their constituent consonants and vowels, and then recognize each component independently. This approach is intuitive and reduces the number of pattern classes drastically, but the problem of character decomposition becomes non-trivial when

the character pattern is degraded, or the image contains some noise or character decorations. The statistical approaches, on the other hand, regard the whole character as a pattern and try to differentiate it from other characters using mesh-based shape information. This approach is simple to implement, but the recognition accuracy is not normally satisfactory because there are many confusing pairs having similar shapes in characters.

Based on the these observations, a statistical approach is selected, which is felt to be more appropriate to the outdoor images containing lighting variations, noise and changes in the use of font styles and character decorations. In addition, this approach has an added advantage of producing a reasonable set of recognition candidates for the input characters by searching similar patterns from the entire character classes.

Once the text image is segmented into isolated characters, the new proposed system classifies the characters one by one using a Multi-Layer Perceptron(MLP).

A shape-based statistical feature is adopted, which is adequate for character recognition, as shown in Fig. 4.

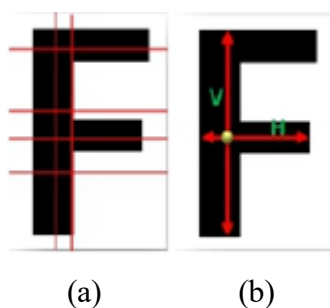


Fig. 4: An example of feature extraction: (a) 3-5 non-linear mesh, (b) Horizontal (H) and vertical (V) segments from the input image,

Assuming that the character pattern is an $H \times W$ image $f(i, j)$, where (i, j) is the pixel at i -th row and j -th column ($1 \leq i \leq H, 1 \leq j \leq W$), then the construction of an $N \times M$ mesh ($N \leq H, M \leq W$) can be described as follows. First, the horizontal and vertical projection profiles, $h(i)$ and $v(j)$, are computed using

$$h(i) = \sum_{k=1}^W f(i, k), 1 < i \leq H, \quad (2)$$

$$v(j) = \sum_{k=1}^H f(k, j), 1 < j \leq W, \quad (3)$$

and the rows of the input image, indexed from 1 to H , are partitioned into N consecutive strips, $[1, e_1], [e_1+1, e_2], \dots, [e_{N-1}+1, H]$, in which the boundary indices e_1, e_2, \dots, e_{N-1} are determined so that the black pixel densities defined by the following equation are the same for all strips:

$$D(k) = \sum_{i=s_k}^{e_k} h(i), \quad (4)$$

where $s_1 = 1, s_{k+1} = e_k + 1, e_N = H$, and $1 \leq k \leq N$. Similarly, the columns of the input image, indexed from 1 to W , are partitioned into M strips with the vertical projection $v(j)$. Combining the boundary indices of the N horizontal strips and M vertical strips, an $N \times M$ mesh can be formed, as in Fig. 4(a), where $N=3, M=5$. The non-linear partitioning of the input pattern is used to allow for the variation of different font styles and character decorations that are commonly used in characters.

Next, the four directional segments are calculated for every black pixel in the image, considering the fact that the structural characteristics of characters are represented by horizontal and vertical lines. Given a black pixel (x, y) in the input character image, the two run-lengths $RLH_{x,y}$ and $RLV_{x,y}$ are computed by measuring the length of horizontal and vertical runs passing through the pixel (x, y) , as shown in Fig. 4(b). Using these two values, compute the values of horizontal and vertical contributions $DCH_{x,y}$ and $DCV_{x,y}$ as follows.

$$DCH_{x,y} = \frac{RLH_{x,y}}{(RLH_{x,y} + RLV_{x,y})} \quad (5)$$

$$DCV_{x,y} = \frac{RLV_{x,y}}{(RLH_{x,y} + RLV_{x,y})} \quad (6)$$

The directional contribution values $DCH_{x,y}$ and $DCV_{x,y}$ are averaged over all the black pixels within each cell of the $N \times M$ mesh to get $2 \times N \times M$ feature values. In the proposed system, a 3×5 mesh is selected, according to statistics of the aspect ratio of characters. So, the character pattern in a digital image is transformed to a $30(3 \times 5 \times 2)$ dimensional feature vector.

4 EXPERIMENTAL RESULT

Experiments have been conducted on Wine label images captured in different light conditions. The text appearance varies with different colors, orientations and font sizes. Wine label images are captured by a Samsung Smart-phone, with an image resolution of 640×480 . The mobile system is equipped with intel XScale PXA270 416MHz and the operating system is MS window mobile 5.0 for Pocket PC. The proposed character recognition algorithm has been evaluated with 517 characters extracted from Wine label images. The recognition accuracy is summarized in Table 1 where it can be seen that the recognition accuracy is 97.5%.

Table 1: Recognition accuracy for characters

Success	Failure	Total
504	13	517
97.5%	2.5%	-

5 CONCLUSIONS

This paper has described a system for recognition of texts in Wine label images captured by a mobile phone camera. The captured images are taken in natural lighting environments, where noise and irregularities of fonts occur frequently. The proposed system can robustly detect and recognize texts from such images. The edge-based method is used for the detection of the candidate text region, and the text is digitized by using fuzzy c-means clustering in local regions of individual characters. For character recognition, a shape-based statistical feature extraction is employed. The experimental

results show that the proposed method has been successfully applied to recognize for the Wine label.

ACKNOWLEDGEMENTS

This research was supported by the MKE(The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute for Information Technology Advancement) (IITA-2009-(C1090-0903-0008)).

REFERENCES

- [1] G. Obinata, A. Dutta, Vision Systems: Segmentation and Pattern Recognition, I Tech, 2007, pp. 307-332.
- [2] K.J. Jung, K.I. Kim, A.K. Jain, "Text information extraction in images and video: a survey," Pattern Recognition, 2004, vo. 37, pp. 977-997.
- [3] C. Thillou, B. Gosselin, "Natural Scene Text Understanding", Vision Systems, Segmentation and Pattern Recognition, 2007, Ch.16, pp. 307-333.
- [4] G. Nagy, "Twenty years of document image analysis" IEEE Trans. on Pattern Analysis and Machine Intelligence, 2000, vol. 22, no. 1, pp. 38-62.
- [5] H. Peng, F. Long, Z. Chi, "Document image recognition based on template matching of component block projections," IEEE Trans. on Pattern Analysis and Machine Intelligence, 2003, vol. 25, no. 9, pp. 1188-1192.
- [6] H. Li, D. Doermann, O. Kia, "Automatic text detection and tracking in digital videos," IEEE Trans. on Image Processing, 2000, vol. 9, no. 1, pp. 147-156.
- [7] J. Xi, X. Hua, L. Wenyin, H.J. Zhang, "A video text detection and recognition system," International Conference on Multimedia and Expo, 2001, pp. 873-876.
- [8] N. Ezaki, K. Kiyota, B.T. Minh, M. Bulacu, L. Schomaker, "Improved text-detection methods for a camera-based text reading system for blind persons," International Conference on Document Analysis and Recognition, 2005, pp. 257-261.
- [9] Q. Ye, J. Jiao, J. Huang, H. Yu, "Text detection and restoration in natural scene images," Journal

- of Visual Communication and Image Representation, 2007, vol. 18, pp. 504-513. [10] J.G. Lim, J.H. Park and G.G. Medioni, "Text segmentation in color images using tensor voting," Image and Vision Computing, 2007, vol. 25, pp. 671-685.
- [11] J. Zhang, X. Chen, A. Hanneman, J. Yang and A. Waibel, "A robust approach for recognition of text embedded in natural scenes," International Conference on Pattern Recognition, 2002, vol. 3, pp. 204-207.
- [12] J. Yang, J. Gao, Y. Zhang and A. Waibel, "Toward automatic sign translation," Human Language Technology, 2001, pp. 269-274.
- [13] R. Mullot, C. Olivier, J.L. Bourdon, P. Courtellemont, J.Labiche, and Y. Lecourtier, "Automatic extraction methods of container identity number and registration plates of cars," Int. Conference Industrial Electronics, Control, Instrumentation, 1991, vol. 2591, pp. 1739-1744.
- [14] Y. Watanabe, Y. Okada, Y.B. Kim and T. Takeda, "Translation camera," International Conference on Image Processing, 1998, pp. 613-617.
- [15] J. Yang, W. Yang, M. Denecke and A. Waibel, "Smart sight: A tourist assistant system," Int. Symp. Wearable Computers, 1999, pp. 73-78.
- [16] Y.W. Lim and S.U. Lee, "On the color image segmentation algorithm based on the thresholding and the fuzzy c-means techniques," Pattern Recognition, 1990, vol. 23, no. 9, pp. 935-952.
- [17] D.A. Forsyth, J. Ponce, Computer Vision A Modern Approach, Prentice Hall, 2003

中文翻译

在酒标图片中的文字识别

摘要:

本文是对酒标图像自动识别系统进行的介绍,该系统包括文字识别的酒标图像检测。对不同的字体样式和字体大小的图像进行处理,以及对照明变化和噪声的影响阻碍进行介绍。首先,文本区域是由边缘直方图萃取,先对图像进行二值化处理。其次,所提取的文本被分成单个的字符,通过使用对字符进行识别。基于形状统计特征被采用和识别结果是每个字符产生的。该系统已经在移动电话中被实现,并表现出以显示可接受的性能。

关键词: 文本检测, 文本分割, 识别

1 引言

自然场景文本通常提供语义信息,而且往往关键的是理解图像内容。随着最近几年的手持式成像装置的火热,对图文图像的分析研究进入需要突破的新时代^[1],^[2]。

自然场景文本的理解面临的主要挑战是设计一个系统,多功能地处理日常生活中的一切变化,这意味着未知的布局的场景文本,其所受条件的干扰,比如几个字符的字体和大小,变异与光线不均匀成像条件变量指标,阴影和混叠等^[3]。虽然许多商业的 OCR 系统目前能很好地控制环境中高品质的扫描文档并具有良好的识别能力,更高的误差率是在现实世界中的情况下的字符识别,常见的图像一般是不符合特定条件下的图像约束限制。

本文的识别是在酒标上的图像的移动应用文本的自动检测和识别。应用场景为:一个相机被用于捕获图像,或含有感兴趣的文本图像的序列。

已经有如文档图像处理^[4]^[5],基于内容的图像或者视频索引^[6]^[7],为视障人士提供援助,以前的检测和识别的书面文字的开发技术,比如目标应用^[8],文本恢复^[9],^[10],以及文字识别在自然场景的图像^[11],^[12]。为了获得在任意的图像或信息,使用摄像机捕获的场景时,识别系统自动识别在现场的各种条件的字符,然后提供

关于文本所捕获的感兴趣区域,并将感兴趣区域分割出来。这项工作涉及到现有的研究对象,特殊对象识别的文本,如车牌^[13]。而所需的早期方法文本区域^{[14] [15]}的手动选择,最近尝试对移动物体自动检测和识别在自然场景的文本,对移动系统的应用程序^{[11] [15]}。

除了用于识别在自然场景文本,在执行时间或算法的整体复杂性的限制也需要实时的环境中执行,如在移动通信系统或 PDA (个人数字助理)。为了克服这些问题,需要用于识别更强大的功能。本文介绍了葡萄酒标签的图像,其中文本通常代表一个葡萄酒名称的自动识别系统。文本检测,识别,用自身所含有的特性,使用有效方法,在下面的章节中给出。

2 文本检测

在酒标上的图像文字是受灯光,方向的影响,那里的方向是由相机的视角决定的。在本文中,假设该文本区位于围绕中心线。在本节中,一个分层检测框架呈现,包括边缘直方图的计算,并使用模糊聚类 and 连接成分分析的文本提取。

2.1 候选文本区域的检测

虽然在数字图像像素的强度是用于文本检测的重要特征,它是不处理在照明变化的鲁棒性。另一方面,边缘分量对光变化较不敏感,因此,适用于识别该文本。Canny 边缘检测器被施加到灰度级图像以获得输入图像的边缘。为了检测包含文本的候选区域,便于边缘的水平轮廓计算。在本文中,假定文本是水平对齐,但所提出的方法也可以扩展到垂直对齐文本。图 1 显示边缘轮廓

在水平和垂直方向。通过垂直扫描时,对图像的上端或下端,从图像的中心行开始,在直方图山谷被检测到,从而导致与水平轮廓的候选区域的识别,如图所示。图 1 (b) 用蓝色框。只有从水平轮廓检测的区域被用于候选区域的垂直检测,如图所示。图 1 (c)。在水平或垂直方向的边缘的总和是通过该区域的大小,这近似于除以沿着扫描线的边缘密度。

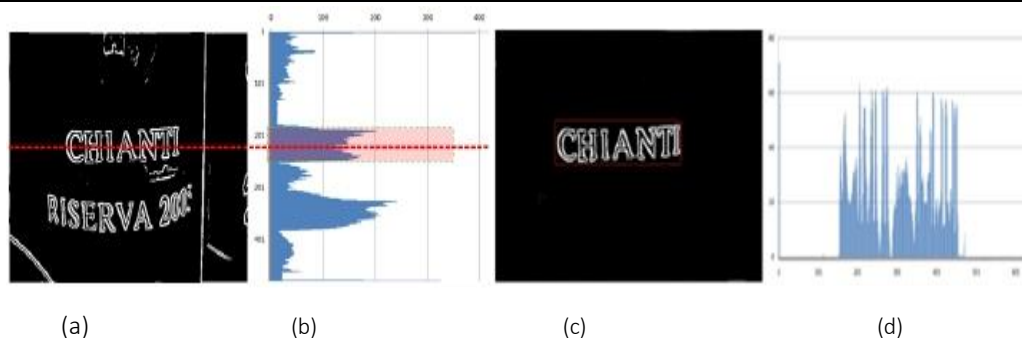


图 1： 检测图像的文字边缘:

(a) 边缘检测 (b) 水平方向边缘检测 (c) 从水平方向识别文本边缘(d) 垂直边缘检测

2.2 文本图像二值化

在检测到的候选文字区域后，检测出的文本被二值化。虽然检测出候选文本区域，该地区的文本仍然可能包含沿着文本本身光线的变化。要处理的图像中的光的变化，(FCM) 聚类被用于文本分割模糊 c 均值，因为它允许一个数据片属于两个或多个聚类并降低丢失正确匹配由于的影响的可能性噪音。它是基于已经在图像分割[16]被使用的目标函数的最小化的无监督的方式。

图 2 示出了用于文本样本的二值化结果。如果文本分割为整个候选区域进行的，如图 2 所示。图 2 (a)，对于第一字符的数字化结果不是很好，因为照度不均匀，通常分割在局部区域可改善结果。因此，个性区域由在候选子区域的垂直边缘型材检测间隙分开。如果在垂直边缘轮廓的值小于平均值/ K 一些常数 K ，我们的候选区域分成局部区域的单个字符。然后，每一个区域由一个红色框包围，如图。

图 2 (b) 具有应用到它的文本分割 FCM 聚类。结果示于图图 2 (c)。



图 2: 分割和二值化所提出的方法: (1) 分割的结果, (2) 独立图像垂直剖面, (3) 分割结果由当地特色。

2.3 字符的分割

字符分割与文本图像的目标是到所连接的组件绑定成字[17]的序列。中的文字区域的二值化图像中, 第一步骤是收集每个连接设备的信息, 即宽度和它的边界矩形的高度, 并且构成像素数等。

第二步骤是去除噪声; 如果一个连接的组件的 C , 其宽度, 高度和像素数为 W_i , H_i 和 N_i 分别满足以下条件, 它被视为噪声, 并从进一步的处理中移除:

$$\{ w_i/h_i > 10 \text{ OR } w_i/h_i < 1/10 \} \text{ AND } \{ n_i/(w_i * h_i) < 0.5 \} \quad (1)$$

图 3 展示了一些 关于字符区域特性的例子



Fig. 3: 字符分隔的例子

3 文字识别

从酒标图像捕获的字符的识别系统比所需的正常文件分析更加困难。对于一个传统的文档分析任务, 具有稳定的嵌入式照明系统的扫描器被用来获得高质量的图像。在葡萄酒标签图像的情况下, 然而, 信噪比高得多, 因为图像通过照相机与各种照明的情况下的户外条件下拍摄。此外, 还有各种各样的字体样式和字体大小。

用于字符识别的, 两种不同类型的方法已经开发 - 结构和统计。结构的方法试图将字符分成其组成辅音和元音, 然后独立地识别每个组件。这种方法是直观的,

并减少图案类的数量急剧,但字符分解的问题变得非平凡当字符图形降低,或图像中包含一些噪音或字符装饰。的统计方法,另一方面,把整个的字符作为图案,并尝试使用基于网格的形状信息从其他的字符区别开来。这种方法实现起来很简单,但识别精度是通常不令人满意的,因为有具有字符形状相似许多混乱对。

基于这些观察,选择一个统计方法,其认为是更适合于含有在使用字体样式和字符装饰照明变化,噪声和变化的室外图像。此外,这种方法有通过从整个字符类别搜索类似的模式产生合理组识别候补为输入字符的一个附加优点。一旦文本图像分割成分离的字符,该新提出的系统分类的字符一个一个使用多层感知器(MLP)。

基于形状统计特征被采用,这是足够的字符识别,如图。4.鉴于制备由黑白图像,网管非线性目和水平(H)两个定向段和垂直(V)方向表示的字符图形,并 then2-NM 特征值是使用网格计算并这两个方向的细分。

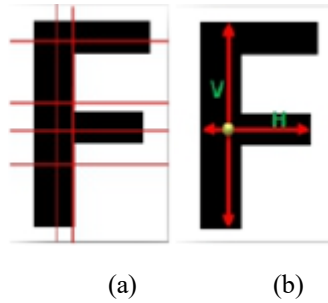


Fig. 4: 图像特征提取的例子: (A) 3-5 非线性网, (B) 水平 (H) 和从输入图像的垂直 (V) 的链段,

假设字符图形是一个 $H \times W$ 图像 $f(I, J)$, 其中 (I, J) 是在像素 i 行和第 j 列 ($1 \leq i \leq H, 1 \leq j \leq W$), 则网管目 ($N \leq H, M \leq W$) 的结构可作如下说明。首先, 在水平和垂直投影轮廓中, $h(i)$ 和 $v(j)$, 使用计算

$$h(i) = \sum_{k=1}^W f(i, k), 1 \leq i \leq H, \quad (2)$$

$$v(j) = \sum_{k=1}^H f(k, j), 1 \leq j \leq W, \quad (3)$$

和输入图像的行，索引从 1 到 H，被划分为 N 个连续条，[1, E1], [E1 +1, E2], ..., [CN-1+1, H]，在该边界指数 E1, E2, ..., EN-1 被确定，使得由下面的等式定义的黑像素密度对于所有条带相同的：

$$D(k) = \sum_{i=S_k}^{E_k} h(i), \quad (4)$$

其中， $S_1 = 1$ ， $S_{K+1} = E_K + 1$ ， $E_N = H$ ，以及 $1 \leq k \leq N$ 。类似地，输入图像的列，索引从 1 至 W，被划分为与垂直投影 VM 条（j）条。N 个水平线条和 M 垂直条的边界指数组合，一个 NxM 网格可以形成，如图图 4（a），其中 $N=3$ ， $M=5$ 。输入模式的非线性分区被用于允许对不同的字体样式和字符装饰常用于字符中使用的变异。

下，在四个方向的段被计算为图像中的每个黑色像素，考虑到字符的结构特性是由水平和垂直线表示的事实。给定了输入的字符图像中的黑色像素（x, y）时，在两个游程 $RLH_{x,y}$ 和 $RLV_{x,y}$ 的测量水平和垂直的长度贯穿像素传递（X, Y）计算，如图在图图 4（b）。使用这两个值，如下计算的水平和垂直的贡献 $DCH_{x,y}$ 和 $DCV_{x,y}$ 的值如下

$$DCH_{x,y} = \frac{RLH_{x,y}}{(RLH_{x,y} + RLV_{x,y})} \quad (5)$$

$$DCV_{x,y} = \frac{RLV_{x,y}}{(RLH_{x,y} + RLV_{x,y})} \quad (6)$$

定向贡献值 $DCH_{x,y}$ 和 $DCV_{x,y}$ 的平均值网管的每个小区之内的所有黑像素网格得到 $2 \times N \times M$ 特征值在所提出的系统中，一个 3×5 网格被选择，根据字符的纵横比的统计。这样，在数字图像的字符图形被变换为 $30(3 \times 5 \times 2)$ 维特征向量。

4 实验结果

实验已经在不同光线条件下拍摄的葡萄酒标签的图像进行。文本外观具有不同的颜色，取向和字体大小而变化。酒标图像通过三星智能手机拍摄的， 640×480 的图像分辨率。该移动系统配备了 Intel XScale PXA270 处理器 416MHz 的和操作

系统是 MS 窗口移动 5.0 的 Pocket PC。所提出的字符识别算法进行了评估与葡萄酒标签图像中提取 517 个字符。识别精度总结于表 1，其中可以看到，该识别精度是 97.5%。

Table 1: Recognition accuracy for characters

Success	Failure	Total
504	13	517
97.5%	2.5%	-

5 结论

本文描述了一个系统，用于识别通过手机相机拍摄的葡萄酒标签上的图像文本。拍摄的图像会采取自然光照环境中，噪音和字体的违规行为时有发生。该系统能够稳健地检测并从这些图像识别文本。的基于边缘的方法，用于检测所述候选文本区域，文本是由单个字符的局部区域使用模糊和均值聚类数字化。对于字符识别，常采用基于形状统计特征提取。实验结果表明，该方法已成功地应用到识别葡萄酒的标签。

参考文献

- [1] G. Obinata, A. Dutta, Vision Systems: Segmentation and Pattern Recognition, I-Tech, 2007, pp. 307-332.
- [2] K.J. Jung, K.I. Kim, A.K. Jain, "Text information extraction in images and video: a survey," Pattern Recognition, 2004, vo. 37, pp. 977-997.
- [3] C. Thillou, B. Gosselin, "Natural Scene Text Understanding", Vision Systems, Segmentation and Pattern Recognition, 2007, Ch.16, pp, 307-333.

- [4] G. Nagy, "Twenty years of document image analysis" IEEE Trans. on Pattern Analysis and Machine Intelligence, 2000, vol. 22, no. 1, pp. 38-62.
- [5] H. Peng, F. Long, Z. Chi, "Document image recognition based on template matching of component block projections," IEEE Trans. on Pattern Analysis and Machine Intelligence, 2003, vol. 25, no. 9, pp. 1188-1192.
- [6] H. Li, D. Doermann, O. Kia, "Automatic text detection and tracking in digital videos," IEEE Trans. on Image Processing, 2000, vol. 9, no. 1, pp. 147-156.
- [7] J. Xi, X. Hua, L. Wenying, H.J. Zhang, "A video text detection and recognition system," International Conference on Multimedia and Expo, 2001, pp. 873-876.
- [8] N. Ezaki, K. Kiyota, B.T. Minh, M. Bulacu, L. Schomaker, "Improved text-detection methods for a camera-based text reading system for blind persons," International Conference on Document Analysis and Recognition, 2005, pp. 257-261.
- [9] Q. Ye, J. Jiao, J. Huang, H. Yu, "Text detection and restoration in natural scene images," Journal of Visual Communication and Image Representation, 2007, vol. 18, pp. 504-513. [10] J.G. Lim, J.H. Park and G.G. Medioni, "Text segmentation in color images using tensor voting," Image and Vision Computing, 2007, vol. 25, pp. 671-685.
- [11] J. Zhang, X. Chen, A. Hanneman, J. Yang and A. Waibel, "A robust approach for recognition of text embedded in natural scenes," International Conference on Pattern Recognition, 2002, vol. 3, pp. 204-207.
- [12] J. Yang, J. Gao, Y. Zhang and A. Waibel, "Toward automatic sign translation," Human Language Technology, 2001, pp. 269-274.
- [13] R. Mullot, C. Olivier, J.L. Bourdon, P. Courtellemont, J.Labiche, and Y. Lecourtier, "Automatic extraction methods of container identity number and registration plates of cars," Int. Conference Industrial Electronics, Control, Instrumentation, 1991, vol. 2591, pp. 1739-1744.
- [14] Y. Watanabe, Y. Okada, Y.B. Kim and T. Takeda, "Translation camera," International Conference on Image Processing, 1998, pp. 613-617.

- [15] J. Yang, W. Yang, M. Denecke and A. Waibel, "Smart sight: A tourist assistant system," Int. Symp. Wearable Computers, 1999, pp. 73-78.
- [16] Y.W. Lim and S.U. Lee, "On the color image segmentation algorithm based on the thresholding and the fuzzy c-means techniques," Pattern Recognition, 1990, vol. 23, no. 9, pp. 935-952.
- [17] D.A. Forsyth, J. Ponce, Computer Vision A Modern Approach, Prentice Hall, 2003

致 谢

四年的大学生活就快走入尾声，我们的校园生活就要划上句号，心中是无尽的难舍与眷恋。从这里走出，意味着将是踏上人生的另一个新征程，意味着我将要把所学的知识应用到实际工作中去，感谢大学四年的学习时光，让我更自信的踏上人生征途。

本篇毕业设计课题的完成，首先要感谢指导老师华钢对我悉心的教导，感谢他在我尝试完成本课程设计期间提供的有利信息，这些信息都是实现本课题时不可或缺的，给我顺利完成本课题的设计带来了很大的便捷。其次感谢学校里的所有教诲我的老师，是他们教学成果让我对生活充满正能量让我更进一步的看清自己，他们悉心的教诲也是我完成此篇论文不可缺少的存在。感谢答辩老师对我进行现场指导。再一次谢谢学校的这些可亲可敬的老师！谢谢！