附录1：（封面、封底用120克白色铜版纸打印，无须彩打。所有红字在打印之前必须删除）



**本科生毕业设计[论文]**

（华文中宋小初号加粗居中）

[**基于机器视觉的条形码识别算法设计**](http://202.114.18.137:8080/teacher/ViewGraduation.jsp?NewsID=9622)

院 系\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

专业班级\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

姓 名\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

学 号\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

指导教师\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

年 月 日（华文中宋3号居中）

**学位论文原创性声明**

（黑体小2号加粗居中）

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包括任何其他个人或集体已经发表或撰写的成果作品。本人完全意识到本声明的法律后果由本人承担。

（宋体小4号）

作者签名： 年 月 日

**学位论文版权使用授权书**

（黑体小2号加粗居中）

本学位论文作者完全了解学校有关保障、使用学位论文的规定，同意学校保留并向有关学位论文管理部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权省级优秀学士论文评选机构将本学位论文的全部或部分内容编入有关数据进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本学位论文属于 1、保密囗，在 年解密后适用本授权书

2、不保密囗 。

（请在以上相应方框内打“√”）

（宋体小4号）

作者签名： 年 月 日

导师签名： 年 月 日

(注：此页内容装订在论文扉页)

**摘□□要** （黑体小2号加粗居中）

××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××。

××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××。

（宋体小4号，行间距固定1.5倍行距，字符间距为标准）

**关键词：**××××；××××；××××；××××

（黑体4号加粗） （宋体小4号）

**Abstract**（Time New Roman小2号加粗居中）

××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××.

××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××××.

（Time New Roman小4号，行间距固定1.5倍行距，字符间距为标准）

**Key Words：**××××; ××××; ××××; ××××

（Time New Roman 4号加粗） （Time New Roman小4号）

**目 录**

**摘要** Ⅰ

**Abstract** Ⅱ

**1 绪论** 1

1.1 课题背景与意义 1

1.2 条码识别算法的国内外研究 3

1.2.1 国外研究现状 7

1.2.2 国内研究现状 7

1.2.3 研究总结 7

1.3 EAN条形码编码规则介绍 3

2 **课题内容** 20

2.1 课题设计目标 20

2.2 条形码识别方案 20

2.3 条形码识别关键技术理论 23

2.3.1 国外研究现状 7

2.3.2 国外研究现状 7

2.3.3 国外研究现状 7

2.4 算法实践平台介绍 23

3 **算法实现** 20

3.1 图像预处理 20

3.2 条码区域提取 23

3.3 灰度域识别 30

3.4 霍夫域识别 30

3.5 基于霍夫域的多重解码策略 30

**4 算法实践及效果展示** 40

**5 结论及展望** 40

**致谢** 42

**参考文献** 44

**附录□××××××** 45

（章为宋体小4号加粗，其余宋体小4号，字母、阿拉伯数字为Time New Roman小4号）

**1 绪论**

**1.1 课题背景与意义**

本课题来源于日常生活中，传统的一维条形码（如图1.1）识别技术面临的困境。一维条形码是一段特定信息的图形标识符，它按照预先规定的编码规则，通过不相等宽度或者间隔的多个白条和黑条排列而成[1]。这项技术起源于20世纪 40年代，在70年代被应用，而在80年代被广泛普及。作为一种自动识别技术，它因为具有输入速度快、准确度高、成本低、可靠性强等优点，在当今的自动识别技术中占有重要的地位。同时，随着计算机技术和生产实践的发展与结合，条形码技术条码技术被广泛应用于商业[2]、仓储和工业生产过程等实际应用[1]。即便与之后的二维码、RFID标签等自动识别方式相比，条形码也有着价格低廉，适应现有货品包装、信息录入速度快这些特点[3]，因此研究条形码的识别技术在现实生活生产中都具有非常实际的现实意义。

一维条形码的主要识别方式分为两种，光电条码识别技术[4]以及图像式的条码识别技术。在早期使用中，人们一般使用光电条码识别技术，依靠条码扫描器（如图1.2）来识别一维条形码，条码的解码和识别录入是通过人工操作使得激光扫描器对准和靠近条形码来完成的。这种方式技术成熟并且不易受外界环境干扰，但读取环境要求苛刻使这项技术在很多场合不能满足人们的需求，比如必须靠近到足够距离，同时只能扫描到少量的条码，而且不能克服条码本身的污损，褶皱（如图1.3）等。复杂的应用场景要求我们探寻更加具有适应性的技术方式来识别录入条码。

第二种图像式的条码识别技术[5]则很有希望完成一些复杂场景下的一维条码识别。图像式的条码识别技术是通过图像采集装置（如图1.4）采集条码图像, 然后利用图像处理技术进行条码图像的去噪和分割处理, 最后调用读码模块, 读出条码值并显示识别结果。相比于传统的光电条码识别技术，利用采集图像识别具有两大优势，一是随着生产工艺的精进和人们消费能力的升级，镜头的分辨率不断提高，移动设备的性能日益强大，社会呈现带有镜头的移动设备越来越普及并且成本不断减低的趋势，这使得基于机器视觉的一维条码识别在硬件上有了现实使用的条件和基础，不管是图像采集还是算法运算。二是从信息量上来说，图像采集的信息远远多于光电识别技术，那么从算法层面上来说，能够进行的算法处理可以更加丰富和多样。综上所述，基于机器视觉的一维条码识别技术具有在复杂情况下识别条码的深远潜力。



图1.2 条码扫描器

图1.1 一维条形码样例





图1.4 一维条形码的污损、褶皱、变形

图1.3 图像采集装置

而本课题的目标正是基于机器视觉，研究超市中一维条码识别的算法。对于一维条码的解码，国内外的研究人员已经提出了很多理论，并做了相应的尝试，但多数研究仅仅针对某一特定的问题，没有结合特定使用场景下的条码识别做出对应的优化。本课题将应用场景确定为超市购物结账，并将结合多种算法来综合处理超市中一维条码的识别问题，符合实际应用需求，使其在具有相当的解码速度的同时可以应对条码的弯曲，褶皱和污损等问题，完成在复杂情况下的识别。

本课题的研究旨在提高如超市这样特定环境下条形码的识别速度和识别准确率，减少营业员扫码错误带来的时间损耗，提高超市运行效率。并且，这项技术的延伸研究也将有助于完善货物商品信息的有效提取和录入，帮助无人超市，无人零售等概念的实现，为中国物联网建设作出技术探索。

**1.2 条码识别算法的国内外研究**

**1.2.1 国外研究现状**

一维条形码作为一项在上世纪80年代普及的技术，在国内外都有许多研究人员做了卓有成效的工作。基于机器视觉，对于一维条码的解码工作主要是解决两部分问题，一是找到条码区域[6]，二是将区域中的条码解码[7]。

针对条码区域的检测，各个研究组采取的方法大同小异，主要利用的是条码区域的白色边缘以及大量黑白条纹产生的方向一致的边缘信息。所以条形码识别问题的重点主要是针对条码区域的准确快速解码，各个研究组这个过程中尝试的方法差别就比较大。

1993年4月，E. Joseph和T. Pavlidis 在IEEE期刊上发表了《Deblurring of bilevel waveforms》，该文献[8]提出了一种处理密集边缘并精确恢复其位置的算法。利用基于相互作用的边缘的卷积失真模型，以三条边为输入，强制两条边的作用相互抵消，使得第三个边缘不受噪声干扰，并使用传统的边缘探测器定位。实验表明，该方法优于当时的商业条码阅读器。

2007年，Sherin M. Youssef等人在《Expert Systems with Applications》上发表了《Automated barcode recognition for smart identification and inspection automation》，该文献[9]提出了一种基于快速分层Hough变换（HHT）的智能条码检测识别系统（SBDR）。他们利用一维条形码周边的白色区域来进行区域提取，并进行轮廓跟踪来分割出更准确的条码矩形区域，最后使用分层霍夫变换（HHT）确定条码区域的倾斜角度，由此，他们准确定位了条码区域。在解码条形码的过程中，他们首先改善条形码区域，然后提取了条码区域的特征。最后，训练反向传播神经网络用于执行条形码识别任务。经过测试，他们证明与激光条形码阅读器（每英寸约15,000点）相比，即使镜头分辨率更低，他们的方法也可以在各种文档图像上定位和解码条形码，并且处理损坏的条形码。

2011年，新加坡团队杨慧娟等人提出了一种新的图像二值化方案。文献[6]提出条形码和背景区域通过搜索窗口中的边缘像素的数量来区分。与现有方法——用固定大小的窗口二值化的像素居中不同，他们将窗口中心移动到最近的边缘像素，以便可以实现对象和背景像素的数量的平衡。窗口大小会适应条形码中最小边缘距离或最小元素宽度。他们使用窗口中的统计信息计算阈值，并证明了其处理非均匀照明和物体尺寸变化情况下条形码的能力。

2012年1月25日，美国加利福尼亚州伯灵格姆研究团队Dwinell, J.等人在SPIE-国际光学工程学会会议上发表了基于霍夫变换的一维条码鲁棒识别，文献[10]提出多层次的方法提供了一个更好的办法来最好地管理竞争限制的复杂的噪音和快速解码。在最低级别，图像以灰度级处理。在中间层，图像被转换为霍夫域。在顶层，全局结果（包括缺失信息）在全局上下文中处理，包括域启发式和OCR。这三个级别通过在级别之间上下传递信息而紧密协作。该算法解决了各种典型的图像失真，例如不均匀的照明，反射，损坏的条形码或模糊等，有效针对了大包裹处理系统。

近年来，针对条形码识别原理上的研究不多，主要是对于各种特殊情况中的条形码，如何处理图像来更快更准确地获得条码信息。这些研究团队也逐步在研究很多实际应用中对于条形码识别的研究。

**1.2.2 国内研究现状**

相比之下，国内基于机器视觉关于一维条形码识别的研究比较晚，但发展迅速，并且与时俱进，研究的算法主要针对生活中的一些具体场景贴合实际，满足产业需求，结合了机器学习[11]等新概念来改进传统方法，也达到了一些不错的效果。

2006年，合肥工业大学的王烨青、杨永跃发表了《机器视觉在流水线条形码识别中的应用》，该文献[12]提出基于机器视觉的条码自动识别技术,利用高速CCD摄像机得到条码的图像,通过几何变换,滤波去噪,阈值处理等有效的图像处理和快速模式识别方法,结合优化设计的条码码制数据库实现了流水线物品条码的快速、精确识读。

2010年，华南理工大学的吴忻生、邓军在《计算机应用》上发表了《利用提取边缘线的方法定位条形码》，该文献[13]提出首先灰度化原始图像，通过大津法获得阈值，然后隔行隔列处理灰度图中小于该阈值的像素，以此提取边缘线。之后分割图像为相同大小的水平条，根据质心位置和各个水平条中边缘点的分布来定位起始条，最后从边缘点拟合的直线来确定条码倾斜的角度，从而提取出条形码区域。而对于条形码的解码，他们没有使用传统的脉冲测量法，而是采用了相似边距离测量法,通过测量条码符号中相邻元素的相似边之间距离来判断字符的逻辑值。

2015年，浙江大学陈俊名发表了《复杂背景下一维条形码定位算法研究》，该文献[14]提出的一维条形码定位算法在普通PC上对常用DPI的图像能够实现毫秒级的定位并且对定位区域具有较高的准确率和完整率。同时,该算法对包含复杂背景的条码图像在条码倾斜率,几何形变,条形码位置,条形码区域占整个图片的比例以及条码数量等因素不敏感,具有较高的算法鲁棒性。利用其算法将可以大大加快在复杂背景下条形码的整个解码速度并提高最终的解码成功率,具有良好的实际应用前景。

国内的这些研究主要是在一些传统的识别理论上进行一些细节处的改良，提高现有算法的识别准确率和效率。

**1.2.3 研究总结**

作为一种成熟的自动识别技术，条形码已经广泛应用于人们的日产生活中，在大型超市、仓库部门、员工卡片等等地方都发挥了非常重要的作用。关于条形码的识别理论本身，从光学探测到的波形到传统的图像处理方式，再到使用机器学习等模型辅助，研究者们已经进行了有相当广度和深度的研究，但在现实使用的情景中，人们对于条形码的识别还是不得不面对无法处理条形码的褶皱，磨损，褪色等问题。所以，目前人们还在对于很多具体的现实条形码识别场景进行研究，力图找到合适的识别方案[15]。例如上述的，针对大包裹处理系统的多级霍夫识别算法，针对流水线物品条码的快速识别算法，针对复杂背景下的鲁棒算法等，现实的识别需求推动了识别算法向变得更加灵活来适应复杂的情况。

在这些解码的研究中，最被关注是算法的准确性和实时性，这两者决定了算法在实际应用中的价值。但这两者一般也具有天然的矛盾，在复杂情况中保持准确需要更多的处理过程，这导致了更多的时间被消耗。如何平衡识别精度和时间也将是识别过程中必须要考虑的问题。

**1.3 EAN条形码编码规则介绍**

由于本课题的工作目标是日常生活中，所以以被广泛使用的EAN码作为课题中条形码识别的目标。EAN13商品码主要可以分为左侧空白区、起始符、左侧数据符、中间分隔符、右侧数据符、校验符、终止符、右侧空白区及供识别字符。EAN13的构成示意图如下图（1.5）所示：



图 1.5

常用的EAN一般是13位或者8位，分别被称为EAN13和EAN8，其中最后一位是由之前数字计算出的校验码。本课题的识别对象是EAN13，EAN-13码由**13个数字**组成，其中这13个数字又分为4组：国家代码（3位）、厂商代码（4位）、产品代码（5位）和检查码（1位）。

国家代码：是国际EAN组织标识各会员组织的代码，我国为690、691和692。其中第一位（即6）为前置码，规定了左侧数据符的编码规则。而且该位是不用条码符号表示的。其条码编码规则如下表（1.1）、（1.2）所示，0为空，1为条。

厂商代码：EAN编码组织在EAN分配的国家代码的基础上分配给厂商的代码。

产品代码：由厂商自行编码，在编制产品代码时，厂商必须遵守产品编码的基本原则：对同一商品项目的商品必须编制相同的商品项目代码；对不同的商品项目必须编制不同的商品项目代码。保证商品项目与其标识代码一一对应，即一个商品项目只有一个代码，一个代码只标识一个商品项目。

校验码：为了校验代码的正确性，由前面的12位数字计算得出。

|  |  |
| --- | --- |
| 前置码 | 左侧数据符的编码规则 |
| 0 | AAAAAA |
| 1 | AABABB |
| 2 | AABBAB |
| 3 | AABBBA |
| 4 | ABAABB |
| 5 | ABBAAB |
| 6（中国） | ABBBAA |
| 7 | ABABAB |
| 8 | ABABBA |
| 9 | ABBABA |

表（1.1）

|  |  |  |  |
| --- | --- | --- | --- |
| 数字符 | 左侧数据符 | | 右侧数据符 |
| A | B | C |
| 0 | 0001101 | 0100111 | 1110010 |
| 1 | 0011001 | 0110011 | 1100110 |
| 2 | 0010011 | 0011011 | 1101100 |
| 3 | 0111101 | 0100001 | 1000010 |
| 4 | 0100011 | 0011101 | 1011100 |
| 5 | 0110001 | 0111001 | 1001110 |
| 6 | 0101111 | 0000101 | 1010000 |
| 7 | 0111011 | 0010001 | 1000100 |
| 8 | 0110111 | 0001001 | 1001000 |
| 9 | 0001011 | 0010111 | 1110100 |

表（1.2）

EAN-13码是按照“模块组合法”进行编码的，本课题识别的条码一共113个模块（如下表（1.3）），每个模块的宽度为0.33mm 。EAN标准码的尺寸，宽：全部37.29mm（113\*0.33mm） 条码31.35mm（95\*0.33mm） 长：数据符条码22.85/23.18mm， 起始符/分隔符/终止符24.50mm 全部26.26mm放大倍数：0.8 至2 倍。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **左侧空白区** | **起始符** | **左侧数据符** | **中间分隔符** | **右侧数据符** | **校验符** | **终止符** | **右侧空白区** |
| **9个模块** | **3个模块** | **42个模块（6）** | **5个模块** | **35个模块（5）** | **7个模块（1）** | **3个模块** | **9个模块** |

表（1.3）

本课题的识别目标正是将条形码还原成113个模块，然后对其内容进行解码，从而获得条形码对应EAN13码的13位数字。

**2 课题内容**

**2.1 课题设计目标**

如前文所述，目前条形码的课题研究主要与实际需求相结合，力求结合实际情况设计灵活算法平衡条形码识别的精度与速度。而本课题选取的实际场景正是人们平常生活中经常遇到的超市收银扫码。日常生活中，超市中常用的激光式解码器常常因为距离条码远、条码褶皱、条码破损等问题不能及时解码，给人们生活带来了极大地不便。

本设计的目标是设计一款算法超市场景中基于机器视觉完成对于条形码的识别，弥补激光式扫码器识别的缺陷。设计的算法能够在正常忍受时间内识别出视野中的条形码，并对于污损、褶皱、油墨不清的情况作出同样鲁棒性的识别。

功能上要求：（1）能够快速识别清晰的条形码。

(2) 对于复杂情况的条形码，可以准确识别。

(3）对于视野中少量的条形码可以同时识别。

性能上要求：(1）识别距离可以达到40cm。

(2) 识别速度平均可以达到200ms一次，对于清晰条码需要做 到100ms一次。

设计的算法各模块需要功能明确，较少依赖于其他库。设置的参数需要有普适性，能够自行对摄取的图像进行分类，从而适应各种情况下的条码识别。

**2.2 条形码识别算法方案**

为了达到对于条形码识别准确性和效率性的兼顾，本课题将采用多级算法来解码不同识别难度的条形码。对于清晰可辨的条形码，算法会停留在灰度域进行处理，而对于有一定程度变形或者损毁条形码的识别则会在霍夫域进行。整体的算法流程如下图（图2.1）所示：



图2.1 条形码识别算法流程示意图

如上图所示，在本课题设计的算法中，主要分为图像预处理、条形码区域提取、灰度域直接识别、转换到霍夫域识别以及对应霍夫域的二叉树多重解码五个部分。其中，图像预处理部分主要分为两部分，第一部分是为提取条形码区域做准备，确保提取区域算法在处理多种类型图像时的稳定与一致。而第二部分的预处理，则是对于已经提取出来的条形码区域图像，同理，预处理是为了后续解码部分算法的稳定和一致。

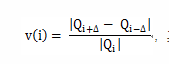
本课题设计的算法核心是条形码区域提取部分、灰度域解码、霍夫域解码。条形码区域提取的工作是是将条形码候选区域提取，便于后续的解码操作，在第一次提取之后也会有对提取区域尺寸、数量的一个判断，出现异常即会自动调整阈值再次提取。在解码部分，针对不同情况的条形码，将会有两个层次的识别手段。首先会直接进行灰度域的解码，当条形码处于比较清晰的情况，简单的二值化就可以提取出条形码黑白条纹的间距，从而解码。如果条形码在比较复杂的情况中，图像质量不高，则需要将原本的灰度图转换到霍夫域处理更多的细节信息，同时利用二叉树对同一副图像模糊的地方进行多重判断，以此提高识别率。利用这两个不同层次的识别，可以将简单图像和复杂图像的处理分流，平衡条形码识别算法的精度与速度。

**2.3 条形码识别关键技术理论**

**2.3.1 条形码区域提取——MSER（最大极值稳定区域匹配）**

MSER使用不同的灰度阈值来进行图像的二值化，它的出现是基于分水岭的概念：对图像进行二值化，二值化阈值取[0, 255]，这样二值化图像就经历一个从全黑到全白的过程（就像水位不断上升的俯瞰图）。在这个过程中，有些连通区域面积随阈值上升的变化很小，这种区域就叫MSER。

 检测连通区域的面积变化快慢可以用公式variation表示：



其中Qi表示第i个连通区域的面积，Δ表示微小的阈值变化（注水），当vi小于给定阈值时认为该区域为MSER。

算完后，每个区域都会有一个自己的variation值，在opencv中，只要某个ER区的variation值小于0.27，就认为这个ER区是稳定的，那么这个函数就会被OpenCV中的MSER保存到regions中返回；

显然，这样检测得到的MSER内部灰度值是小于边界的，想象一副黑色背景白色区域的图片，显然这个区域是检测不到的。因此对原图进行一次MSER检测后需要将其反转，再做一次MSER检测，两次操作又称MSER+和MSER-。将这两次操作求交集，就可以找到图像中像条码这样强对比黑白条纹密集的区域。

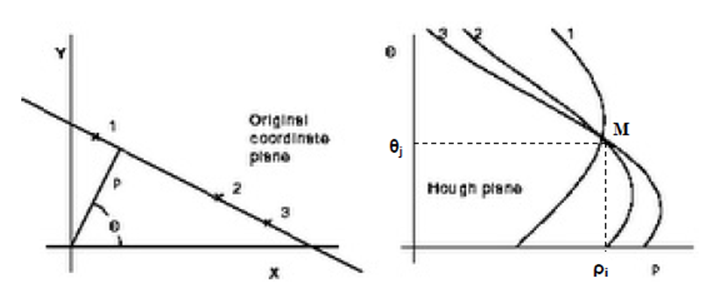
MSER能有效检测仿射不变性区域，它的仿射变化对于图像灰度具有不变性，对于区域相对的灰度变化也具有稳定性，同时它对于图像不同精细程度的大大小小的区域都能检测。MSER通过局部阈值集的操作，使得最大及稳定区域的像素变化是最小的，从而可以用来帮助定位黑白交替的条形码区域。

**2.3.2 复杂情况条形码识别——霍夫域变换**

对于识别条件较差的条码识别的准确性与速度。本课题选择在霍夫变换层次解码，因为条形码由粗细不同的黑白条码组合而成，而霍夫变换对于线的位置以及角度十分敏感，可以较好地展示出线与线之间位置的关系，便于进一步提取黑白条纹的间距。其中要使用的直线霍夫方程为：

（公式4.1）

其中，ρ是距原点的距离，θ是垂直于该线的角度。由公式可知，图像中像素的可从笛卡尔系转换到与直线直接相关的距离和角度。对多个点做霍夫变换可得下图：

****

在条形码的识别中，霍夫变换将有利于将质量不高的像素点汇总，上图的M点的密度显示出了这三个点最可能组成的直线的角度和距离。而对于完整条形码的识别，霍夫变换则将直线的质量通过这种交汇点反映出来，便于算法判断这些质量不高的像素点构成的直线之间的位置关系。在右图中，M点密度为三，则表示了有三个点在这个方向θ和这个距离ρ的直线上。而在条形码中使用霍夫变换会得到下面这样的结果。

面对条形码的褶皱、扭曲、污损，算法在霍夫域中仍能找到大部分的条码线条信息，而这些大量未被破坏的条码部分将在霍夫域中的对应（θ，ρ）中显示，通过查看（θ，ρ）的密度，条形码的真实黑白间隔也就可以被确定。

霍夫域转换的问题在于，如果对于每个θ，每个ρ，每个像素点都做霍夫变换，那么对于计算机而言会是一个巨大的负荷。因此，减少计算量才能使霍夫域转换的应用具有可行性。在本课题中，只有提取出的条形码区域才会进行霍夫变换，并且仅仅针对之前就提取出的条形码方向，以此减少大量的计算，提升算法的效率。

**2.3.3 基于霍夫域的多重解码策略——二叉树**

面对复杂情况的条形码识别，有时条形码的黑白条纹的宽度会和标准宽度相差甚远，死板的宽度阈值设置或者仅仅与相邻的条码比对宽度都不能有效识别条形码信息。所以，本课题对于这种情况，依靠霍夫变换的特性，对于宽度信息不明朗的条形码部分采取多重判断的方式。他的判断流程如下图所示：

由上图可知，每一个不明朗的宽度信息都会对应多个结果，并判断到最后。这大大提升了条形码被识别出的概率。如果在判断过程中，条形码获得了多个结果，那么算法将会计算筛选出其中最接近阈值的那个结果。

这种解码方式的可行依托于两点。一是霍夫域中的条形码信息提供了多重解码的指导，让解码流程的分叉不再只是猜测，更是对于现有数据的一种情况判断。二是条形码本身的编码形式就具有校验性，如果中间置信了某个错误的结果，那么在接下来每个数字的字节以及分隔符都会无法对应，所以错误的结果难以保留到最后。

对于复杂情况中的条形码，识别情况本身确实不好，但是通过基于霍夫域的多重二叉树判断，所有可能的结果都会被保留出来，再通过条形码的校验筛选出最后的结果。并且这个过程只是增加解码的结果，是基于之前对图像的处理的，所以计算量上并没有增加。综上，基于霍夫变换的二叉树多重解码策略有效提升了算法的识别准确率。

**2.4 算法实践平台介绍**

本算法的实践平台如下表所示：

|  |  |
| --- | --- |
| 配置种类 | 具体配置 |
| IDE | Visual Studio 2015 |
| 用到的库 | Opencv 3.4.0 |
| 电脑系统 | Windows 10 |
| CPU | i7 |

算法整体通过C++编写而成，在一些基本操作上使用了Opencv的库函数，而算法的核心霍夫域变换、二叉树多重判断等全部是自行编写的。算法采用的阈值具有自适应性，适用于预处理后的条形码图像，并保证输出结果的稳定和一致。

**3算法实现**

**3.1 图像预处理部分**

为了让算法对于采集的图像有准确稳定的输出结果，我们需要对于采集的图像数据进行预处理，一方面统一对于算法输入图像的大小格式一致，另一方面则对于原图像的特征加强或转变，便于进一步提取条码区域或者解码。基于这两个目标，本课题设计了图像预处理部分的算法。

**3.1.1 第一部分图像预处理——归一化图像大小**

对于第一部分整体图像的预处理流程设计，如下图（3.1）所示：

如上图所示，第一部分的预处理的主要工作是归一化获取图像的大小，并灰度化。在原图像特征上并没有加强，这是因为在提取条形码区域的部分采用了MSER技术，如上文所述，它对于输入图像已经做了较好的处理，所以第一部分预处理的条形码特征加强可以省去。

而主要工作其中归一化大小使用的是基于区域子块提取图像缩放。一般的图像缩放基于等间隔提取，f(x,y)转换到g(x’,y’)的矩阵映射关系如下：

这样的处理方式比较简单快速，但是很多细节特征在新图像中不会保留，使得图像容易变得模糊，同时，如果不与相等的话，这样的均匀采样会造成在宽度和高度上缩放的程度的不同，从而使得图像扭曲。

而相比之下，基于区域子块提取图像缩放，则更具有普适性，它的提取方式可以表达为：

其中M,N分别表示原图像的高度和宽度，而m,n则表示缩放的比例。大括号内的像素值便是提取出的字块，通过对于这些子块求取均值，获得的缩放后图像的像素，这样的缩放方式保留了部分图像中像素变化的信息，并保证了像素变化曲线的平滑。

**3.1.2 第二部分图像预处理——归一化图像大小+矫正条形码旋转角度+OTSU二值化**

对于第二部分条形码区域图像的预处理流程设计，如下图（3.2）所示：

如上图所示，第二部分的预处理的主要工作是归一化获取图像的大小，并矫正它的旋转角度，再通过OTSU对候选区域进行二值化操作。其中归一化大小的操作与第一部分一致，不作赘述。操作中没有滤波、膨胀腐蚀，因为处理清晰的图像不需要这些操作，而复杂的情况则会在霍夫域中得到处理，这些滤波、膨胀腐蚀的操作作用也很小。

关于提取的条码图像的角度矫正，则是利用傅里叶变换中时域和频域的关系获得旋转角度，再通过仿射变换获得矫正的图像。傅里叶变换中，图像的低频部分集中在边缘角，而高频部分在图像中心，但需要的获取的是条码的方向，所以需要将高低频部分交换中心，从而进行对于旋转角度的检测。

为了在灰度域中能够解码，被提取的条形码区域需要被提取他们的黑白条纹间距信息。这里可以采取二值化操作或者边缘检测。本课题中采用的方法是将原图像划分为若干区域，分别进行OTSU二值化操作。

常用的阈值化处理如下表（）所示：

|  |  |
| --- | --- |
| 常用检测方法 | 检测特点 |
| 固定阈值化 | 阈值固定 |
| 自适应阈值化 | 阈值随周围领域块变化而变化， |
| 双阈值化 | 适用于有明显双分界特征的图像 |
| 半阈值化 | 适用于对于字符区域分割 |
| OTSU阈值化 | 阈值为图像阈值的最大类间方差 |

它们的处理结果如下图所示：

分析上图可知，边缘检测，二值化则

而通过将原图像划分为若干区域，分别进行OTSU二值化操作则更进一步灵活调整了图像二值化的阈值，并且较好保留了提取的条形码区域的黑白条纹间距信息。它的二值化效果如下图所示：

**3.2 提取条形码区域**

常见的方法是用一维条形码周边的白色区域来进行区域提取，并进行轮廓跟踪来分割出更准确的条码矩形区域，最后使用分层霍夫变换（HHT）确定条码区域的倾斜角度。但是这种方法需要的条码区域中常常有别的字符，导致方块内的边缘较多且杂乱无序，会影响最初的提取。

所以相对来说，通过搜索图片中的边缘像素的数量来区分条形码和背景区域是个更好的选择。通过对边缘的检测，找到条形码的候选区域，通过针对性的膨胀和腐蚀和对于条码方向的判断来进一步确认条码的区域，并将之提取出。

常用的边缘检测法处理如下表（）所示：

|  |  |
| --- | --- |
| 常用检测方法 | 检测特点 |
| Sobel检测算子 | 效率较高，对于和X轴、Y轴同方向的像素点排列较好，其他方向的边缘不够精确 |
| Laplace检测算子 | 对于奇异点和边缘点敏感，一般用于锐化图像 |
| Roberts检测算子 | 垂直检测精度高，但抑制噪声能力弱 |
| Prewitt检测算子 | 可以抑制噪声，定位精度不如Roberts和Sobel算子 |
| Canny检测算子 | 错误率低、定位性高、要求响应小 |
| Marr-Hildreth检测算子 | 定位精度高，能抑制噪声 |

对于图像进行边缘检测的结果如下图（）所示：

总体而言，边缘检测能够将黑白条纹的条形码边缘检测出，但是并不能把它与背景中的边缘有效分辨出，因为常规的边缘检测不能反映出对比度的强烈，而这恰恰是条形码与其他区域边缘最大的区别。

而本文采取的方法就是利用对于灰度图进行MSER+和MSER-的操作，再将两个结果进行与操作，从而获得图像中对比度最明显的区域，也就是黑白条码对比最密集的区域。最后通过最小轮廓提取获得条码的区域。提取的结果如下图所示：

提取出了对应的条形码区域以后，算法通过轮廓检测不断逼近目标的边缘，最后得出最小的轮廓矩形，再对它的长度、宽度以及长宽比进行一次筛选，筛选出合适的轮廓区域，也就是需要提取出的条形码区域。在这里我们采用矩形在原图像中截取图像，但在后续操作中会转化为旋转矩阵，便于之后解码前的旋转矩阵矫正的进行。截取的条形码区域如下图（）所示：

这个环节的工作仅仅是提取出条形码区域，最后这个旋转矩阵会在原图中截取原始的条形码区域，再输入到下一个环节进行解码。

**3.3 灰度域中解码**

对于清晰的条形码图像，它的边缘信息明显而完整，可以直接在灰度域中处理。本课题中，即是在OTSU二值化的基础上进行解码。下图（）是解码的流程：

如上图所示，解码的流程主要包括两个转换，一是从黑白图像转换到关于黑白间距的存储结构，二是将黑白间距转换到0、1的各个条形码模块组合。完成这个两个转换之后，就是单纯对0、1序列的解码。解码的具体判断流程如下图（）所示：

下图（）是一个清晰条形码的识别案例。它被归一化一个固定大小后，会被分为若干部分，分别进行OTSU阈值化，然后直接取吉个横坐标进行黑白间距信息的获取，再转化为0、1,复原识别的条形码图像并进行校验和解码。

传统方法通过对原图像黑白条码宽度进行脉冲测量法，但是这种方法处理量大，而且脉冲测量容易受印刷质量影响下条码标准宽度不准的影响。

所以，本课题选择将条形码区域转换到霍夫平面，通过条码的方向信息找到霍夫平面对应条码的结果行，这样计算量会减少很多。黑白色条码的间隔将会被计算出，再利用相似边距离测量法,通过测量条码符号中相邻元素的相似边之间距离来判断字符的逻辑值。本课题是通过每个黑白段的前面的两组黑白段确定的单位长度来丈量当前黑白区域的相对长度。

在这里条形码的解码选择直接OTSU阈值化逐行扫描记录黑白条纹间距，而不是通过边缘算子检测边缘来测算黑白区域的边缘再来测算间距。因为在边缘检测的过程中，容易因为微小的变化，比如涂抹、褪色而出现意料之外的检测边缘，会对条形码识别当前像素是黑是白造成干扰，容易造成错位。

**3.4 霍夫域中解码**

它们的处理结果如下图所示：

常见的方法是用一维条形码周边的白色区域来进行区域提取，并进行轮廓跟踪来分割出更准确的条码矩形区域，最后使用分层霍夫变换（HHT）确定条码区域

**3.5 基于霍夫域的多重解码策略**

解码部分策略如下：



传统方法通过对原图像黑白条码宽度进行脉冲测量法，但是这种方法处理量大，而且脉冲测量容易受印刷质量影响下条码标准宽度不准的影响。

所以，本课题选择将条形码区域转换到霍夫平面，通过条码的方向信息找到霍夫平面对应条码的结果行，这样计算量会减少很多。黑白色条码的间隔将会被计算出，再利用相似边距离测量法,通过测量条码符号中相邻元素的相似边之间距离来判断字符的逻辑值。

**4算法实践及效果展示**

**4.1 简单情况下条形码识别的精度与速度**

1）：

**4.2 复杂情况下条形码识别的精度与速度**

**4.2.1 污损情况**

**4.2.2 褶皱情况**

**4.2.3 扭曲情况**

**4.2.4 反光情况**

传统方法通过对原图像黑白条码宽度进行脉冲测量法，但是这种方法处理量

**4.3 解码策略部分实践**（

××××××××××××××××××××××××××××××××××××××××××××××××，其×××××可表示如下：

 (3-1）

 (3-2）

×××××××××××××××××××××××××××× (如表3-1所示)

表3-1□××××××××××

|  |  |  |  |
| --- | --- | --- | --- |
| ××××× | ××× | ××× | ××× |
| ××××× | ×× | ×× | ×× |
| ××××× | ×× | ×× | ×× |
| ××××× | ×× | ×× | ×× |
| ××××× | ×× | ×× | ×× |

（表标题：位于表格上方，黑体小4号，字母、阿拉伯数字为Time New Roman小4号，表内容：宋体5号，字母、阿拉伯数字为Time New Roman 5号）

××××××××××××××××××××××××××× (如图3-1所示)



图3-1□××××××××××

（图标题：位于图下方，黑体小4号，字母、阿拉伯数字为Time New Roman小4号）

.......

--------章与章之间插入分页符----------

**参考文献** (黑体小2号加粗居中)

[1]□王静康,张凤宝,夏淑倩等.论化工本科专业国际认证与国内认证的“实质性”.高等工程教育研究,2014,5:1-4

[2]□Stone J A, Howard L P. A simple technique for observing periodic nonlinearities in Michelson interferometers. Precision Engineering,1998,22(4):220-232

[3]□朱印红,袁衍明.Dreamweaver完美网页设计——技术入门篇.(第一版).北京:中国电力出版社,2006:19～20

[4]□Lewis S L. Physics and chemistry of the solar system.北京:北京大学出版社,2014.1～2

[5]□陈剑.上博简《民之父母》“而得既塞於四海矣”句解释[EB/OL］.简帛研究网站，http://www.bamboosilk.org/Wssf/2003/chenjian03.htm．2003-01-18

( 宋体小4号)

……

……

……

**附录** (黑体小2号加粗居中)

×××××××××××××××××××××××××××

( 宋体小4号)

……

……

……