

# 文本复制检测报告单(全文标明引文)

№:ADBD2019R\_20190604091611449822311355

检测时间:2019-06-04 09:16:11

检测文献: 基于机器视觉的智能裁布机控制系统

作者: 朱海鹏

检测范围: 中国学术期刊网络出版总库

中国博士学位论文全文数据库/中国优秀硕士学位论文全文数据库

中国重要会议论文全文数据库

中国重要报纸全文数据库

中国专利全文数据库

图书资源

优先出版文献库

大学生论文联合比对库

互联网资源(包含贴吧等论坛资源)

英文数据库(涵盖期刊、博硕、会议的英文数据以及德国Springer、英国Taylor&Francis 期刊数据库等)

港澳台学术文献库

互联网文档资源

CNKI大成编客-原创作品库

个人比对库

时间范围: 1900-01-01至2019-06-04

## 检测结果

去除本人已发表文献复制比: 4.8%

跨语言检测结果: 0%

去除引用文献复制比: 4.8%

总文字复制比: 4.8%

单篇最大文字复制比: 0.8% (布料运送位置校正机器人视觉系统设计)

重复字数: [1892]

总字数: [39577]

单篇最大重复字数: [308]

总段落数: [4]

前部重合字数: [615]

疑似段落最大重合字数: [878]

疑似段落数: [3]

后部重合字数: [1277]

疑似段落最小重合字数: [311]

指标: ☐ 疑似剽窃观点 ☒ 疑似剽窃文字表述 ☐ 疑似自我剽窃 ☐ 疑似整体剽窃 ☐ 过度引用

表格: 0 公式: 没有公式 疑似文字的图片: 0 脚注与尾注: 0

7.4% (703) 基于机器视觉的智能裁布机控制系统\_第1部分 (总9473字)

2.8% (311) 基于机器视觉的智能裁布机控制系统\_第2部分 (总11244字)

7.5% (878) 基于机器视觉的智能裁布机控制系统\_第3部分 (总11630字)

0% (0) 基于机器视觉的智能裁布机控制系统\_第4部分 (总7230字)

(注释: 无问题部分 文字复制部分 引用部分)

## 指导教师审查结果

指导教师: 张继

审阅结果:

审阅意见: 指导老师未填写审阅意见

## 1. 基于机器视觉的智能裁布机控制系统\_第1部分

总字数: 9473

### 相似文献列表

去除本人已发表文献复制比: 7.4%(703) 文字复制比: 7.4%(703) 疑似剽窃观点: (0)

1	布料运送位置校正机器人视觉系统设计 秦永润 - 《大学生论文联合比对库》 - 2016-05-18	3.3% (308) 是否引证: 否
2	IC探针图像模式识别与测距技术研究 黄传霞(导师: 赵勋杰) - 《苏州大学博士论文》 - 2013-05-01	2.0% (188) 是否引证: 否
3	陶瓷阀芯表面缺陷视觉检测方法的研究 何小凡(导师: 钟毓宁;汪威) - 《湖北工业大学博士论文》 - 2018-05-25	1.0% (96) 是否引证: 否

4	基于机器视觉的弹体轮廓在线检测技术的研究	0.7% ( 66 )
崔洪岩(导师：王春阳) - 《长春理工大学博士学位论文》 - 2011-03-01		是否引证：否
5	基于人体脉搏跳动的机器视觉可视化系统设计	0.5% ( 52 )
梅志远 - 《大学生论文联合比对库》 - 2018-06-08		是否引证：否
6	基于机器视觉的手机板参数测量	0.4% ( 37 )
李壮 - 《大学生论文联合比对库》 - 2015-05-14		是否引证：否
7	基于机器视觉的机械精度测量技术研究	0.3% ( 31 )
张思佳(导师：段振云;赵文辉) - 《沈阳工业大学博士学位论文》 - 2014-02-21		是否引证：否

原文内容

学号：15417133

常州大学

毕业设计（论文）

（2019届）

题目基于机器视觉的智能裁布机控制系统

学生朱海鹏

学院信息数理学院专业班级自动化151

校内指导教师张继专业技术职务讲师

校外指导老师专业技术职务

二〇一九年六月I

基于机器视觉的智能裁布机控制系统

摘要：激光裁剪技术不断发展，在此基础上加入机器视觉技术具有非接触、速度快、产

品质量高等优点，能够适应现代工业中大部分的成产要求。本文相对于传统的裁布机提出了一种基于机器视觉的智能裁

布机控制系统，并在现有条件下进行相关分析和研究。

首先系统通过获取 PLC 外触发信号采集图像并判断是否为待裁剪布料，然后机器停止送料，控制相机水平移动拍摄图像

，再利用 OpenCV 图像处理进行图像的匹配、拼接、边缘路径拟合，然后将拟合出的路径通过标定数据转换为实际物理路径

，最后通过 Modbus

通讯将路径发送给裁剪机构进行裁剪，并基于 MFC 做出简单、易操作的人机交互界面。

本文重点对该智能控制系统的视觉部分的硬件选型和图像算法进行了阐述，其中包括工业相机、镜头、光源的选型和介

绍，以及相机手眼标定算法、二维仿射变换、图像模板匹配、灰度直方图匹配、ROI 图像拼接、相位相关分析、边缘跟踪、边

缘路径拟合等，

并详细的介绍了图像在软件中的处理过程。

关键词：机器视觉；手眼标定；图像拼接；边缘跟踪；Modbus；MFC

II

Intelligent Cutting Machine Control System Based on Machine

Vision

Abstract: With the continuous development of laser cutting technology, machine vision technology has the advantages

of non-contact, fast speed and high product quality, and can meet most of the production requirements of modern industry.

In this paper, compared with the traditional cloth cutting machine, an intelligent cutting machine control system based on

machine vision is proposed, and the correlation analysis and research are carried out under the existing conditions. Firstly,

the system acquires the plc external trigger signal to capture the image and determines whether it is the cloth to be cut, then

the machine stops feeding, controls the camera to move horizontally to capture the image, and then uses the OpenCV

image processing to match and stitch the image. The edge path is fitted, then the fitting path is converted into a real physical

path by calibration data. Finally, the path is transmitted to the clipping mechanism through Modbus communication for

clipping, and a simple and easy-to- operate human-computer interface is made based on MFC. This paper focuses on the

hardware selection and image algorithm of the machine vision intelligent control system, including the selection and

introduction of camera, lens, light source, image acquisition card, camera hand- eye calibration algorithm, 2D-affine

transformation, image template matching, gray histogram matching, ROI image mosaic, phase correlation, edge path fitting,

etc. And the processing process of the image in the software is introduced in detail.

Key words: Machine Vision; Template Matching; Image Mosaic; Edge tracking; Modbus;

MFC

III

目录

摘要 ..... I

Abstract ..... II

第 1 章. 绪论 ..... 1

1.1 课题研究背景 ..... 1

1.2 课题研究的目的是意义 ..... 1

1.3 课题国内外研究现状 ..... 2

1.4 论文主要内容 ..... 3

第 2 章. 智能裁布机系统总体设计 .....	5
2.1 系统总体框架 .....	5
2.1.1 机器视觉系统介绍 .....	5
2.1.2 系统方案流程介绍 .....	6
2.2 系统硬件选型 .....	7
2.2.1 相机、镜头、光源介绍 .....	7
2.2.2 硬件选型 .....	8
2.3 本章小结 .....	9
第 3 章. 智能裁布机系统软件设计 .....	10
3.1 软件设计流程 .....	10
3.2 摄像机标定 .....	12
3.3 模板匹配 .....	15
3.4 图像拼接 .....	18
3.4.1 基于 Sift 图像拼接 .....	18
3.4.2 灰度直方图匹配 .....	20
3.4.3 相位相关分析 .....	22
3.5 边缘提取 .....	24
3.5.1 形态学预处理 .....	24
3.5.2 边缘跟踪 .....	26
3.6 通讯协议 .....	29
3.7 本章小结 .....	31
第 4 章. 人机交互 .....	32
4.1 MFC 简介 .....	32
4.2 MFC 界面搭建 .....	32
4.3 本章小结 .....	33
第 5 章. 结论与展望 .....	35
参考文献 .....	36
致谢 .....	37
附录 .....	38

## 常州大学本科生毕业设计（论文）

第 1 页 / 共 46 页

### 第一章绪论

1.1 课题研究背景随着科技的发展，工业中的检测、测量和定位等相关技术在高质量产品制造和高效生产环境中发挥了巨大作用，并日益增多。随着纺织业的逐步发展，许多人的消费水平也逐渐的提高了，消费者对服装、窗帘等布料的需求越来越大。对于布料的需求现在已经不仅仅局限于经典的风格款式，越来越多的另类风格布料逐渐出现。例如，形状随意、

富有时代气息等风格的面料深受消费者的喜爱，且具有很大的市场。

布料的加工生产一般需要经过裁剪、修边、整理等工序。裁剪的主要任务就是根据图案或路径将布料裁剪成不同形状或特定图案。裁剪是布料产品的生产加工过程中一个非常重要的环节，其质量的好坏不仅会影响产品的尺寸规格，使产品无法达到预期的要求，还会直接影响到产品的成本以及收益。一般情况下，裁剪的质量问题都会成批出现，这会直接导致人力物力的损失以及供货不足导致的客户流失。因此裁剪加工环节是纺织业生产的关键环节，必须高度重视，需高质量、高效率的完成布料的加工裁剪。

国家出台的鼓励裁剪设备行业发展的相关政策，推进了纺织业的发展。但随着改革开放的进一步深化，中国经济的产业结构也正在发生变化，人均获利摊薄，劳动力的成本持续上涨，这也使得纺织业等劳动较密集企业的劳动力成本压力增大，所以使用先进的自动化设备代替劳动力的需求越来越迫切[1]。

裁布机的出现减少了一定的劳动力，但问题也层出不穷，跟人工裁剪相比，虽然速度能够有所提高，但是裁剪精度还不如熟练的裁剪工，而且裁剪出来的有较多毛边，需要增加修边工序。为此，国内外逐渐又出现了非接触式的裁剪机，其中激光裁布机较为突出，主要因为其不与布料直接接触，能做到裁剪平滑少有毛边。但是以上方案都仅限于一台机器裁剪一种形状的布料，而且需要更换裁剪刀头等工具。

所以先阶段的裁布机种类存在许多局限性，如：

- （1）人工裁剪的效率低下、易产生视觉疲劳、劳动强度大；
- （2）现有机器裁剪精度不够、有毛边；
- （3）无法做最优规划、浪费布料；
- （4）裁剪、修边等工序多，繁琐；
- （5）每台机型基本只能适应一种布料的裁剪，成本高；

机器视觉控制系统的出现正好能够满足并迎合这些要求。因此，在现有激光裁剪技术的基础上加入机器视觉技术，将会使裁剪类设备进入真正的自动化，既能节约人力、

节约布料又能高质量高效率的生产，而且一种机型可以应对不同环境下不同布料的裁剪，

这只需在人机交互的界面中提前设置好裁剪类型即可，此类设备成本更低、维护更便捷简单，是未来的发展趋势。

1.2 课题研究的目的是和意义布料在加工之前要根据生产要求进行裁剪，目前的布料裁剪主要是通过刀具进行裁常州大学本科生毕业设计（论文）

第 2 页 / 共 46 页剪,但由于时代生产技术的快速发展,旧裁剪机的暴露出的种种质量问题,同时随着劳动力价格的提高,进而出现了激光裁剪、超声波裁剪等其他裁剪方式。与传统人工裁剪方式相比,裁布机在提高工作效率的同时也保证了生产质量,对布料进行裁剪,剪出不同形状,不同角度,不同长度的布料。同时, 随着人工智能的不断发展,机器视觉技术也能越来越广泛地用于工业。

目前各种自动裁布机机器层出不穷,但是现在的一些自动化裁布机大都是机械化的只能裁剪一种布料或者是需要手工输入裁剪的角度、长度等信息,其次还需要人工修边等工序,较为繁琐。而且由于布料的放置难免会有些倾斜,所以容易导致裁剪的精度低等问题。

在现代科学技术中,激光的运用以及是一种比较新型的科技手段了,激光裁剪切割具有能量稳定集中、可控性强、光束细、方向性好的特点[[2]]。现将激光技术加以应用于布料裁剪加工行业,采用激光来裁剪服饰或特定面料既能提高布料裁剪的技术水平、

降低生产的成本,又能提高产品质量、,而且能增强产品的市场竞争力,提高经济收益。

激光技术早在 90 年代初就在国外的某些服装加工厂企业得到运用,该技术早以成为那些企业用来减少加工工序、提高生产效率并实现工厂生产自动化的主要技术之一。

由于现代社会的进步,视觉逐渐走入工业领域中,使得许多工业设备实现真正的全自动化。机器视觉已经成为以后工业发展的重要一部分,只有给机器加入“眼睛”,它才可以实现全自动化的运行,推动工业的全自动化发展。

现在机器视觉在工业中的发展主要有物体定位、缺陷检测、尺寸测量等方向。结合布料裁剪这一课题,加入机器视觉可以实现 24 小时不间断全自动化裁剪,大大提高生产效率,并且能够将精度控制在 1mm 以内,实现高质量的裁剪。应用机器视觉还能适用于不同布料、不同裁剪需求,与以前设备相比,大大节省了机器成本,实现一机多用的功能。

利用机器视觉技术配合激光裁剪出的布料、面料,与我们人工裁剪出的相比有许多优点:

(1) 机器激光裁剪可运用计算机等控制器,实现自动化裁剪功能;

(2) 材料各种布料可做到最优设计,无损边,且精度高;

(3) 无需多工序修剪锁边等繁琐事项就可以以高质量、高强度的锁边完成裁剪,

提高功效;

(4) 一台机器可以应对不同布料的裁剪,且维护简单;所以,将激光裁剪技术与现在热门的机器视觉结合起来设计出基于机器视觉的智能激光裁剪控制系统,不仅能提高裁剪速度、保证产品质量,又能节约人工成本、机器成本

1.3 课题国内外研究现状截止到现在而言, 机器视觉依然是一个十分活跃的研究领域。在此之前,机器视觉系统的发展主要经历了模式识别、积木世界、起步发展、蓬勃发展等几个阶段。

在机器视觉发展的初级阶段,真正的机器视觉技术在市场上的销售额微乎其微,国际上的机器视觉厂商还未将眼光看向中国市场。仅仅在一些大学和研究所会设立机器视觉常州大学本科毕业设计(论文)

第 3 页 / 共 46 页觉研究所。在接下来的 10 年中,这些研究所或实验室的工程师们成立了自己的视觉公司,他们开发出了第一代图像处理产品,从而能够让机器视觉代替人工处理一些简单的图像识别和处理的工作。但是因为起步晚,技术落后,这些机器视觉技术在实际运用中并没有取得理想中的效果。还有一个原因就是机器视觉这个概念并没有引起全部工程师的重视,在发展上力度有所欠缺,还有就是当时对产品质量认证没有重视。

在机器视觉发展的中级阶段,机器视觉这一概念逐步被人们所熟知,在很多高校都开设了与机器视觉相关的课程,并且引进了国外知名学者前来授课。不仅在教育上,在工业上也引进了国外先进技术和人员,吸引美国和德国等先进企业来大陆投资建厂,落户上海和深圳。随着这股潮流,国内许多厂家也将目光集中到机器视觉这方面,希望也能在激烈的市场竞争中占据一席之地。在外资的刺激下,国内的 机器视觉发展迎来一波较大的发展浪潮,各行各业都对机器视觉技术进行开发,其中自动化行业抓住了这个机遇,在此带动下,走上了机器视觉图像处理发面的发展方向。经过长期的发展,不仅仅在高端精密行业机器视觉的应用已经十分广泛,在民用行业,比如汽车、食品包装、玩具包装、饮料等传统行业也对机器视觉的应用有了庞大的需求。机器视觉在此时期得到了很大的发展。

在现如今全球“工业 4.0”的发展和自动化需求之下,具有精准、高效等特点的机器视觉正受到各个国家的一致关注与认可。自上世纪六十年代提出概念以来,以美国、日本和欧洲为代表的先进国家在相关研发工作中发挥了带头作用。近年来,随着它们在半导体和电子器件领域的卓越表现,它们在国际市场上的发展逐渐进入成熟阶段。

然而,在我国,机器视觉研发和应用相对比较晚,它始于二十世纪八十年代的技术引进,并在 2010 年前平稳发展中才迎来成长期。在此期间,随着“工业 4.0”、“中国制造

2025”等概念的持续深化,以及我国研发技术的不断突破,现如今我国的机器视觉行业

已经从一种单一的应用产品,逐渐的向更多的领域、更多的行业、更多的层次进行延伸。

随着国际市场的成熟和中国市场的增长,两者之间的发展差距和进程的速度发生了明显的变化。在市场方面,日本、美国等企业仍处于领先地位。他们在技术、人才和产品方面具有不可动摇的绝对优势,并在全球市场占有很大份额。然而,就增长率和市场规模而言,中国正在迎头赶上,并取得了显著成绩。目前,中国已稳步发展,成为了世界上第三大机器视觉市场。

在技术研发方面与专利申请等方面,国外专利申请的数量已逐渐开始落后于中国。

自 2016 年以来,中国专利申请的年平均数量已超过 1000 件,但专利申请数量仍远低于国外。与此同时,技术人才也广泛集中在研发领域,操作和应用领域的人才供给仍然不足。在品牌建设方面,国内品牌企业数量正接近 100 家,接近国外品牌数量。然而,在品牌质量和影响力方面仍存在差距,民族品牌的崛起仍需加快。

目前市场上的基于机器视觉的激光裁布机主要剪裁花型较为简单的布匹且裁剪速度较慢,本文设计的系统能够快速的地裁剪各种镂空的、复杂的蕾丝布,这类机型目前在国内外市场上尚未出现。目前这种蕾丝布的裁剪全部依靠工人手工裁剪,且成本很高,

每裁剪 1 米人工成本达到接近 1 元。如下图 1.1 的蕾丝边布料的裁剪,人工裁剪的难度将会非常大。

常州大学本科毕业设计(论文)

第 4 页 / 共 46 页图 1.1 蕾丝边布料原图

1.4 论文主要内容本文从机器视觉的生产和应用出发,针对现在工业中纺织业对于裁布机的精度、速度等要求,提出了一



种基于机器视觉的智能裁布机控制系统，该系统同时使用激光技术和机器视觉技术进行布料的加工裁剪，能更好的实现高效率、高精度、省物、省力的生产。并详细的介绍了该控制系统的设计方法和实现过程，最后通过实际运行结果论证该智能控制系统的可行性。

本文的结构如下：

(1) 分析机器视觉及激光裁布机控制系统的实际应用背景及相关研究现状，引出本课题主要的研究方向、内容和实际意义。

(2) 简述机器视觉系统的构成，以及该控制系统的硬件组成与选型。

(3) 提出系统处理方案，并对图像处理部分进行详细论述。

(4) 简单讲述人机交互界面及流程。

(5) 该课题相关的实验结论及展望。

常州大学本科生毕业设计（论文）

第 5 页 / 共 46 页

第二章智能裁布机系统总体设计

2.1 系统总体框架

2.1.1 机器视觉系统介绍机器视觉是成像系统通过计算机或其他特殊的图像处理软件对目标图像进行的数字处理。一般来说，计算机被用来代替人脑，照相机被用来代替人眼，自动程序和图像处理技术被用来识别目标的大小、形状和颜色。

典型的机器视觉系统可以分为以下几个部分：图像采集、图像处理以及运动控制部分。基于 PC 机的视觉系统由以下几个部分组成，如图 2.1 所示：

图 2.1 基于 PC 的视觉系统

1.工业相机与镜头：这一部分主要最为视觉系统的成像部分，一般一个视觉系统有一套或者多套相机镜头组成，在多相机工作时候可以通过切换采集卡来分别获取数据。

常用的有双目视觉检测，立体检测匹配等。

2.光源：光源是与相机配套存在的器件，其在图像成像过程中发挥着及其重要的作用。工业中常用的一般是 LED 光源，因为其优点众多。

3.传感器:该器件主要用来检测被测物体的状态以及运动情况，一般为光电开关或者是接近开关等。能够及时判断被测物体状态并告知相机准时采集图像。

4.图像采集卡:这一部分器件一般像显卡一样插在 PC 上，它将相机拍摄到的图像以固定的数据流传输到计算机，并将其转化为数字信号。与此同时，它还可以控制许多相机的参数，比如相机的软触发、外触发信号、相机的曝光时间等。

5.PC 平台：该平台一般指的是电脑，大部分的图像数据的处理在这里完成。对一些特殊的应用，通常需要更高频率的 CPU 来处理，高频 CPU 可以减少算法的处理时间。

于此同时还要减少电磁、灰尘、温度、振动等干扰[3]，因此，在工业现场，一般会选择工业计算机来客服现场的不良环境。

6.视觉处理软件：机器视觉处理软件种类较多，一般实现对采集到的图像的处理、识别、定位、检测等，并输出处理结果以供控制单元进行下一步工作。常用的机器视觉编程软件通常以图形式编程环境、C/C++图像库，以及相关控件等形式出现，可以是仅用于 LCD 的检测或 BGA 的检测或是模版对准的专用功能，也可以是通用目的的（包括定位、测量、条码/字符识别、斑点检测等）[4]。

常州大学本科生毕业设计（论文）

第 6 页 / 共 46 页

7.控制单元：该部分主要作为执行机构来控制下一步的工作流程，在图像处理完成之后，对于得到的位置、状态等结果一般需要通过一定的通讯协议发送到控制机构。一些简单的操作常用 PLC 开关量直接控制，如果遇到一些复杂的控制逻辑、比如机械臂的控制等也可依靠运动控制卡来控制。

2.1.2 系统方案流程介绍PLC 工业相机工控机外触发图像传输发送处理结果激光切割图 2.2 硬件框架本文提出的控制系统主要创新点之一在机器的裁剪速度，即对复杂布料的裁剪速度加快。该控制系统设计速度 1 米/秒，即拼接后的 3M 幅面，3-4 秒要裁剪完，对于这种花纹很小的让人工裁剪至少要 1-2 分钟。本文主要做的是智能裁布机视觉系统软件部分。

图 2.3 裁布机实物图常州大学本科生毕业设计（论文）

第 7 页 / 共 46 页方案的原理（如图 2.2 所示）：

首先由 PLC 向工业相机发送外触发信号，工业相机将采集到的图像传输给工控机，在工控机上进行图像处理，并自动规划裁剪路径，最终将处理好的数据传输给控制器，进而控制切割机构进行裁剪切割。

该课题的核心任务主要是机器视觉部分，需要通过工控机接受来自 PLC 的控制信号，实现图像采集、处理、路径拟合规划等工作并设计 MFC 交互界面等。

2.2 系统硬件选型

2.2.1 相机、镜头、光源介绍在机器视觉系统中，最关键的方面是正确的选择照明方案。机器视觉光源不像简单照明物体那样简单，它会直接影响图像质量，进而会影响到系统的性能。我们正在讨论的光源的功能通常是获得清晰的对比度图像，即在保证物体位置改变不影响成像质量的情况下给与足够的亮度的同时增加对比度。

机器视觉系统的核心部分有图像采集和处理。光源对于图像采集过程的影响是非常大的，一个合适的光源能够使成像质量很高，降低算法的复杂度，也能提高系统的定位、

测量等精度，从而提高了系统的可靠性和全面性[5]。与之相反，如果光源选择的不太合适，图像处理算法和成像系统的设计将会事半功倍。在常见的机器视觉系统中光源常见作用一般有以下几种：

(1) 照亮目标物体并提高目标物体的亮度；

(2) 客服环境光的干扰，确保图像的稳定性；

(3) 形成对图像处理最有利的成像效果；  
(4) 用做测量的工具；  
目前工业中最常用的有 LED 光源和卤素灯。其中 LED 光源最为常见，这体现在其节能环保、寿命长、散热效果好、相应时间短、小型化、颜色可自由选择、形状尺寸可定制、成本低等特点。常用的 LED 光源又有很多种，如：环形光源、条形光源、背光源等。目前还没有一种光源能够适应所有环境，因此需要因地制宜，需要根据实际环境的不同以及最优成像来选取合适的光源。  
在一套机器视觉系统中，镜头是必不可少的器件，它会直接影响到成像的质量和算法的实，镜头的主要功能是将成像目标放置在图像传感器的光敏表面上，镜头的合理选择和安装是机器视觉系统设计中的一个重要环节[6]。  
不同的工业镜头的成像质量差异很大，即使是同一类型的工业镜头。这主要是由于不同的材料、加工精度和镜片结构等原因。与此同时，这也导致了不同等级的工业镜头价格的巨大差异，从几百元到几千元不等。对镜头设计及生产厂家来说，一般用光学传递函数 OTF 来综合评价镜头的成像质量和好坏。  
像差是影响图像质量的重要方面。常见的像差包括球差、场曲率、色差、散光、和畸变。当评估工业镜头时，通常根据几个使用参数来判断，例如分辨率、清晰度和景深。  
一个工业镜头的好坏，体现在其分辨率、清晰度、景深等方面是否都有很好的性能。  
它可以很好地校正各种像差，但同时它的价格会增加几倍甚至几百倍[7]。在掌握了一些常州大学本科生毕业设计（论文）

第 8 页 / 共 46 页规律和经验，就可以用同样水平的工业镜头来达到更好的效果。  
工业相机也被称为摄像机，主要应用于工业方面。与传统的民用相机相比，它们具有高精度，抗干扰强传输能力高的优点。目前，市场上大多数工业相机都是基于 CCD 或 CMOS 芯片的相机。这两种相机在成像方面没有太大的区别[8]。前者开发的比较早，所以种类偏多，但是因为体积大、成本高，在逐渐发展中人们已经将视野放到后者的开发中去。固体摄像器件虽然发展晚，但是具备了前者不具备的优点：体积小、灵敏度高、抗干扰强、失真小、寿命长等优点。所以后者逐步取代了前者。也标志着工业摄像机逐步在向小型化、高效化、廉价化发展。而主流的工业摄像机基本是按照 CCD 和 CMOS 感光件来区分的。  
表 2.1 CCD 与 CMOS 参数比较特点 CCD CMOS  
输出的像素信号电荷包电压芯片的输出信号电压数据位相机输出信号数据位数据位填充因子高中系统噪声低中到高放大器适配性高中相机组件 PCB+多芯片+镜头单芯片+镜头  
2.2.2 硬件选型图 2.4 大恒 Gig E 相机图 2.5 12mm 工业镜头硬件选型及依据：  
裁布机视觉部分硬件框架如图 2.2 所示。主要硬件包括工业相机（含镜头、光源等）、工控机、PLC 等。本课题待裁剪幅面宽为 3m，但摄像机每次成像大约能覆盖 40cm，所以需要进行多幅图像采集后进行拼接处理。

指 标
疑似剽窃文字表述
1. 要求。本文相对于传统的裁布机提出了一种基于机器视觉的智能裁布机控制系统，并在现有条件下进行相关分析和研究。
2. 详细的介绍了该控制系统的设计方法和实现过程，最后通过实际运行结果论证该智能控制系统的可行性。
3. 应用，通常需要更高频率的 CPU 来处理，高频 CPU 可以减少算法的处理时间。
4. 不同的工业镜头的成像质量差异很大，即使是同一类型的工业镜头。这主要是由于不同的材料、加工精度和镜片结构等原因。

2. 基于机器视觉的智能裁布机控制系统_第2部分			总字数：11244
相似文献列表			
去除本人已发表文献复制比：2.8%(311)      文字复制比：2.8%(311)      疑似剽窃观点：(0)			
1	基于机器视觉的环冷机台车车轮状态监测系统 陈翔(导师：邹俊) - 《浙江大学博士论文》 - 2018-11-01	0.8% ( 95 )	是否引证：否
2	图像拼接技术的研究 孙瑶琨 - 《大学生论文联合比对库》 - 2017-04-18	0.6% ( 68 )	是否引证：否
3	基于中值滤波与灰度级形态学的综合去除椒盐噪声方法 岳俊华,李岩,纪玉波,郭立红 - 《长春理工大学学报》 - 2004-09-30	0.6% ( 64 )	是否引证：否
4	优化综合滤波方法抑制医学图像强脉冲干扰噪声的研究 秦然; - 《农业网络信息》 - 2009-04-26	0.6% ( 64 )	是否引证：否
5	骨髓细胞图像分割算法研究 范晓静(导师：田秀华) - 《辽宁工程技术大学硕士论文》 - 2008-06-01	0.5% ( 61 )	是否引证：否
6	基于视觉的室内巡检机器人导航技术研究 李鹏(导师：朱华) - 《中国矿业大学博士论文》 - 2018-05-01	0.5% ( 59 )	是否引证：否
		0.5% ( 58 )	

7	基于超像素时空显著性的运动目标检测算法 云红全;徐力;孙晓;明德烈;鞠雯; - 《红外技术》 - 2015-06-04 1	是否引证：否
8	基于图像处理技术的织物疵点检测算法研究 崔月平(导师：韩润萍) - 《北京服装学院博士论文》 - 2011-06-01	0.5% ( 58 ) 是否引证：否
9	基于数字图像空域处理的图像差异检测技术 柴西林; - 《甘肃科技》 - 2012-02-15	0.5% ( 55 ) 是否引证：否
10	基于医学数字X线影像频谱特征的图像检索 马安然;任宁宁;韩立博;孙勇;绍岩;邱建峰; - 《中国医学物理学杂志》 - 2016-09-25	0.5% ( 52 ) 是否引证：否
11	基于GLCM木材树种识别方法的研究 梁勇(导师：徐魁梧) - 《南京林业大学博士论文》 - 2010-06-01	0.4% ( 42 ) 是否引证：否
12	基于图像匹配的应用研究 武潇(导师：肖夏) - 《天津大学博士论文》 - 2011-12-01	0.4% ( 42 ) 是否引证：否
13	基于H-CFAR的SAR图像快速舰船目标检测 陈利民;孟俊敏;杨学志;郎海涛;张婷; - 《合肥工业大学学报(自然科学版)》 - 2012-12-28	0.3% ( 38 ) 是否引证：否
14	基于机器视觉和PLC的瓶盖瑕疵检测系统 吴宇 - 《大学生论文联合比对库》 - 2018-06-02	0.3% ( 37 ) 是否引证：否
15	病理图像拼接算法优化 周明 - 《大学生论文联合比对库》 - 2018-05-25	0.3% ( 37 ) 是否引证：否

#### 原文内容

考虑到处理算法的速度和精度以及视野大小常州大学本科生毕业设计（论文）

第 9 页 / 共 46 页等因素，这些硬件最终选型为工业相机采用大恒 130 万像素外触发 GigE 相机，镜头 12mm，相机镜头分别如图 2.4 和图 2.5。工控机 i3 或 i5 CPU。选型依据如下：大恒相机

现为工业中主流的相机，其有着工作稳定、帧率高、SDK 资料丰富等优点，由于需要进行相机的外触发，所以选择了能够外触发的大恒 GigE 相机，又因为考虑到对算法处理速度的要求，所以相机像素不能太大，不然传输过程会浪费掉大量时间，所以选择了 130

万像素；根据相机固定高度和视野大小，镜头选择了 12mm 的；光源分好多种，例如条形光源、面光源、环形光源等，由于相机固定不高，所以选择环形光源较为合适；对算

法和传输速率的要求比较高，所以工控机也需要有较高的主频，所以选择性能较优的 i5 CPU 的工控机比较合适。

2.3 本章小结本章主要从系统总体框架和系统硬件选型两个方面进行介绍。在系统总体框架中主要介绍了机器视觉系统以及本系统的方案流程。在硬件部分先对相机、镜头、光源进行了详细的介绍，并分别对硬件的选型以及选型依据给出了详细的阐述。

常州大学本科生毕业设计（论文）

第 10 页 / 共 46 页

第三章智能裁布机系统软件设计

3.1 软件设计流程设计步骤：

（1）首先要做的是标定，将机械坐标系与图形坐标系进行标定关联。

（2）获取图像，然后进行图像处理。

（3）将处理结果发送给控制器，进而控制切割机构。

（4）制作 MFC 人机交互界面，将以上算法进行移植在一起。

（5）进行软硬件联调。

机械标定流程，如图 3.1 所示。

首先 PLC 先读取工控机上提前设置好的机械标定数据，控制激光头在指定位置打标定点；打完所有标定点后，返回相机拍照位，并给工控机通讯，通知工控机可以采集图像；工控机软触发采集图像，提取图像上的标定点，并结合机械标定点，分别建立机械

坐标系和图像坐标系，经过图像的仿射变换完成机械坐标标定。

PLC读取工控机上

的机械标定数据PLC控制机械部分

完成标定动作PLC向工控机发送标定完成信号工控机软触发相机采集图像提取图像上的标定点和机械标定点，完成机械标定图 3.1 机械标定流程常州大学本科生毕业设计（论文）

第 11 页 / 共 46 页连续工作流程，如图 3.2 所示。分为 3 个阶段：

（1）提取定位花型阶段。PLC 控制传送带匀速前进，并向工控机发送定位花型信号；

工控机控制相机每隔 200ms（视现场调试效果可修改）连续软触发采集图像，并寻找图像中是否包含指定花型；若找到指定花型，向 PLC 发送反馈；PLC 收到反馈后停止传送带前进。

（2）相机横移、图像拼接阶段。PLC 向工控机发送相机横移信号，工控机将相机工作方式切换为外触发方式，相机等待 PLC 给的外触发信号；当相机收到外触发信号时，进行图像的采集，再将图像发送至工控机，工控机将当前采集到的图像与上一次采集到的图像拼接；拼接完成后，向 PLC 发送完成信号，PLC 控制相机横移至下一位置。

（3）提取裁剪路径阶段。PLC 先向工控机发送横移完成信号；工控机确认所有图像拼接完成，开始提取裁剪路径并生成路径文件；完成后向 PLC 发送完成信号，PLC 读取



路径文件，控制激光头裁剪。

PLC

工控机工控机工控机发送定位花型信号发送相机横移信号发送横移完成信号工控机联系软触发相机采集图像，并寻找指定花型相机等待PLC发送来的硬触发信号工控机提取裁剪路径并生成路径文件向PLC发送找到花型信号，PLC控制传送带停止PLC硬触发相机获取图像，图像拼接向PLC发送完成路径提取信号图 3.2 连续工作流程常州大学本科生毕业设计（论文）

第 12 页 / 共 46 页图像算法流程，如图 3.3 所示。

当匹配到待裁剪花型之后，PLC 控制电机停止，并发送外触发信号控制相机采集图像，然后步进一定位移继续发送外触发信号采集，直到横向采集完成。然后在 PC 端做图像的拼接，即将以上水平横移的多幅图像拼接成一个宽视角图像，然后做图像预处理并确定待裁剪边缘。再将确定好的边缘经过标定数据转换到机械坐标下，并通过 modbus 通信协议发送到裁剪机构进行裁剪，并在传输裁剪之前需要对数据进行 0.5mm 间隔差分提取。

图像获取图像拼接图像预处理

确定裁剪边缘PLC硬触发裁剪路径规划图 3.3 图像算法流程

3.2 摄像机标定我们通常所说的工业相机的标定大致分为两种，一种是相机内参的标定，此标定的作用一般是用来矫正相机的畸变，获得相机的内参进行测量等。相机内参的标定应用做多、效果也比较好的目前是张正友标定算法。

本文详细介绍的是第二种方法，即相机与机械手之间的标定，也叫手眼标定。也就是建立起图像坐标系和机械手坐标系之间的关系，即给机械手装上“眼睛”[9]。

如下图所示：OXY 为机械手坐标系，O'X'Y'为相机坐标系。theta 为两个坐标系之间的夹角。假设 P 点在图像上的位置如图，则 P 在机械手坐标系有一个坐标，在图像坐标系也有一个坐标。本文要做的工作就是图像上的任意一点都可以转化为机械手坐标系上的坐标点：P(Machine) = f(P(Image))。接下来介绍如何找到这个关系。

常州大学本科生毕业设计（论文）

第 13 页 / 共 46 页

图 3.4 手眼标定坐标系

OXY 是机械坐标系，O'X'Y'为相机图像二维坐标系从上图可以看出坐标转化关系： $x = r \cos(\theta)$

$y = r \sin(\theta)$  (3.1)

$x = r \cos(\theta) + x_0$  (3.2)

其中 r 是毫米像素比、(mm/pixel)就是一个毫米有几个像素，theta 为两个坐标系之间的夹角，(x0,y0)为图像坐标原点到机械坐标原点的距离。

简化公式得到： $x = a + b \cos(\theta) + c$  (3.3)  $y = b \sin(\theta) + d$  (3.4)

其中： $a = x_0$ ,  $b = r$ ,  $c = x_0$ ,  $d = y_0$ 。

由上面可知，图像坐标系与机械坐标系之间的转换关系，其实就是一个包含旋转和平移的 2\*3 矩阵。

常用的手眼标定算法有四点标定、九点标定、多点标定等。由于对精度要求较高，所以本文采用的为多点标定[10]。标定步骤如下：

- (1) 将相机位置、机械手位置固定好，固定好之后不能移动。
- (2) 制作标定板，本文采用的为打点制作，即在裁剪位通过激光头控制步进电机打出多个点，如图 3.5。
- (3) 记录下(2)中点的机械坐标。
- (4) 在不移动标定板的情况下，退回到拍照位打开相机拍照，获取这些点的图像坐标。
- (5) 进行全仿射变换，求解变换矩阵。

常州大学本科生毕业设计（论文）

第 14 页 / 共 46 页图 3.5 打点标定图

标定部分核心代码：

```
vector<Point2f> _image_piont_; vector<Point2f> _real_piont_; for (size_t i = 0; i < 20; i++) {
    _image_piont_.push_back(Point2f(Row[i], Col[i]));
    _real_piont_.push_back(Point2f(qx[i], qy[i])); }
```

```
Mat _affine_matrix_ = estimateRigidTransform(_image_piont_, _real_piont_, true); cout << "\n_affine_matrix_ : \n" << _affine_matrix_ << endl;
```

其中 estimateRigidTransform 为矩阵标定函数，输入为图像坐标和机械坐标，输出为 2\*3 的变换矩阵。

标定结果如图 3.6：

图 3.6 标定结果常州大学本科生毕业设计（论文）

第 15 页 / 共 46 页

3.3 模板匹配模板匹配是在工业中经常会用到的一项图像处理技术，它是在图像中寻找目标的方法之一。模板匹配的工作方式跟直方图的反向投影基本一样，大致是通过在源图像上滑动图像块对模板图像和源图像进行匹配，比较各处与模板是否“相似”，当相似度足够高时，就认为是找到了目标[11]。

用 T 表示模板图像，I 表示待匹配图像，切模板图像的宽为 w 高为 h，用 R 表示匹配结果，匹配过程如下图所示：

图 3.7 模板匹配过程图常见的模板匹配是基于灰度匹配的方法。Opencv 中提供了以下几种算法：

(1) 平方差匹配  $R(x, y) = \sum [ (I(x, y) - T(x, y))^2 ]$  (3.5)

该方法利用两幅图像像素差的平方求和进行匹配，最好为 0，匹配越差，匹配值越大。

(2) 归一化平方差匹配  $R(x, y) = \sum [ (I(x, y) - T(x, y))^2 ] / \sqrt{\sum (I(x, y))^2 \sum (T(x, y))^2}$  (3.6)

该方法和(1)类似，只不对图像和模板进行了归一化处理。

(3) 相关匹配  $R(x, y) = \sum [ (I(x, y) - T(x, y)) ]$  (3.7)

该方法采用模板和图像互相关计算作为相似度的计算方法，匹配结果较大表示匹配程度较好，0 表示最坏的匹配。



(4) 归一化相关匹配  $R(x, y) = \frac{\sum [(\bar{x} - \bar{x})(\bar{y} - \bar{y})]}{\sqrt{\sum (\bar{x} - \bar{x})^2} \sqrt{\sum (\bar{y} - \bar{y})^2}}$  (3.8)

该方法和(2)类似，都是去除亮度线性变化对相似度计算的影响。

常州大学本科生毕业设计 ( 论文 )

第 16 页 / 共 46 页

(5) 相关系数匹配  $R(x, y) = \frac{\sum [(\bar{x} - \bar{x})(\bar{y} - \bar{y})]}{\sqrt{\sum (\bar{x} - \bar{x})^2} \sqrt{\sum (\bar{y} - \bar{y})^2}}$  (3.9)  $\bar{x} = \frac{1}{N} \sum (\bar{x})$  (3.10)  $\bar{y} = \frac{1}{N} \sum (\bar{y})$  (3.11)

该方法简单说就是把图像和模板都减去了各自的平均值，使得这两幅图都没有直流分量。

(6) 归一化相关系数匹配  $\bar{x} = \frac{1}{N} \sum (\bar{x})$  (3.12)  $\bar{y} = \frac{1}{N} \sum (\bar{y})$  (3.13)

$R(x, y) = \frac{\sum [(\bar{x} - \bar{x})(\bar{y} - \bar{y})]}{\sqrt{\sum (\bar{x} - \bar{x})^2} \sqrt{\sum (\bar{y} - \bar{y})^2}}$  (3.14)

该方法较复杂，简答的说就是在减去各自的平均值之外，还要各自除以各自的方差。

但是以上算法在应对光照、噪声等条件时处理较差。由于本课题匹配过程只是为了确定是否为待裁剪花型，全程只判断一次，所以本文可以考虑改进算法。所以本文提出了一种基于图像边缘梯度的 NCC 模板匹配[12]。

该算法，首先基于 Sobel 梯度算子得到以下参数:  $[1 \ 0 \ 1 \ 2 \ 0 \ 2 \ 1 \ 0 \ 1] [1, 1, 1+1, 1 \ 1, , +1, 1, +1, +1+1, +1] (3.15) = [1 \ 2 \ 1 \ 0 \ 0 \ 0 \ 1 \ 2 \ 1] [1, 1, 1+1, 1 \ 1, , +1, 1, +1, +1+1, +1] (3.16)$

图像 XY 梯度  $G = \sqrt{2 \ 2}$  (3.17)

通过 Canny 边缘检测算法得到边缘图像，再基于轮廓得到所有的轮廓点集，基于每个点计算该点的如上式三个值，生成模板信息。再对输入图像进行 Sobel 梯度变换，最后根据梯度模型信息进行匹配。

//计算目标图像梯度核心代码如下：

Mat gray\_Image, mag0, angle0;

常州大学本科生毕业设计 ( 论文 )

第 17 页 / 共 46 页 cvtColor(src0, gray\_Image, COLOR\_BGR2GRAY);

Mat gradx, gready;

Sobel(gray\_Image, gradx, CV\_32F, 1, 0);

Sobel(gray\_Image, gready, CV\_32F, 0, 1); cartToPolar(gradx, gready, mag0, angle0);

匹配算法基于 NCC  $(, ) = \frac{1}{N} \sum ((, ) + (, )) \sqrt{2+2\sqrt{(, )^2 + (, )^2}} = 1$  (3.18)

在匹配过程中，给定一个最新阈值 minScore，如果在任意一点求和匹配小于最小阈值，就停止，继续下一个点开始匹配  $= \{(1+1 \ 1), (, )\}$  (3.19)

//NCC 模板匹配核心代码如下： for(int row = 0; row < gray\_Image.rows; row++){ for(int col = 0; col < gray\_Image.cols; col++){

// 目标边缘梯度 double sdx0 = gradx.at<float>(cur0Y, cur0X); double sdy0 = gready.at<float>(cur0Y, cur0X);

// 模板边缘梯度 double tdx0 = contours\_Info[m][n].Derivative\_X; double tdy0 = contours\_Info[m][n].Derivative\_Y;

// 计算匹配 if((sdy != 0 || sdx0 != 0) && (tdx0 != 0 || tdy0 != 0)){ double n\_Magnitude = mag0.at<float>(cur0Y, cur0X);

if(n\_Magnitude != 0) sum0 += (sdx0 \* tdx0 + sdy0 \* tdy0) \* contours\_Info[m][n].Magnitude / n\_Magnitude; }

// 任意节点 score 之和必须大于最小阈值

Partial\_Score = sum / num; if(partial\_Score < min((minScore - 1) + (n\_Greediness \* num), n\_MinScore \* num)) break;

if(partial\_Score > result\_Score) { // 保存匹配起始点 result\_Score = partial\_Score; result\_X = col; result\_Y = row;

常州大学本科生毕业设计 ( 论文 )

第 18 页 / 共 46 页 }}

本文创建的模板以及匹配结果如图 3.8 和 3.9：

图 3.8 模板图像图 3.9 匹配结果

3.4 图像拼接图像拼接是计算机视觉的一个重要研究热点，也是“智慧城市”、“智能交通”建设与发展中的一项关键技术，我们所说的图像拼接，就是把两张具有公共重叠区域的图像进行无缝的拼接在一起，也就是合成所谓的全景图像。这种应用可应用于车站的动态检测、

商城的人流检测、十字路口的交通检测等，给人以全景图像，告别目前的监控墙或视频区域显示的时代，减轻工作人员“眼”的压力。

3.4.1 基于 Sift 图像拼接基本思想：图像的拼接过程并不是简单的将左右两张图片拼合在一起，而是需要将左右图像出去重叠区域拼接到一块。但是这类拼接也存在许多不足之处。因为两幅图像虽然有重叠区域，但由于拍摄时相机的抖动等因素，使得利用重叠位置拼接不够合理，

而且存在上下位移。因此，关于图像的拼接需要进行图像配准也即是仿射变换来处理变形和旋转等，最后通过仿射变换后的图像进行简单的平移操作即可与基准图像的公共区域进行无缝拼接[13]。拼接基本过程如下：

常州大学本科生毕业设计 ( 论文 )

第 19 页 / 共 46 页图 3.10 Sift 图像拼接流程

(1) 图像预处理是为了增强图像的特征，常用预处理有：图像灰度化、滤波去噪、

矫正畸变校正等。

(2) 特征点提取常用的方法有：sift、surf、fast、Harris 等，在方法上 sift 具有旋转与缩放不变性。

(3) 求解单应性矩阵时首先要弄清楚他们之间的映射关系，是前图到后一图的映射，还是后图到前图的映射，这在变换的时候是非常重要的。

(4) 判断左右 ( 上下 ) 图像是为了明确拼接关系。

图 3.11/12 拼接前左图右图读入图像图像预处理特征点提取特征向量提取特征向量匹配求取单应性矩阵透视变换与拼接输出拼接图像常州大学本科生毕业设计 ( 论文 )

第 20 页 / 共 46 页图 3.13 特征点配准图有以上图像可知，由于特征点、特征向量过多，所以获取特征点、特征向量时间

较长，并且特征向量匹配的时间也会比较长，再通过图像配准和拼接处处理使其光滑连接。

这将会耗费大量的时间，本课题一副宽视野图像基本要拼接五到六次，这样的低效率匹配是不可取的。

图像拼接能否达到期望的效果主要取决于能否选择出一个具有鲁棒性而且高效、快速的图像对准方法，但各种方法都存在他们的这样那样的缺陷[14]。图像拼接算法主要包含两个步骤，一个是图像的配准，另一个是图像的融合，但由于这种算法耗时太大，这在工业中是不被允许的[15]。又因为本课题的图像采集都是水平位移，并不存在旋转的因素。所以，本文采用了一种基于直方图匹配和相位相关分析的一套图像拼接算法。

#### 3.4.2 灰度直方图匹配图像的灰度直方图：

一幅图像通常为 3 通道彩色图像，当我们将其转换为单通道的灰度图时，它的一个非常重要特征是该图像灰度值的分布情况。这种分布情况我们可以通过图像的灰度直方图来描述，图像的灰度直方图能够很方便的观测出图像中每个灰度值所占的多少[16]。

一个图像的灰度直方图是其灰度的函数，描述的是图像中某个像素值所占的个数：

其中，横坐标是灰度值，纵坐标是像素灰度值出现的频率。

直方图匹配：

对于输入的两幅图像经过直方图均衡化和灰度直方图计算计算后，可以对两幅图像进行直方图特征匹配，并通过匹配的结果得到我们需要的内容。直方图匹配常用的方法常州大学本科生毕业设计（论文）

第 21 页 / 共 46 页有：相关匹配、十字形交叉、卡方匹配和巴氏距离法[17]。本文选用的为相关性匹配法。

因为相对于其他方法，相关性匹配法应用更为简单便捷，效率较高。

相关性比较的公式如下：
$$d(1, 2) = \sum (1(i) 1)(2(i) 2) \sqrt{\sum (1(i) 1)^2 \sum (2(i) 2)^2} \quad (3.20)$$

其中： $\sum = 1 \sum ()$

如果  $L1 = L2$ ，即两个图的直方图一样，分子等于分母，值为 1，所以在不严格的情况下，当值为 1 时，我们便认为两个图是一样的。但也有可能出现两个图不一样的情况。

因为灰度直方图计算的是像素点个数的分布情况，不会显示像素点位置，所以很有可能会出现两幅图片不一样，但是相同像素的个数完全一样，那他们的直方图也是一样的，

不过这种情况，不常有。

下面是对该公式更加详细的取值范围分析：并且如果两个变量的相关性越强，相关系数就会越接近 $\pm 1$ ，相关性越弱，相关系数越接近 0。相关系数的值如果是正，则称为正相关；相关系数的值如果是负，则称为负相关；相关系数的值为 0，称为不相关。

表 3.1 相关性匹配值的相关性相关系数的绝对值细分大体划分

1.0 - 0.9 相关性非常强相关相关相关

0.9 - 0.7 相关性有点强

0.7 - 0.5 相关性有点弱未满足 0.5 相关性非常弱不相关灰度直方图的绘制与匹配：

由图 3.11 和 3.12 拼接左图和拼接右图可知，左图和右图是有一部分为重叠区域。所以，可以采用两幅图分别对一组相同列进行绘制灰度直方图。首先，我取了拼接右图的前 50 列，进行灰度直方图的绘制，如图 3.16 所示，然后对拼接左图取最右侧 50 列并每次前移一列绘制灰度直方图，在绘制左图直方图的同时与右图的前 50 列进行灰度直方图相关性匹配，取得一次相关性最高的，作为左图拼接位置。如图 3.15 为拼接左图的最优匹配的 50 列的灰度直方图。

以下为获取最优匹配列的核心程序：

```
Mat Right_5Col = srcImage_right(Rect(0, 0, N, srcImage_right.rows)); //右边图前 N 列 cvtColor(Right_5Col,
hsvImage_right, CV_RGB2GRAY); calcHist(&hsvImage_right, 1, channels, Mat(), hist_right, dims, histSize, ranges); int Index
= -1; double maxHist = 0; for (size_t i = 0; i < 100; i++) {
```

常州大学本科生毕业设计（论文）

第 22 页 / 共 46 页

//依次取左边图后 N 列

```
Mat Left_5Col = srcImage_left(Rect(srcImage_left.cols - i - N, 0, N, srcImage_left.rows)); cvtColor(Left_5Col,
hsvImage_left, CV_RGB2GRAY); calcHist(&hsvImage_left, 1, channels, Mat(), hist_left, dims, histSize, ranges);
```

```
//直方图相关性匹配 double base_hist = compareHist(hist_right, hist_left, 0); if (base_hist > maxHist) { maxHist =
base_hist;
```

```
Index = i; } }
```

其中，maxHist 为相关性匹配的相关系数，Index 为拼接左图与右图的最优拼接列为左图的右边起第 Index+N 列，N 为 50。右图前 N 列灰度直方图及左图最优匹配处 50 列的灰度直方图如图 3.14 和图 3.15。

图 3.14 左图灰度直方图图 3.15 右图灰度直方图

3.4.3 相位相关分析由于相机在水平移动拍照过程中，很难保证机械抖动导致的相位差，以及灰度直方图匹配获得的最优匹配位置也很难做到精准。所以本文在基于灰度直方图匹配的基础上加入了在最优匹配位置进行图像的相位相关分析。

获得直方图最优匹配处于拼接右图的相关位移量核心程序如下：

```
Mat src1, src2, dst1, dst2; src1 = srcImage_left(Rect(srcImage_left.cols - Index - N, 0, N, srcImage_left.rows));
```

常州大学本科生毕业设计（论文）

第 23 页 / 共 46 页 src2 = srcImage\_right(Rect(0, 0, N, srcImage\_right.rows)); src1.convertTo(dst1, CV\_32FC1); //转换为 32 位浮点型 src2.convertTo(dst2, CV\_32FC1);

```
Point2d phase_shift; phase_shift = phaseCorrelate(dst1, dst2);
```

其中 Index+N 为拼接左图最优匹配处，N 为 50。最终获得的水平位移量和垂直位移量分别为 phase\_shift.x 和 phase\_shift.y，且为亚像素精度。

图 3.16 未做相关位移拼接图 3.17 相关位移拼接可以明显的看出，未做相关位移拼接的图片中间拼接处有明显的断层

，拼接不平滑。

经过相关位移拼接的中间拼接处过度平滑，且拼接完整。

经过灰度直方图匹配求得最优匹配位置以及相关分析法求得的最优匹配处的位移量之后，便可进行多幅图像连续拼接。  
拼接步骤为：

(1) 创建一个左图宽度+右图宽度、高度与左右图相同的新图像。

常州大学本科生毕业设计(论文)

第24页 / 共46页

(2) 将左图的左侧到距离右边 Index+N+phase\_shift.x 处，与右图的 phase\_shift.y 行以下部分拼接到创建好的新图像上。其中 Index 为直方图匹配到的最优位置，N 为

50，phase\_shift.x 为水平位移量。

选取左右图 RIO 并进行拼接部分核心程序如下：

```
Mat Left_Gray, Right_Gray; cvtColor(srcImage_left, Left_Gray, CV_RGB2GRAY); cvtColor(srcImage_right, Right_Gray, CV_RGB2GRAY);
```

```
Mat Show_Image = Mat::zeros(Left_Gray.rows, Right_Gray.cols + Left_Gray.cols -  
(int)(N + Index + phase_shift.x + 0.5), CV_8UC1);
```

```
Mat image_roi_left = Left_Gray(Rect(0, 0, Left_Gray.cols - (int)(N + Index + phase_shift.x + 0.5), Left_Gray.rows));  
image_roi_left.copyTo(Show_Image(Rect(0, 0, Left_Gray.cols - (int)(N + Index + phase_shift.x + 0.5), Left_Gray.rows)));
```

```
Mat image_roi_right = Right_Gray(Rect(0, (int)(phase_shift.y + 0.5), Right_Gray.cols,
```

```
Right_Gray.rows - (int)(phase_shift.y + 0.5))); image_roi_right.copyTo(Show_Image(Rect(Left_Gray.cols - (int)(N +  
Index + phase_shift.x + 0.5), 0, image_roi_right.cols, image_roi_right.rows)));
```

图 3.18 多幅图像拼接效果图本文提出的此算法运行时间为 1.5ms 左右，是 shift 图像拼接算法的几十倍，有着很大的速度优势，这也是本文提出的一个较大的算法创新点。

指 标
疑似剽窃文字表述
1. 一个图像的灰度直方图是其灰度的函数，描述的是图像中某个像素值所占的个数： 其中，横坐标是灰度值，纵坐标是像素灰度值出现的频率。 直方图

3. 基于机器视觉的智能裁布机控制系统_第3部分			总字数：11630
相似文献列表			
去除本人已发表文献复制比：7.5%(878) 文字复制比：7.5%(878) 疑似剽窃观点：(0)			
1	Opencv图像识别从零到精通(32)----直方图对比，模版匹配，方向投影 - 小木匠的博客 - CSDN博客 - 《网络( <a href="http://blog.csdn.net">http://blog.csdn.net</a> )》 - 2017	1.0% ( 116 )	是否引证：否
2	学生考试管理系统 孙月明 - 《大学生论文联合比对库》 - 2015-05-30	1.0% ( 111 )	是否引证：否
3	通信工程-张峻康-201403203128-基于胶囊神经网络的车牌识别系统设计 - 《大学生论文联合比对库》 - 2018-06-01	0.8% ( 95 )	是否引证：否
4	041220526 王阳 毕设论文 王阳 - 《大学生论文联合比对库》 - 2016-05-30	0.7% ( 84 )	是否引证：否
5	3258942_王阳_基于视频图像的眼睑位置检测方法研究 王阳 - 《大学生论文联合比对库》 - 2016-05-16	0.7% ( 84 )	是否引证：否
6	集装箱堆场堆存策略的比较研究 汪兴东(导师：靳志宏) - 《大连海事大学博士论文》 - 2011-06-01	0.7% ( 77 )	是否引证：否
7	无线传感器网络前台监控软件的设计与实现 张帆(导师：南国芳;沈强) - 《天津大学博士论文》 - 2012-05-01	0.6% ( 70 )	是否引证：否
8	核燃料棒智能识别与定位系统研究 钱煜(导师：龚志辉;陈铁坚) - 《湖南大学博士论文》 - 2018-04-20	0.5% ( 63 )	是否引证：否
9	基于机器视觉的智能布料图案切割系统设计 陶鹤鸣(导师：赵不赓) - 《江苏大学博士论文》 - 2018-04-01	0.5% ( 60 )	是否引证：否
10	基于纹理图像的琵琶面板选取方法研究 高文韬(导师：黄英来) - 《东北林业大学博士论文》 - 2018-04-10	0.5% ( 58 )	是否引证：否
11	1521041715338_刘鹏伟_基于机器视觉的番茄生长点特征识别 刘鹏伟 - 《大学生论文联合比对库》 - 2018-03-15	0.5% ( 58 )	是否引证：否
		0.4% ( 52 )	



12	基于机器视觉的导光板缺陷检测系统研发 李明睿(导师：李俊峰;宓施峰) - 《浙江理工大学博士论文》 - 2018-12-05	是否引证：否
13	4293267_李涛_Android平台下图像字符的识别研究与实现 李沙沙 - 《大学生论文联合比对库》 - 2017-06-02	0.4% ( 51 ) 是否引证：否
14	通信院-通信-李涛-13120789 李涛 - 《大学生论文联合比对库》 - 2017-06-06	0.4% ( 51 ) 是否引证：否
15	土豆无损检测分级系统设计 陈健韵 - 《大学生论文联合比对库》 - 2016-05-18	0.4% ( 42 ) 是否引证：否
16	1206100016 陈健韵 - 《大学生论文联合比对库》 - 2016-05-23	0.4% ( 42 ) 是否引证：否
17	环糊精共聚物制备与性能研究 陈玉瑛 - 《大学生论文联合比对库》 - 2018-06-05	0.3% ( 40 ) 是否引证：否
18	基于男士西服用机织物性能的自动裁剪工艺参数研究 王雅芝(导师：傅菊芳) - 《苏州大学博士论文》 - 2017-05-01	0.3% ( 38 ) 是否引证：否
19	自然场景图片中文本定位方法研究 潘立(导师：张再跃) - 《江苏科技大学博士论文》 - 2017-12-30	0.2% ( 29 ) 是否引证：否
20	袋装料包智能装车系统研发 黄兆力(导师：杨世锡) - 《浙江大学博士论文》 - 2019-03-01	0.2% ( 29 ) 是否引证：否
原文内容		

3.5 边缘提取边缘提取是图像处理中的一种重要的图像处理技术，在很多领域有着广泛的应用，如工业图像处理、身份鉴定、文本图像分析处理和识别、生物医学图像处理、机器人视觉等。不同类型的图像有着不同的处理方法，对于一些边缘提取方法也只适用于某些特殊类型的图像。

由于本课题的要求是提取出拼接后的宽视野图像中的连续花纹的边缘。所以对于常用的一些图像分割算法便不再适用，主要原因是，原图杂乱的线较多，而且底部有白色链条，导致预处理和分割难度。所以本文提出了一种基于边缘跟踪的边缘提取算法。

常州大学本科生毕业设计（论文）  
第 25 页 / 共 46 页

3.5.1 形态学预处理形态学处理在数字图像处理中一般是指以数字形态学从图像中提取出有利于描述特定形状有用处的分量，比如骨架、凸壳、边界以及过滤、修剪、细化等。

常用的形态学处理运算有：腐蚀、膨胀、开运算、闭运算。并且基于这些基本运算可以推导或者组合出其他各种各样的形态学实用算法[18]。

（1）腐蚀运算

- 1、定义一个卷积核 B，卷积核大小可以是任何形状，比如椭圆、方形等。
- 2、将卷积核 B 与图像 A 进行卷积，计算卷积核覆盖区域的像素点最小值。
- 3、计算出最小值将其赋给参考点的指定像素。图像中高亮区域减小。

4、腐蚀基本原理图如 3.19 所示。

图 3.19 腐蚀原理图结果：使二值化图像减少一圈。

（2）膨胀运算

- 1、定义一个卷积核 B，卷积核大小可以是任何形状，比如椭圆、方形。
- 2、将卷积核 B 与图像 A 进行卷积，计算卷积核覆盖区域的像素点最小值。
- 3、计算出最小值将其赋给参考点的指定像素。图像中高亮区域减小。

4、膨胀基本原理图如 3.20 所示。

图 3.20 膨胀原理图常州大学本科生毕业设计（论文）  
第 26 页 / 共 46 页结果：使二值化图像扩大一圈。

（3）开运算开运算是先对图像进行腐蚀，再对腐蚀后的图像进行同比膨胀。这样可以腐蚀滤掉小的噪点，并且可以对于较大物体在不改变其面积的情况下对其的边缘起到平滑的作用。

（4）闭运算闭运算是先对图像进行膨胀操作，再腐蚀。这样可以对一些空心图形进行一定的填充，或者使得相邻的进行平滑的连接起来。

图 3.21 腐蚀膨胀效果图如图 3.21 即为布料图像经过形态学预处理后的图像。需要提取出的为中间两条边缘。

由于布料原图图像有较多杂乱的线还有链条底板。所以预处理之后中间还有许多噪点，如果要用形态学处理把噪点全去掉就会导致边缘失真严重。为此，本文提出了一种不用考虑中间噪点的边缘跟踪的边缘路径提取算法。

3.5.2 边缘跟踪边缘算法框架如图 3.22：

首先需要找到第一列白到黑跳变点和黑到白跳变点两个边缘点的行坐标，分别记为 up 行和 down 行。由图 3.22 腐蚀膨胀后的图可知，第一列中有许多镂空的黑色区域，所以会有许多黑白跳变点，为此，本文需要每次求出黑白跳变点之后要做如下判断：down

要大于 up，down-up 要大于一定的值。也就是下边缘所在行一定要大于上边缘所在行，下跳变点与上跳变点距离要大于一定值。所以遍历第一列元素加上以上判断条件即可获

得第一个上边缘点和第一个下边缘点。伪代码如下：

```
For(从第一行到最后一行) { up = 白到黑跳变行; down = 黑到白跳变行; if(down>up 且 down-up>=一定宽度) up、down  
即为所求; else  
继续往下找; }  
找到第一列的上下边缘 up 行和 down 行之后, 就将此行为基准, 以 10 个像素为上常州大学本科生毕业设计 ( 论文 )  
第 27 页 / 共 46 页下宽度进行第 2 列的边缘点寻找。以上边缘为例, 即第 2 列在 up 行上下 10 个像素的区间进行搜寻白  
到黑跳变点, 如果没有就判断一下现在边缘的趋势, 如果是向上的趋势,  
就将区间向上平移一个区间继续搜索, 找到白到黑边缘点从新记为 up 点, 以此类推。  
下边缘点同理。  
读入图像图像预处理寻找第一列两个边缘点行坐标分别为 up、down  
第二列分别在 up、down 两行的基础上向上向下找 5 个元素, 找到边缘点记为 up、down  
第 i 列同上若能找到将 up、down 分别上下移动一个区间(10 个元素)继续  
查找  
YES  
NO
```

图 3.22 边缘跟踪流程图根据以上流程可知, 此边缘提取算法的根本是遍历图像元素, 但在遍历的基础上加以改进, 进行边缘跟踪, 这样使得没列的纵向扫描时间缩短十几倍, 一副宽视野图像整体的算法时间提升了很多, 并且不会被上下边缘中间的早点所干扰。边缘提取得到的结果图像如下图:

图 3.23 边缘提取结果常州大学本科生毕业设计 ( 论文 )

第 28 页 / 共 46 页将提取好的路径通过标定数据的转换分别可以得到上下边缘的机械坐标, 并将其保存为 txt 文件, 坐标数据如下:

图 3.24 上边缘机械坐标图 3.25 下边缘机械坐标由于点太过密集, 如果每个点都进行激光裁剪就会导致布料裁剪边缘糊掉, 所以需要对数据进行差分提取, 本课题采取 0.5mm 差分间隔提取, 即每两个激光点坐标的横坐标需要间隔 0.5mm。进行差分提取的另一个原因是因为一副宽视野图像处理出来的数据量较大, 一个坐标有两个数据, 所以一共有将近一万个数据, 这样会导致时间大量的浪费在通讯的传输上, 所以进行 0.5mm 差分提取后发送, 只需要默认发送 y 坐标即可, 这常州大学本科生毕业设计 ( 论文 )

第 29 页 / 共 46 页样可以使通讯时传输的数据量降低至少一倍。分别对上下边缘进行差分提取后还需要将上下边缘路径规划合并连接一起, 就近原则选择 S 形合并。经过差分提取和上下边缘合并后的机械坐标如图 3.25 所示。

图 3.26 差分提取后机械坐标

3.6 通讯协议经过以上图像处理算法得出待裁剪边缘路径后, 需要将其发送给裁剪机构来控制激光常州大学本科生毕业设计 ( 论文 )

第 30 页 / 共 46 页光头进行裁剪。由于是 PC 与 PLC 之间通讯, 所以本文采用了工业中常用的 Modbus 通讯协议。

Modbus 通讯协议是 MODICON 公司最先倡导和使用的一种通讯规则, 现在已被逐渐认可并得到实际应用, 成为了一种标准的通讯协议。目前, 在基于 RS232/485 通讯的过程中, 更是被广泛的采用。常用的 Modbus 通讯协议有两种, 一种是 Modbus-ASCII,

一种是 Modbus-RTU。本文采用的则为 Modbus-RTU 协议[19]。

以下为 Modbus-RTU 数据帧格式以及报文格式:

表 3.2 Modbus-RTU 报文模型设备地址功能代码数据格式 CRC 校验 L CRC 校验 H

8bit 8bit 8bit 8bit 8bit

表 3.3 主机对从机写数据操作

0x01 06 00 01 00 17 98 04

从机地址功能号数据地址数据 CRC 校验表 3.4 主机对从机读数据操作

0x01 03 00 01 00 01 D5 CA

从机地址功能号数据地址读取数据个数 CRC 校验表 3.5 从机对主机返回内容

0x01 03 02 00 17 F8 4A

从机地址功能号数据字节个数两个字节数据 CRC 校验表 3.6 Modbus-RTU 常用功能码数据类型读功能码写功能码对象类型离散量输入 02 单个位线圈状态 01 05,15 单个位输入寄存器 04 16 位字输出寄存器 03 06,16 16 位字表 3.7 寄存器起始地址数据类型参数地址, 寄存器编号离散量输入 00001~0FFFF

线圈状态 10001~1FFFF

输入寄存器 30001~3FFFF

保持寄存器 40001~4FFFF

常州大学本科生毕业设计 ( 论文 )

第 31 页 / 共 46 页本文采用封装好的 modbus-RTU.dll 动态链接库进行与 PLC 之间的通讯。通讯报文格式为: 先往 R2100 地址, 即 42101 写入边缘点数量; 再从 R2102 地址, 即 42103 开始写入坐标数据。

核心程序如下: int iValue = mfc06(1, 1, 2100, &number); for (int i = 0; i < number; i++) { int xValue = mfc06(1, 1, 2102 + 2 \* i, &EdgePoint.x); int yValue = mfc06(1, 1, 2103 + 2 \* i, &EdgePoint.y); }

其中 number 为边缘点数量, EdgePoint 为边缘路径坐标点数组。

函数原型如下: function mbrtufcn06(nport, node, address, value:LongInt)

参数说明: nport: 串口号, 取值为 1~32, 代表 COM1~COM32; node: 从站号, 取值为 1~255; address: 元件地址, modbus 标准地址 40001~4XXXX 地址区; value: 给元件赋的值, 取值范围为-32768~+32767;

返回值：操作成功返回:1，操作失败返回:-1。

3.7 本章小结本章主要针对该智能控制系统的整体软件算法进行了详细的阐述。首先该系统要做的是标定部分，即图像坐标与机械坐标的转换关系；然后对待裁剪布料花型的匹配；然后进行图像的相关分析进而做拼接操作；再进行路径的拟合提取并通过标定矩阵转换为实际坐标；最后将经过 0.5mm 差分提取后的机械坐标通过 modbus 发送给 PLC。

常州大学本科生毕业设计（论文）

第 32 页 / 共 46 页

#### 第四章人机交互

人机交互现在已经成了各个项目必不可少的一个环节。一个好的人机交互的上位机界面，能够简化操作过程，实时显示我们需要的信息等。

现在常见的上位机界面的开发软件及语言有：MFC、QT、C#、MatlabGUI 等。由于本课题是基于 Opencv 的图像处理相关算法，所以在上位机界面制作的选择上有一定的限制。最终选择了在 VS2013 下基于 VC++ MFC 的上位机界面的开发。

4.1 MFC 简介MFC ( Microsoft Foundation Class ) 微软基础类库，用基于 C++编写的一个类集，常用于 Windows 程序的编写。该类集以层次结构组织起来，其中封装了大部分的 Windows

接口函数和 Windows 控件[20]。

MFC 的应用程序框架优点：

MFC 根据 C++类组织在一起，该类封装了 Windows 的接口函数并提供了 Windows

应用程序常见的任务。它的几个基类提供一般功能，由基类派生类来进行具体的实现。

4.2 MFC 界面搭建一个好的开端是设计一个用户界面。首先，必须决定哪些类型的用户可以使用该程序，并需要根据设置相应的用户界面对象。Windows 用户界面有一些标准控件，如按钮、

菜单、滚动条和列表，这些都是 Windows 用户已经熟悉的[21]。搭建界面如图 4.1：

图 4.1 界面搭建图常州大学本科生毕业设计（论文）

第 33 页 / 共 46 页

图 4.2 界面运行效果图MFC 界面操作流程：

- (1) 首先进行相机标定，得到标定矩阵，并保存；
- (2) 相机进行软触发获取图像；
- (3) 进行图像匹配，确认待裁剪布料花型；
- (4) 继续进行软触发获取图像，获取 4 副图像为一次拼接周期；
- (5) 进行图像拼接；
- (6) 对拼接后的图像进行路径拟合；
- (7) 保存路径坐标并通过 Modbus 发送给 PLC；

其中进行图像匹配、拼接、拟合路径效果图如 4.2。

MFC 界面换肤：

由上面效果图可知，MFC 制作出的上位机界面并不美观。所以考虑将界面进行美化。

本文采用了运用较多的 Skin++皮肤库来进行界面美化。界面美化效果图如图 4.3 以及美化运行效果图如图 4.4。

4.3 本章小结本章主要对人机交互进行了简单的介绍。通过对 MFC 的介绍，搭建出合理美观的 MFC

交互界面。并通过常用的 Skin++换肤库进行了界面的换肤美化。

常州大学本科生毕业设计（论文）

第 34 页 / 共 46 页图 4.3 界面美化效果图图 4.4 美化运行效果图常州大学本科生毕业设计（论文）

第 35 页 / 共 46 页

#### 第五章结论与展望

本文主要研究并设计开发一个基于机器视觉的智能裁布机控制系统，该系统能够有效的对纺织业中布料的加工裁剪工序进行高效、高质量、低成本的生产，以代替现有的人工裁剪。

本文研究的课题与常州市翔云智能控制系统股份有限公司合作开发，并获得校卓越计划课题资助，本文成果也正在申报国家发明专利和软件著作权。

本文的主要创新点之一是该系统的速度，对于复杂布料的裁剪速度在 1mi/秒，远远快与人工裁剪。并且在图像的拼接算法上有一定的创新。

本文主要对以下部分进行了研究和讨论：

1. 本文先论述了裁布机的应用背景和国内外研究现状、发展趋势等。又介绍了机器视觉与传统裁布机在裁剪上的对比与优势，说明了基于机器视觉裁布机的可行性、必要性和相关价值。
  2. 本文根据总体方案的设计，进行的硬件的分析与选型，并简单的阐述了相关硬件的工作原理及特征参数。
  3. 提出了一种改进的图像拼接算法，此方法效率高，算法运行速度快。但此方法只适用于只有平移关系的图像之间的拼接。
  4. 对系统整体以及相关图像处理算法进行了详细的阐述，其中包括相机的手眼标定、确定花型时的图像匹配、直方图与相关法的图像拼接、边缘跟踪算法路径提取拟合以及 Modbus 通讯等。
  5. 在系统开发方面，使用 C++语言，基于 MFC 开发上位机人机交互界面。图像处理算法基于 OpenCV3.0 进行开发。
- 由于准备时间和能力的不足以及收集资料的不全面，本文的研究方法及设计方案尚存在一些问题。并在即将结束论文研究之际对今后的一些展望：
1. 软硬联调较为麻烦，很大的精度影响也取决于硬件以及相关机械结构。
  2. 随着设备运行时间的正常，环境光的变化或者光源亮度的变化都会引起图像的质量发生变化，这使得算法可能会失效。为此需要增加算法的鲁棒性，使其能够实现参数的定期自整定。



参考文献 [1]

王政, 寿弘毅, 盛卫民. 自动铺布机在服装裁剪自动化系统中的应用[J]. 浙江纺织服装职业技术学院学报, 2015, 14(1). [2]

周洪宇. 激光服装自动裁剪系统的设计[D]. 长春理工大学, 2002. [3]

黄传霞. IC 探针图像模式识别与测距技术研究[D]. 苏州大学. [4]

刘少勇. 机器视觉在 RFID 标签封装设备中的应用研究[D]. 华中科技大学, 2007. [5]

李玉广, 张小虎, 庄萍萍. 摄像测量中光源应用研究[J]. 照明工程学报, 2010, 21(1):19- 22. [6]

何修远. 基于机器视觉的 PET 瓶盖缺陷在线检测技术研究[D]. 2015. [7]

刘圣. 工业图像检测系统关键技术研究及应用[D]. 上海交通大学. [8]

雷敏华, 陈良. 基于机器视觉的端子尺寸检测系统[J]. 机电工程技术, 2013(7). [9]

付贵. 基于机器视觉的工业机器人标定方法研究[J]. 南方农机, 2018, v.49 ;

No.301(09):153-154. [10]

吴安成, 何卫锋, 欧阳祥波. 基于 OpenCV 的码垛机器人手眼标定方法[J]. 制造技术与机床, 2018, No.672(6):54-58. [11]

王丛政, 李启光. 基于 Opencv 的图像多目标模板匹配算法[J]. 电子技术与软件工程, 2018. [12]

Brunelli R. Template Matching Techniques in Computer Vision: Theory and Practice[M]. 2009. [13]

Lecoq H, Fabre F, Joannon B, et al. Search for factors involved in the rapid shift in

Watermelon mosaic virus (WMV) populations in South-eastern France[J]. Virus Research, 2011, 159(2):115-123. [14]

王军. 基于三基色原理的彩色图像拼接算法改进[D]. 内蒙古工业大学. [15]

王国锐. 遗传算法及其在图象拼接中的应用[D]. 华中科技大学. [16]

关晨曦, 周途. 直方图零点与图像无损压缩定量关系研究[J]. 电子设计工程, 2018(16):24-28. [17]

Shen D. Image registration by local histogram matching[J]. Pattern Recognition, 2007, 40(4):1161-1172. [18]

Jia X. Fabric Defect Detection Based on Open Source Computer Vision Library

OpenCV[C]// International Conference on Signal Processing Systems. 2010. [19]

马岩. 基于 PLC 的移动机器人控制系统研究与开发[D]. 2014. [20]

Jones R M. Introduction to MFC programming with visual C++[J]. Prentice Hall, 2000. [21]

廖晓芳. VC 编程技巧研究[J]. 湖北科技学院学报, 2009, 29(3):54-55.

常州大学本科生毕业设计 ( 论文 )

第 37 页 / 共 46 页

致谢

大学四年已接近尾声。在此本科论文结束之际, 我首先需要感谢的是我的指导老师张老师, 他从一开始的课题方向的选择, 到最后的论文的完成和排版, 都一直非常有耐心的进行指导。前期张老师给我提供了许多课题相关资料, 并在中期进行算法编写过程中给我许多算法思路, 最后又在论文的写作以及各式排版等问题上——给我详细的讲解和改正。他对机器视觉以及模式识别方向的研究使我受益匪浅。也正是张老师从大三就开始带我入门图像处理, 才使得我的毕设做起来不那么困难。在此, 我谨向张老师致以由衷的感谢! 同时, 我还要感谢在我学习期间给予我极大鼓励和关心的老师同学和朋友, 尤其是在我算法部分没有头绪的时候, 正是他们的帮助才让我感觉有了很大的动力。

常州大学本科生毕业设计 ( 论文 )

第 38 页 / 共 46 页附录

1. 图像拼接算法Mat Hist\_Together(Mat srcImage\_left, Mat srcImage\_right) {

// 【1】声明储存 HSV 图像的矩阵

Mat hsvImage\_left, hsvImage\_right;

// 【2】初始化计算直方图需要的实参 const int nimages = 1; int channels[] = { 0 }; int dims = 1; int histSize[] = { 256 };

float hranges[] = { 0.0, 255.0 }; const float\* ranges[] = { hranges };//一定要有 const

MatND hist\_right, hist\_left;

// 【3】计算右图灰度直方图

Mat Right\_5Col = srcImage\_right(Rect(0, 0, N, srcImage\_right.rows)); //右边图前 N 列 cvtColor(Right\_5Col, hsvImage\_right, CV\_RGB2GRAY); calcHist(&hsvImage\_right, 1, channels, Mat(), hist\_right, dims, histSize, ranges);

// 【4】平移计算左图直方图, 并匹配找到最优匹配 int Index = -1; double maxHist = 0; for (size\_t i = 0; i < 200; i++) {

Mat Left\_5Col = srcImage\_left(Rect(srcImage\_left.cols - i - N, 0, N, srcImage\_left.rows)); cvtColor(Left\_5Col, hsvImage\_left, CV\_RGB2GRAY); calcHist(&hsvImage\_left, 1, channels, Mat(), hist\_left, dims, histSize, ranges); double base\_hist = compareHist(hist\_right, hist\_left, 0);

//printf("方法 [%d] 的匹配结果如下: \n\n【基准图 - 基准图】: %f \n", i, base\_hist); if (base\_hist > maxHist) { maxHist = base\_hist;

Index = i; } }

// 【5】相位相关分析法求匹配好的相位差

Mat src1, src2;

Mat dst1, dst2; src1 = srcImage\_left(Rect(srcImage\_left.cols - Index - N, 0, N, srcImage\_left.rows)); src2 = srcImage\_right(Rect(0, 0, N, srcImage\_right.rows)); cvtColor(src1, src1, CV\_BGR2GRAY); //转换为灰度图像常州大学本科生毕业设计 ( 论文 )

第 39 页 / 共 46 页 src1.convertTo(dst1, CV\_32FC1); //转换为 32 位浮点型 cvtColor(src2, src2, CV\_BGR2GRAY);

src2.convertTo(dst2, CV\_32FC1);

```

Point2d phase_shift; phase_shift = phaseCorrelate(dst1, dst2);
//cout << endl << "warp : " << endl << "\tX shift : " << phase_shift.x << "\tY shift : " << phase_shift.y << endl;
// 【6】进行拼接
Mat Left_Gray, Right_Gray; cvtColor(srcImage_left, Left_Gray, CV_RGB2GRAY); cvtColor(srcImage_right, Right_Gray,
CV_RGB2GRAY);
Mat Show_Image = Mat::zeros(Left_Gray.rows, Right_Gray.cols + Left_Gray.cols -
(int)(N + Index + phase_shift.x + 0.5), CV_8UC1); //创建一个两倍宽的图像
Mat image_roi_left = Left_Gray(Rect(0, 0, Left_Gray.cols - (int)(N + Index + phase_shift.x + 0.5), Left_Gray.rows));
image_roi_left.copyTo(Show_Image(Rect(0, 0, Left_Gray.cols - (int)(N + Index + phase_shift.x + 0.5), Left_Gray.rows))); //将
roicopy 到宽图的最左边
Mat image_roi_right = Right_Gray(Rect(0, (int)(phase_shift.y + 0.5), Right_Gray.cols,
Right_Gray.rows - (int)(phase_shift.y + 0.5))); image_roi_right.copyTo(Show_Image(Rect(Left_Gray.cols - (int)(N +
Index + phase_shift.x + 0.5), 0, image_roi_right.cols, image_roi_right.rows))); return Show_Image; }
2. 边缘跟踪算法 int main() {
_affine_calibrate_ = Calibrate(Col, Row, qx, qy, 20);
Mat ero_dil_image = imread("ero_dil_image.bmp", 0);
Mat dstOut = ero_dil_image(Rect(Roi, 0, ero_dil_image.cols - Roi, ero_dil_image.rows));
//先找到上下边缘的第一个点 int upPoint = 0, downPoint = 0; uchar *image = dstOut.data; int step = dstOut.step; for
(size_t i = 0; i < dstOut.rows; i++) { int pix[2]; pix[0] = image[0]; image += step; //下一行 pix[1] = image[0]; if (pix[0] == 255 &&
pix[1] == 0) //白-黑跳变 {
常州大学本科生毕业设计 ( 论文 )
第 40 页 / 共 46 页 upPoint = i; } if (pix[0] == 0 && pix[1] == 255) //黑-白 { downPoint = i; } if (upPoint != 0 && downPoint
!= 0) { if (downPoint > upPoint) { if (downPoint - upPoint >= 60) { break; } else { upPoint = 0; downPoint = 0; } } else {
downPoint = 0; } } } vector <Point> upVector, downVector; int last_upPoint = upPoint; int last_downPoint = downPoint;
//先找上边缘点, 边缘跟踪算法 for (size_t lie = 0; lie < dstOut.cols; lie++) //x 方向 { uchar *image1 = dstOut.data;
image1 += step * (last_upPoint - N); int get_w2b_flag = 0; for (size_t i = 0; i < 2 * N; i++) { int pix[2]; pix[0] = image1[lie];
image1 += step; //下一行 pix[1] = image1[lie]; if (pix[0] == 255 && pix[1] == 0) //白-黑跳变 { get_w2b_flag = 1;
upVector.push_back(Point(lie + Roi,

```

指 标

疑似剽窃文字表述

1. 卷积核 B 与图像 A 进行卷积, 计算卷积核覆盖区域的像素点最小值。  
3、计算出最小值将其赋给参考点的指定像素。
2. 2、将卷积核 B 与图像 A 进行卷积, 计算卷积核覆盖区域的像素点最小值。  
3、计算出最小值将其赋给参考点的指定像素。
3. 一个好的开端是设计一个用户界面。首先, 必须决定哪些类型的用户可以使用该程序, 并根据需要设置相应的用户界面对象。
4. 论文结束之际, 我首先需要感谢的是我的指导老师张老师, 他从一开始的课题方向的选定, 到最后的论文的完成和排版, 都一直非常有耐心的进行指导。前期张老师给我提供了
5. 在此, 我谨向张老师致以由衷的感谢! 同时, 我还要感谢在我学习期间给予我极大鼓励和关心的老师同学和朋友,

#### 4. 基于机器视觉的智能裁布机控制系统\_第4部分

总字数 : 7230

相似文献列表

去除本人已发表文献复制比 : 0%(0) 文字复制比 : 0%(0) 疑似剽窃观点 : (0)

原文内容

```

last_upPoint - N + i)); last_upPoint = last_upPoint - N + i;
常州大学本科生毕业设计 ( 论文 )
第 41 页 / 共 46 页 } } if (get_w2b_flag == 0) //如果小区间内没有找到, 将区间上下平移继续跟踪 { if
(upVector[upVector.size() - 1].y > upVector[upVector.size() - 2].y) //区间往下移动 { uchar *image2 = dstOut.data; image2 +=
step * (last_upPoint - N + 2 * N); int get_w2b_flag1 = 0; for (size_t i = 0; i < 2 * N; i++) { int pix[2]; pix[0] = image2[lie]; image2
+= step; //下一行 pix[1] = image2[lie]; if (pix[0] == 255 && pix[1] == 0) //白-黑跳变 { get_w2b_flag1 = 1;
upVector.push_back(Point(lie + Roi, last_upPoint - N + 2 * N + i)); last_upPoint = last_upPoint - N + 2 * N + i; } } if
(get_w2b_flag1 == 0) //如果还没找到, 继续向下移动 { uchar *image3 = dstOut.data; image3 += step * (last_upPoint - N + 4 *
N); for (size_t i = 0; i < 2 * N; i++) { int pix[2]; pix[0] = image3[lie]; image3 += step; //下一行 pix[1] = image3[lie]; if (pix[0] ==
255 && pix[1] == 0) //白-黑跳变 { upVector.push_back(Point(lie + Roi, last_upPoint - N + 4 * N + i)); last_upPoint =
last_upPoint - N + 4 * N + i; } } } } if (upVector[upVector.size() - 1].y < upVector[upVector.size() - 2].y) //区间往上移动 { uchar
*image2 = dstOut.data; image2 += step * (last_upPoint - N - 2 * N); int get_w2b_flag1 = 0;
常州大学本科生毕业设计 ( 论文 )

```

第 42 页 / 共 46 页 for (size\_t i = 0; i < 2 \* N; i++) { int pix[2]; pix[0] = image2[lie]; image2 += step; //下一行 pix[1] = image2[lie]; if (pix[0] == 255 && pix[1] == 0) //白-黑跳变 { get\_w2b\_flag1 = 1; upVector.push\_back(Point(lie + Roi, last\_upPoint - N - 2 \* N + i)); last\_upPoint = last\_upPoint - N - 2 \* N + i; } } if (get\_w2b\_flag1 == 0) //如果还没找到, 继续向下移动 { uchar \*image3 = dstOut.data; image3 += step \* (last\_upPoint - N - 4 \* N); for (size\_t i = 0; i < 2 \* N; i++) { int pix[2]; pix[0] = image3[lie]; image3 += step; //下一行 pix[1] = image3[lie]; if (pix[0] == 255 && pix[1] == 0) //白-黑跳变 { upVector.push\_back(Point(lie + Roi, last\_upPoint - N - 4 \* N + i)); last\_upPoint = last\_upPoint - N - 4 \* N + i; } } } }  
//再找下边缘, 边缘跟踪 for (size\_t lie = 0; lie < dstOut.cols; lie++) //x 方向 { uchar \*image1 = dstOut.data; image1 += step \* (last\_downPoint - N); int get\_w2b\_flag = 0; for (size\_t i = 0; i < 2 \* N; i++) { int pix[2]; pix[0] = image1[lie]; image1 += step; //下一行 pix[1] = image1[lie]; if (pix[0] == 0 && pix[1] == 255) //黑-白跳变常州大学本科生毕业设计 ( 论文 )

第 43 页 / 共 46 页 { get\_w2b\_flag = 1; downVector.push\_back(Point(lie + Roi, last\_downPoint - N + i)); last\_downPoint = last\_downPoint - N + i; } } if (get\_w2b\_flag == 0) //如果小区间内没有找到, 将区间上下平移继续跟踪 { if (downVector[downVector.size() - 1].y > downVector[downVector.size() - 2].y)

//区间往下移动 { uchar \*image2 = dstOut.data; image2 += step \* (last\_downPoint - N + 2 \* N); int get\_w2b\_flag1 = 0; for (size\_t i = 0; i < 2 \* N; i++) { int pix[2]; pix[0] = image2[lie]; image2 += step; //下一行 pix[1] = image2[lie]; if (pix[0] == 0 && pix[1] == 255) //黑-白跳变 { get\_w2b\_flag1 = 1; downVector.push\_back(Point(lie + Roi, last\_downPoint - N + 2 \* N + i)); last\_downPoint = last\_downPoint - N + 2 \* N + i; } } if (get\_w2b\_flag1 == 0) //如果还没找到, 继续向下移动 { uchar \*image3 = dstOut.data; image3 += step \* (last\_downPoint - N + 4 \* N); for (size\_t i = 0; i < 2 \* N; i++) { int pix[2]; pix[0] = image3[lie]; image3 += step; //下一行 pix[1] = image3[lie]; if (pix[0] == 0 && pix[1] == 255) //黑-白跳变 { downVector.push\_back(Point(lie + Roi, last\_downPoint - N + 4 \* N + i)); last\_downPoint = last\_downPoint - N + 4 \* N + i; } } } if (downVector[downVector.size() - 1].y < downVector[downVector.size() - 2].y)

常州大学本科生毕业设计 ( 论文 )

第 44 页 / 共 46 页

//区间往上移动 { uchar \*image2 = dstOut.data; image2 += step \* (last\_downPoint - N - 2 \* N); int get\_w2b\_flag1 = 0; for (size\_t i = 0; i < 2 \* N; i++) { int pix[2]; pix[0] = image2[lie]; image2 += step; //下一行 pix[1] = image2[lie]; if (pix[0] == 0 && pix[1] == 255) //黑-白跳变 { get\_w2b\_flag1 = 1; downVector.push\_back(Point(lie + Roi, last\_downPoint - N - 2 \* N + i)); last\_downPoint = last\_downPoint - N - 2 \* N + i; } } if (get\_w2b\_flag1 == 0) //如果还没找到, 继续向下移动 { uchar \*image3 = dstOut.data; image3 += step \* (last\_downPoint - N - 4 \* N); for (size\_t i = 0; i < 2 \* N; i++) { int pix[2]; pix[0] = image3[lie]; image3 += step; //下一行 pix[1] = image3[lie]; if (pix[0] == 0 && pix[1] == 255) //黑-白跳变 { downVector.push\_back(Point(lie + Roi, last\_downPoint - N - 4 \* N + i)); last\_downPoint = last\_downPoint - N - 4 \* N + i; } } } }

Mat srcImage = imread("2345.bmp", 1); //在原图 3 通道图中画出边缘

Mat srcImage1 = imread("2345.bmp", 1); for (int i = 1; i < upVector.size(); i++) { line(srcImage, Point(upVector[i - 1].x, upVector[i - 1].y + 25), Point(upVector[i].x, upVector[i].y + 25), Scalar(0, 0, 255), 3); } for (int i = 1; i < downVector.size(); i++)

常州大学本科生毕业设计 ( 论文 )

第 45 页 / 共 46 页 { line(srcImage, Point(downVector[i - 1].x, downVector[i - 1].y - 25), Point(downVector[i].x, downVector[i].y - 25), Scalar(0, 0, 255), 3); }

//机械坐标转换

FILE \*upFile;

FILE \*downFile; fopen\_s(&upFile, "Calibrate\_up.txt", "w"); fopen\_s(&downFile, "Calibrate\_down.txt", "w"); vector<Point2f> upVector\_Real, downVector\_Real; for (int i = 0; i < upVector.size(); i++) { Point2f temp; temp.x = \_affine\_calibrate\_.at<double>(0) \* upVector[i].x + \_affine\_calibrate\_.at<double>(1) \* upVector[i].y + \_affine\_calibrate\_.at<double>(2); temp.y = \_affine\_calibrate\_.at<double>(3) \* upVector[i].x + \_affine\_calibrate\_.at<double>(4) \* upVector[i].y + \_affine\_calibrate\_.at<double>(5); upVector\_Real.push\_back(temp); fprintf(upFile, "[%3f, %3f]\n", temp.x, temp.y); } for (int i = 0; i < downVector.size(); i++) { Point2f temp; temp.x = \_affine\_calibrate\_.at<double>(0) \* downVector[i].x + \_affine\_calibrate\_.at<double>(1) \* downVector[i].y + \_affine\_calibrate\_.at<double>(2); temp.y = \_affine\_calibrate\_.at<double>(3) \* downVector[i].x + \_affine\_calibrate\_.at<double>(4) \* downVector[i].y + \_affine\_calibrate\_.at<double>(5); downVector\_Real.push\_back(temp); fprintf(downFile, "[%3f, %3f]\n", temp.x, temp.y); } fclose(upFile); fclose(downFile);

//间隔 0.5mm 差分提取 for (int i = 0; i < upVector\_Real.size() - 1; i++) { while (abs(upVector\_Real[i].x - upVector\_Real[i + 1].x) < 0.5) { upVector\_Real.erase(upVector\_Real.begin() + i + 1); if (i == upVector\_Real.size() - 1) break; } } for (int i = 0; i < downVector\_Real.size() - 1; i++) { while (abs(downVector\_Real[i].x - downVector\_Real[i + 1].x) < 0.5) { downVector\_Real.erase(downVector\_Real.begin() + i + 1);

常州大学本科生毕业设计 ( 论文 )

第 46 页 / 共 46 页 if (i == downVector\_Real.size() - 1) break; } } vector<Point2f> \_Real\_Point\_; for (int i = 0; i < upVector\_Real.size(); i++) {

\_Real\_Point\_.push\_back(upVector\_Real[i]); } for (int i = 0; i < downVector\_Real.size(); i++) { \_Real\_Point\_.push\_back(downVector\_Real[downVector\_Real.size() - 1 - i]); }

FILE \*finalFile; fopen\_s(&finalFile, "totlePoint.txt", "w"); for (size\_t i = 0; i < \_Real\_Point\_.size(); i++) { fprintf(finalFile,



```
"[%3f, %3f]\n", _Real_Point_[i].x, _Real_Point_[i].y); } fclose(finalFile); return 0; }
```

```
3. 标定算法Mat Calibrate(float Col[], float Row[], float qx[], float qy[],int Length) { vector<Point2f> _image_piont_  
vector<Point2f> _real_piont_; for (size_t i = 0; i < Length; i++) {  
_image_piont_.push_back(Point2f(Col[i], Row[i]));  
_real_piont_.push_back(Point2f(qx[i], qy[i])); }  
Mat _affine_matrix_ = estimateRigidTransform(_image_piont_, _real_piont_, true); return _affine_matrix_; }
```

说明：1.总文字复制比：被检测论文总重合字数在总字数中所占的比例

2.去除引用文献复制比：去除系统识别为引用的文献后，计算出来的重合字数在总字数中所占的比例

3.去除本人已发表文献复制比：去除作者本人已发表文献后，计算出来的重合字数在总字数中所占的比例

4.单篇最大文字复制比：被检测文献与所有相似文献比对后，重合字数占总字数的比例最大的那一篇文献的文字复制比

5.指标是由系统根据《学术论文不端行为的界定标准》自动生成的

6.红色文字表示文字复制部分;绿色文字表示引用部分;棕灰色文字表示作者本人已发表文献部分

7.本报告单仅对您所选择比对资源范围内检测结果负责



 [amlc@cnki.net](mailto:amlc@cnki.net)

 <http://check.cnki.net/>

 <http://e.weibo.com/u/3194559873/>