

班 级 05121201

学 号 05121214

西安电子科技大学

本科毕业设计论文



题 目 基于树莓派的显微成像

系统

学 院 物理与光电工程学院

专 业 电子科学与技术

学生姓名 彭峰

导师姓名 刘杰涛

毕业设计（论文）诚信声明书

本人声明：本人所提交的毕业论文《基于树莓派的显微成像系统》是本人在指导教师指导下独立研究、写作成果，论文中所引用他人的无论以何种方式发布的文字、研究成果，均在论文中加以说明；有关教师、同学和其他人员对文本的写作、修订提出过并为我在论文中加以采纳的意见、建议，均已在我的致谢辞中加以说明并深致谢意。

本文和资料若有不实之处，本人承担一切相关责任。

论文作者：_____ (签字) 时间：2016年04月12日

指导教师已阅：_____ (签字) 时间：2016年04月12日

摘要

显微成像的应用开始在越来越多的领域中得到应用。光学薄膜的表面粗糙度对其光学特性的测量精度有较大影响。在常见的加工生产过程中，薄膜表面的毛刺、划痕等都可能对精密产品的性能造成影响。为提高产品的光学特性的精度，对相应产品出厂前表面进行实时的、高精度的缺损情况测试具有实际重要意义。使用电子仪器的手段相比于传统人力目测检验方法，具有单位效率高、判定率高、可靠性高等特点，便于元器件表面缺损测试方面向现代微型组件化智能化发展。

本文依托ARM处理器平台设计了数字化显微成像技术处理系统，并对零部件表面的缺损情况进行了实时测试。此检测系统具有智能化、小型化、图像采集方便实时的特点，可以进一步提高产品出厂前合格率、降低后续产品返修率

。整体上，采用索尼的堆栈式CMOS图像传感器IMX135，作为核心数字显微图像采集芯片，借助CSI-2接口将图像传至ARM主板，并利用ARM处理器完成显微图像的图像增强、图像识别等算法处理，并实现了显微图像的后续存储、归档和显示。存储、归档上可以放在存储卡或云端，通过HDMI或网络将图像传输到LED平板显示器上进行显示，达到实现系统集成化、桌面一体化、处理器实时处理的常见功能，同时备选两套方案，即可使用HDMI线缆将显微图像传输到桌面显示，利用以太网或802.11协议，借助路由也可以在异地通过互联网的方式，通过浏览器访问显示显微图像。更好实现产品缺损检测效果的远程实时控制功能。搭建了高精度的实时缺损检测平台，针对图像检测研究了显微图像的拼接。

本论文完成下面工作：

- 1.完成嵌入式显微成像系统的整体设计，包括光学显微成像子系统、图像显示处理子系统、上位机图像显示子系统等。
- 2.依据情况，设计显微图像采集的硬件设计方案、软件显示方案。分析技术要求、选型相应硬件，借助树莓派为主的ARM处理器平台，利用发布的CSI接口适配图像处理驱动和Web显示B/S方案，实时采集的图像通过CSI-2接口传输到基于ARM的主板，经过图像增强和图像滤波算法处理后，将相应图像显示出来。
- 3.对设计的ARM平台下的嵌入式显微成像系统进行相应的实验测试，在不同的光照强度下，对缺损的光学薄膜表面情况进行成像检验，改进系统不足，最后满足系统的整体设计要求。

摘要

关键词：数字显微成像 光学薄膜 ARM

Abstract

The acquisition and processing of image signal are used widely in the fields of scientific research and industrial production, medicine, biology and so on. At the same time, microscopic image acquisition can be the most popular basic of microscopic imaging system. Usually during the production of the film burrs, scratches can all have an impact on the performance of precision products. In order to improve the accuracy of the optical properties of the product, the corresponding products manufactured prior to the surface in real time, high accuracy defect testing is of practical importance. Compared to traditional, the way of Using electronic instruments means of human visual inspection methods, with more high efficiency, high rate of judgment, high reliability, and easy component to the modern surface defect testing tiny components of intelligence.

Based on ARM processor platform design for digital Microscopy Imaging Processing System, it can be used to detect the parts of surface in real time test. This detection system with intelligent characteristics, small size, convenient image acquisition in real time, can further enhance factory pass rate, reduce product failure rates. As a whole, Sony IMX135 CMOS image sensor, as the core of digital microscopic image chip, transfer image data via CSI-2 interface to ARM boards. And ARM processor is ready to complete micrograph image enhancement, image recognition algorithms, subsequent storage, archiving and display of images. About Storage, archive can be put in storage RAID or cloud, through HDMI cable or via network will transfer image signal to LED display, achieved system intergration, and desktop integration. Processor can process common function in real-time, and then transfer micro image signal to the desktop, by using Ethernet or 802.11 protocol. Through the Internet, this image can also be displayed in the browser.

This paper includes the following:

1. Completed a embedded microscopic imaging system with hardware design, software graphical user interface and mainly optical modules. This system includes some special modules such as image acquisition module, image processing module and image displayed modules.
2. According to the situation, we designed microscopic imaging plans for hardware de-

Abstract

sign, Web graphical user interface. According to the requirements of resolution system , we used raspberry pi as a based ARM platform and designed the micro image acquisition platform with the image sensor of SONY IMX135, The image data will be designed with the output interface of CSI-2 MIPI.

3.The experimental verification of the embedded micro imaging system using raspberry pi can carried out, and the accuracy of this based system is verified by different useful way.

Key Words: digital microscope microscopic imaging CMOS image
sensor ARM

目录

第一章 引言	1
1.1 选题背景及意义	1
1.1.1 选题研究内容	1
1.1.2 解决问题	1
1.2 国内外研究现状	2
1.2.1 显微成像国内外发展现状	2
1.2.2 光学薄膜检测的发展现状（未完成）	2
1.3 本文主要任务和安排	3
1.3.1 主要任务和安排	3
1.3.2 本章小结	4
第二章 显微成像系统整体方案设计	5
2.1 显微基本原理	5
2.1.1 光学成像原理（回去测试）	5
2.1.2 显微镜结构构造	6
2.2 显微成像处理平台器件选型	7
2.2.1 显微镜头物镜目镜选型（不开工）	7
2.2.2 图像传感器的选型	8
2.2.3 图像处理方案研究	9
2.2.4 数据传输接口介绍	10
2.3 系统平台选择搭建	11
第三章 显微图像处理平台系统设计	13
3.1 树莓派平台简介	13
3.2 图像传感器接口驱动程序	14
3.2.1 接口驱动简介	14
3.3 显微图像预处理	16
3.3.1 预处理图像算法研究分析	16

3.4 实时图像处理算法分析	17
3.5 上位机图像显示处理	18
3.6 本章小节	19
第四章 显微成像系统的检测实验	21
4.1 薄膜表面缺陷检测测试	21
4.1.1 实验仪器介绍和安装	21
4.1.2 缺陷检测结果分析（未完成）	22
4.1.3 本章小节	23
第五章 总结与展望	25
5.1 论文工作总结	25
5.2 未来发展展望	25
致谢	27
参考文献	29
附录 A 数据	31
A.1 代码	31

第一章 引言

1.1 选题背景及意义

1.1.1 选题研究内容

伴随着技术的革新和工艺上的不断进步，工业产品的表面精度检测要求不断提升，目前的缺陷检测方面技术开始越加复杂，各种的技术的工作方式和原理开始出现变更，体现在生活中的各个领域，高效的技术手段和实现方法直接决定了检测系统的可靠性和检测方法的便捷性。表面缺损一般是采用的加工方法和机器磨合有关联造成的，缺损的表象可能包括裂纹、气泡、毛刺等缺陷。目前，光学显微镜仍然是使用最广泛的显微镜。但是在同时，传统人工方式仍然存在不少的局限性，如：使用人员使用显微镜时，视野不畅，容易造成视觉疲劳；在观察时，无法多人观察进行交流；不能对图像进行必要的处理，加强所需目标的对比度，快速存储、归档、显示。

数字图像处理技术和图像处理算法的不断发展，能够结合传统光学仪器和计算机技术各自的优势，更好的实现多种数据采样和处理功能，实现系统平台的图像处理，包括去噪，模块识别等功能。实现对显微镜中的显微图像实时/远程的监测，可以在公网上进行相应图片的显示。同时，关于数据的归档，显示和报表可进行额外的分析和处理、综合。在显微成像的数字化平台上，国内外一些仪器厂已经开始尝试研制，部分已有商品化的产品出售。相比和计算机直接进行显微镜连接的情况，使用本文的平台可以更方便的在小空间内使用，具有系统集成化和小型化的特性。^{[1][2][3]}

1.1.2 解决问题

硬件成本的不断下降使曾经昂贵的技术使用更加广泛，显微成像的单一器具已经开始应用在生活的部分方面，在此基础的改进上，增加数据的处理模块，在不同指标不同场合的跟进测试，是目前的一个研究热点。

在此需求上，本文设计了基于树莓派的数字显微成像系统，尝试在缺陷材料上进行进一步测试。在设计中，选用了索尼的第二代BSI光照CMOS图像传感器IMX135，利用低压差分信号抗干扰来提高信噪比，使图像成像效果更佳。

1.2 国内外研究现状

1.2.1 显微成像国内外发展现状

显微镜的起源较早，但是在后期才发展快速。早在2500年前的《墨经》中，就已经有了凹透镜的记载，然而凸透镜的发展却一直没有进展。直到16世纪末期，荷兰的眼镜商詹森(Zaccharias Janssen) 和他的儿子尝试了一次将两个凸透镜放入到同一个镜筒中，发明了人类历史上第一个光学显微镜。而后在1609年，著名的天文学之父伽利略在这个实验的基础上，分解出相关的物理学原理，并依据自身的理论发明了更好聚焦的显微镜。

而后的列文虎克将放大倍数大大的提升了一截，第一次能够发现微观的生物和非生物现象，出版了极多的论文，同时极大的推进了生物学的发展。后之显微镜之父，罗伯特·虎克仿制了一台与列文虎克一样的显微镜，证实了水中微小微生物的发现，1830年，利斯忒 (Joseph Jackson Lister, 1786-1869) 用几个有特定间距的透镜组减小了球面像差，从而进一步改进了显微镜。而后，1872年德国的数学家和物理学家 Ernst Abbe 改进了玻璃的制造和生产工艺，改进并生产出有均一折射率的光学级玻璃。^{[4][5]}

受到衍射限制，光学显微镜的分辨率达到显微镜的分辨率极限，即：

$$R = \lambda / (n \cdot \sin(\alpha))$$

这里的 R 是分辨率， α 是孔径张角， λ 是波长， $n \cdot \sin(\alpha)$ 是数值孔径， n 是折射率。

1.2.2 光学薄膜检测的发展现状（未完成）

光学薄膜广泛应用于液晶显示屏、触摸屏、太阳能电池板等行业，对表面质量有极高的要求，如何保证其质量要求，必须要借助有效的的检测手段进行检测。与其他如金属、印刷品、玻璃制品等行业同类检测技术相比，由于薄膜的高透光性、缺陷尺寸极小、薄膜运行速度高，致使曝光时间短，CCD感光不足，图像整体偏暗，最终导致目标与背景的灰度级差极小。^[6]传统的人工目视检验或离线抽检方法劳动强度大、精度低、效率不高、可靠性差等缺点，无法适应现代化的生产要求。

起源

商业应用

检测

完善

重要性

1.3 本文主要任务和安排

1.3.1 主要任务和安排

硬件成本的不断下降使曾经昂贵的技术使用更加广泛，显微成像的单一器具已经开始应用在生活的部分方面，在此基础的改进上，增加数据的处理模块，在不同指标不同场合的跟进测试，是目前的一个研究热点。

在此需求上，本文设计了基于树莓派的数字显微成像系统，尝试在缺陷材料上进行进一步测试。在设计中，选用了索尼的第二代BSI光照CMOS图像传感器IMX135，IMX135有最高60fps，4208x3120的分辨率，小至1umx1um的像元尺寸，总像素为1300万，利用低压差分信号抗干扰来提高信噪比，使图像成像效果更佳。在研究过程中，更希望是显微成像方面在传统设备转向电子化，数字化，智能化，互联网化的一种可行性的尝试。进一步改善整体流程，加强后续的研究和信息处理的长尾方式，提高部署的高效性，降低生产成本。

具体章节结构 本文主要的内容设置如下：

第一章对本文的研究原理和背景做基本的阐述，希望解决的问题。并查阅相关显微成像和光学薄膜的国内外发展现状，在此基础上进一步改进，介绍本文的主要结构和目前安排。

第二章设计显微成像系统硬件部分，对显微成像的光学原理做出阐述，并进行目镜物镜的选型。图像传感器功能和类型性能比较选型，数据传输接口的比较，显示处理方式的比较和选择，设计出一个较完整的低耦合化方案。

第三章对显微图像处理平台进行系统设计，查阅研究了相关CSI的接口在树莓派的驱动程序，介绍V4L2框架的应用和使用。并分析图像算法的在预处理和处理上的研究，包括线性滤波、中值滤波、高斯滤波，边缘检测等方法，并研究了显微图像的相关拼接算法，设计了阈值分割和像素投影算法。最后在Web上展示处理后的图像，同时可以查阅前期归档后的图像。

第四章设计显微成像的实验测试系统，验证系统可行性，并依据照明光线变量验证可以达到的系统精度，满足系统的测试要求指标。

第五章为论文的总结和展望，总结主要研究学习工作，提出了可能存在的问题和未来的改进思路方向。

1.3.2 本章小结

本章首先介绍了选题的来由和解决的问题，简要说明了显微镜的发展史，介绍了

显微成像的光学结构和组成原理，同时简单说明了光学薄膜检测的问题和传统方式。在介绍基本情况之后，详细说明论文的架构安排、学习的内容和完成的主要工作。

第二章 显微成像系统整体方案设计

本显微成像检测系统由光学模块、图像处理模块、图像采集模块、图像归档显示模块等构成。在整体设计流程中,关键的技术包括图像传感器的分析与选择、图像主处理器及预处理器的选择等。本章在光学基础上,根据需求和情境重点分析了各个不同模块需要的硬件优势方面和选型,并对此进行较为详细的讲述。可见的一个分类的图2.1示如:

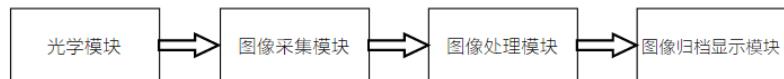


图 2.1: 模块介绍

2.1 显微基本原理

2.1.1 光学成像原理（回去测试）

显微镜的成像如图2.2所示,小物体AB在物镜的焦距之外,人眼在另一边的距离除观察, AB形成放大倒立的实像A'B',而这一实像正好在目镜的焦距以内的附近处,再一次经过目镜放大之后,在明视距离d处形成正立虚像A''B''^[7]。(一定要测试,找到位置,不然打脸了)

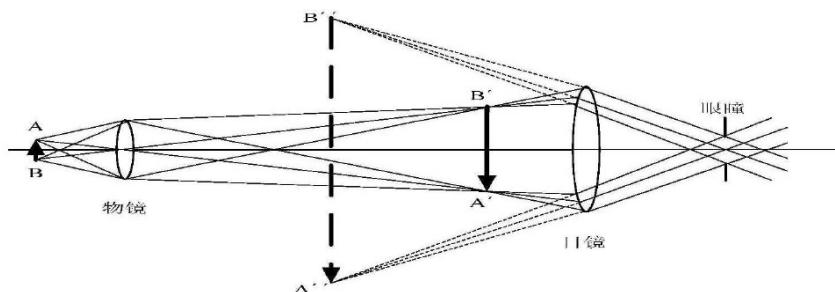


图 2.2: 显微镜成像光路图

显微镜的视觉放大率定义为:

$$J = l_0 \cdot d / f_1 f_2$$

式??中 d 是明视距离， l_0 为物镜到目镜的距离， f_1 为物镜的焦距， f_2 为目镜的焦距。此处的角放大率为物镜的线放大率和目镜的角放大率的乘积。数值孔径的孔径角的正弦与透镜和物体之间介质的折射率的乘积，而显微镜的分辨率与光的波长和物镜的数值孔径有关，波长越短，数值孔径越大，分辨率越大。^[8]

2.1.2 显微镜结构构造

普通光学显微镜的构造可分为两部分：一为机械装置，二为光学系统。机械装置由镜座、镜筒、物镜转换器、载物台、推动器、粗动螺旋和微动螺旋等部件组成。光学系统由目镜、物镜、聚光器、光源、滤光片、虹彩光圈等组成。一个典型显微镜如图2.3所示。

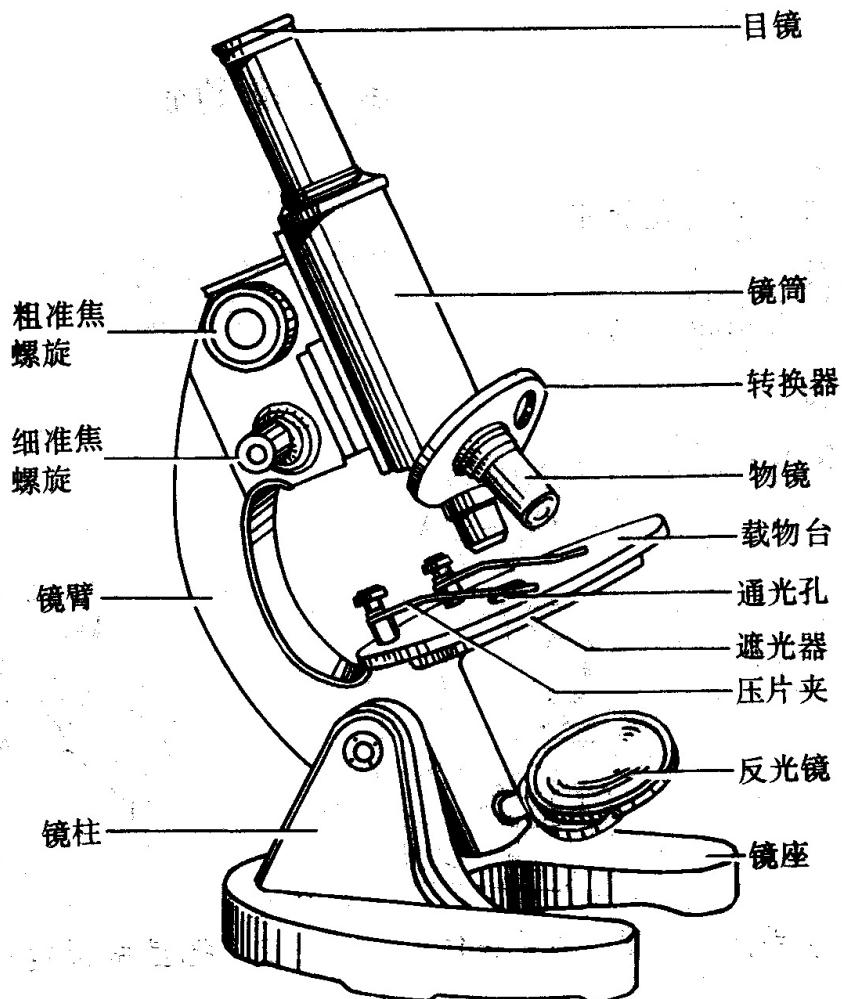


图 2.3: 典型显微镜结构图

物镜是决定显微镜性能的最重要部件，安装在物镜转换器上，接近被观察的物体，

故叫做物镜或接物镜。通常目镜由上下两组透镜组成，上面的透镜叫做接目透镜，下面的透镜叫做会聚透镜或场镜。聚光器也叫集光器。位于标本下方的聚光器支架上。它主要由聚光镜和可变光阑组成。其中，聚光镜可分为明视场聚光镜和暗视场聚光镜。反光镜是一个可以随意转动的双面镜，直径为50mm，一面为平面，一面为凹面，其作用是将从任何方向射来的光线经通光孔反射。平面镜上反射光线的能力较弱，适合在光线较强时使用，凹面镜反射光线的能力较强，适合在光线较弱时使用。

2.2 显微成像处理平台器件选型

2.2.1 显微镜头物镜目镜选型（不开工）

(1) 显微物镜的设计在本系统设计中，对测量精度要求较高，要求系统分辨本领达到 0.5m。本显微成像系统的分辨率由光学镜头的分辨率和 CMOS 摄像头的分辨率共同决定。系统中光学镜头部分对物体进行放大，先考虑光学系统的分辨率：上式中， n 为物空间介质折射率， λ 为照明光源的中心波长， δ 是系统的分辨率，因此在照明波长确定时，只能增大镜头的数值孔径来提升分辨率。整个系统光学放大率由系统要求的分辨率大小和 CMOS 的像元尺寸决定，如下式，式中， δ 是系统中 CMOS 芯片的最小可分辨尺寸， N 为系统的分辨率， A 为系统总的放大率。根据上述的设计分析，选用了消色差物镜中放大倍数中等的李斯特物镜。它的结构采用两个分离的双胶合透镜组，垂轴放大率达到 10 倍，数值孔径 (NA) 设计值为 0.61。李斯特型物镜的设计原则包括：(一) 各个双胶合透镜组具有相同的偏角，后组比前组的偏角略大。(二) 光阑位置处于第一个双胶合透镜处。采用两组双胶合透镜来抵消球差和慧差，物镜的总焦距就等于两组双胶合透镜之间的距离。前一个双胶合组的焦距两倍于物镜焦距。物镜的总焦距与第二个双胶合组焦距相同。(三) 采取两个双胶合透镜分别单独校正系统的球差、慧差和色差，这种设计的优点是采用两个双胶合透镜组合，组合在一起为一个中倍显微物镜，移去一个双胶合组后可作为低倍显微物镜[28]。按照物镜设计要求：物镜要求放大倍数为 10 倍，系统光源照明采用中心波长为 550nm 的光源，由式 $NA = n \sin u$ 计算可得，透镜数值孔径大小为 0.61，通过以上几个参数的确定，选出符合参数要求的镜片组合。选定透镜组合后采用 ZEMAX 对设计进行仿真，对光线进行追迹、计算像差，对设计不满意的参数，再次重新选择玻璃材料，重复上面的仿真计算[29]，直到达到设计要求。

2.2.2 图像传感器的选型

CMOS和CCD的比较

固体图像传感器已经广泛地应用于生活各处，包括像电子照相机、DV、智能监控、手机、平板和摄像头等。固体图像传感器是利用半导体材料的内光电效应原理制成的光电转换器件,依据工艺结构可以分为两大类:一类是电荷耦合器件(CCD)图像传感器;另一类是互补金属氧化物(CMOS)图像传感器。两种目前常见的图像传感器都是上世纪60年代开始研制,在当时由于CCD图像传感器灵敏度更高、噪声较低而成为当时图像传感器的主流。而CMOS图像传感器由于工艺上的提升限制,长时间未能摆脱光照灵敏度低、噪声无法下降和图像分辨率低等不足。于此同时, CCD图像传感器由于敏感元件和信号处理电路无法集成在同一芯片上,使得照相机体积大、功耗大。^[6]

进入新世纪,时间发展,随着互联网向移动互联网的转移,智能手机的发展十分迅速,由于CMOS图像传感器却有集成度高、体积小、功耗小和造价低等优点,十分适合在手机上进行集成,考虑到两种图像传感器的技术特点和缺点改进方式,随集成电路设计技术和迭代工艺水平的提高,CMOS图像传感器过去存在的缺点,现在开始逐渐进行技术攻关,常见的背照式结构给CMOS图像传感器在光线摄入上有长足的发展,而且它其固有的特点更是CCD器件所无法比拟的,因而它再次成为研究和工业需求的热点。

CCD和CMOS两者在多个维度上有着较大特性上的差异:

1.系统集成

CMOS图像传感器易于在SoC上集成,便于在同一芯片中同时进行内部的信号降噪、数据整合和数据处理,片上数据传输率高,与周边电路的整合性高,更可将与信号处理器整合在一起,如图所示,便于大幅度减小对应模块的主板面积占用。考虑到CCD和CMOS两者采用不同的制造工艺,所以CCD难以装入SoC。因此,一般认为CMOS图像传感器可以广泛使用在各个嵌入式领域。

2.性能特性

在实际应用方面,性能特性是进行芯片选型取舍的极为重要的因素。CMOS图像传感器由于多个放大器的存在,不同一组生产的放大器放大情况有细微差异,导致最终的图像输出噪声较多。此外,由于集成度高的缘故,各元件、电路之间距离较近,相互之间的光、电、磁干扰显得严重,噪声对图像质量影响很大。在需要的电源数方面,相对于CCD图像传感器需要三或四组电源,CMOS图像传感器则只需一个即可。此外CMOS图像传感器利用3.3V电源即可驱动,相比之下电源消耗量比CCD图像传感器低。因此,在功耗和电压方面,CMOS图像传感器比CCD图像传感器在小型化嵌入式设备中有更大的优势。此外, CMOS信号读取的方式较为简易, 电路设计相应也可简单。

3. 可靠性

两种图像传感器在商用及工业应用领域具有等价的可靠性。在极端恶劣的应用环境中,由于CMOS图像传感器充分提高系统集成度、降低模块间耦合,借助设计良好的封装和焊接技术,其可靠性优于CCD图像传感器。

图 2.4: 类型比较

	CMOS图像传感器	CCD图像传感器
感光灵敏度	低	高
噪声电子数	200	50
电路集成度	高	低
工艺难度	小	大
成本	低	高
功耗	低	高

图像传感器的选择

在图像传感器的选择上,需要考虑成像质量,分辨率,集成度,接口匹配,适配度等几项重要指标。由上面的比较可以知道, CMOS图像传感器的集成度远远优于CCD图像传感器。CMOS图像传感器可将多个处理器,控制器模块集成到一块芯片上,直接输出数字信号,减小开发难度。而在成像质量,分辨率上,两者均能达到要求,最关注的集成性,低功耗,采样速度高等特点, CMOS图像传感器占有很大的优势。

在考虑到成本问题之后,决定选用CMOS传感器作为图像采集的模块。这里经过多次比对,决定选择SONY的IMX系列传感器。SONY最近数年在CMOS技术上的大幅度改进使得CMOS传感器的图片质量开始接近数码相机的质量。尤其是背照式技术和堆栈式技术的使用,让CMOS图像传感器进一步接近CCD图像传感器的功能。最后考虑选用IMX135进行相关实验。

2.2.3 图像处理方案研究

实时图像处理系统要求系统需要在有限的时间内完成大量数据的运算。可尝试实现的实现有下面几种: 基于PC的数据处理, 基于DSP的数字信号处理, 基于ARM的通用事务处理。

(1) 在PC上的数据处理: 是常见的图像处理方式, 利用高级编程语言进行算法编程, 但在实时图像数据处理上可能需要占用CPU大量的浮点计算能力, 然而CPU更适合进行整数计算, 在PC中大量的浮点运算通常放在GPU中运行。

(2) 基于DSP的数字信号处理: 通用可编程的DSP在嵌入式领域中有数字信号处

理精度高，速度快，一定的编程性等优势，相对而言算法较为复杂，在高速图像处理中一定优势。

(3) 基于ARM的数据处理：随着VLSI技术的不断进步，ARM在近年来在嵌入式领域上的使用比率逐渐上升，市场上借助手机的快速发展ARM的出货量和迭代速度变更极快。此外，ARM有比较强的事物管理功能，可利用通用的语言和驱动，也可方便设计图形界面和应用程序。进一步的，随着硬件功能的不断加强，ARM系列中Cortex-M0处理器开始能以较高的性价比和单片机进行比较。

下面表2.5将三者的情况作出下面的对比。

图 2.5: 类型比较

	主要实现方法	速度	性价比	应用优势
ARM	高级编程语言	较快	高	通用软件
DSP	专用指令	快	中等	高精度计算
PC	高级编程语言	中等	低	通用软件

综合上述因素，探究ARM有着良好的事物管理的功能，可以实现较多的PC端的功能，内容和课扩展性比较大。DSP主要用于高精度计算，如加密解密，调制解调等，有着极好的数据处理能力和较高的运行速度。PC端可以通过通用型计算的特点实现要求功能，但在实际使用中，GPU的调用功能不明显，CPU的浮点运算能力较低，如添加显卡则又一步增加了成本和功耗，且不利于实现小型化，功能化的特点。加之现在的趋势是单一的DSP功能正在被有较强信号处理功能的ARM取代，考虑使用ARM。

2.2.4 数据传输接口介绍

如图2.6所示，树莓派包含的接口很多，包括USB电源接口、SD卡接口、GPIO、 I^2C 、USB2.0接口、CSI-2接口、HDMI接口、SVideo接口、以太网接口、3.5mm音频接口等。更可以通过USB接口连接分线器和鼠标键盘摄像头等实现扩展功能^[9]。

CSI属于MIPI标准之下。MIPI是一个比较新的标准，其规范也在不断修改和改进，目前比较成熟的接口应用有DSI(显示接口)和CSI（摄像头接口）。CSI/DSI分别是指其承载的是针对Camera或Display应用，都有复杂的协议结构。CSI-2是一个单或双向差分串行界面，包含时钟和数据信号。CSI-2的层次结构：CSI-2由应用层、协议层、物理层组成。由于串行接口一般采用差分结构，利用几百mV的差分信号，在收发端之间传送数据。串行比并行相比：更节省PCB板的布线面积，增强空间利用率；差分信号增强了自身的EMI抗干扰能力，同时减少了对其他信号的干扰；低的电压摆幅可以做到更高的速度，更小的功耗。

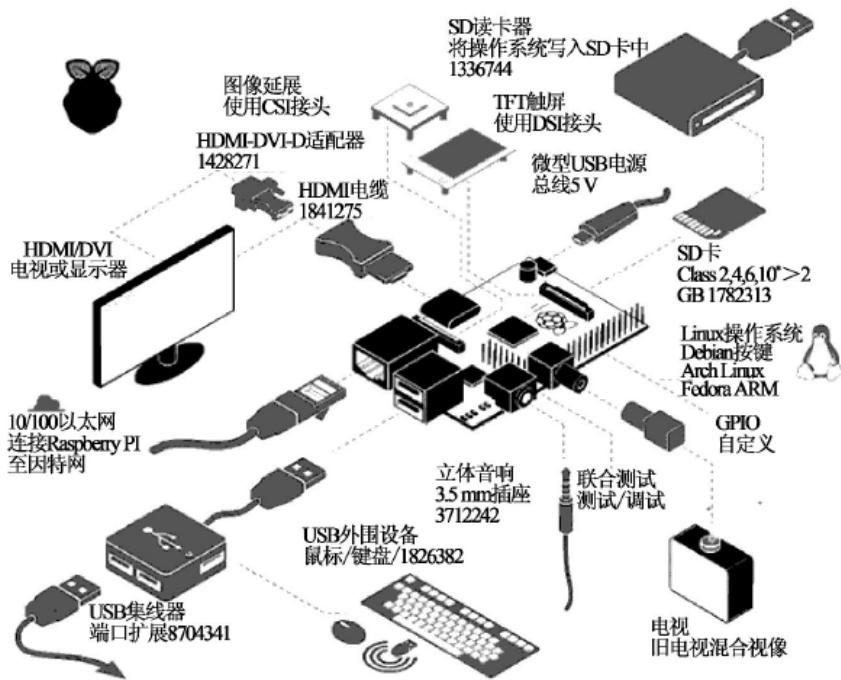


图 2.6: 树莓派基本硬件接口

USB是连接电脑系统与外部设备的一种串口总线标准，也是一种输入输出接口的技术规范，被广泛地应用于个人电脑和移动设备等信息通讯产品，并扩展至摄影器材、数字电视（机顶盒）、游戏机等其它相关领域。目前含有多种型号。

以太网（Ethernet）是一种电脑局域网技术。IEEE组织的IEEE 802.3标准制定了以太网的技术标准，它规定了包括物理层的连线、电子信号和介质访问层协议的内容。以太网是目前应用最普遍的局域网技术，取代了其他局域网标准如令牌环、FDDI 和 ARCNET。这里用来连接互联网。

高清多媒体界面（HDMI）一种全数字化影像和声音发送接口，可以发送未压缩的音频及视讯信号。HDMI可以同时发送音频和视讯信号，由于音频和视讯信号采用同一条线材，大大简化系统线路的安装难度。通过HDMI接口方便的连接到显示器进行图像显示。

2.3 系统平台选择搭建

常见的架构方式包括B/S架构和C/S架构。B/S(Browser/Server)架构,是浏览器和服务器的通信连接模式。在这一结构下,用户通过浏览器完成绝大多数的显示和回调，数据处理和归档还是主要通过服务器实现。这样可以大大简化的任务,减轻了系统维护与升级的成本,为用户的使用带来方便。但是,考虑到网络的时延等,该方案有着系统运行速度和稳定性不够高的局限性。C/S(Client/Server)架构,即客户机和服务器的通信连接

模式。它属于应用软件系统,充分利用客户端和服务器端的硬件环境,较为常见的是将具体的运算和数据的处理被放在客户端,从而使客户端变得很“胖”,一般称为“胖客户机”;相对地,服务器端的任务较轻,称为“瘦服务器”,和操作系统的连接较为紧密^[10]。

相对于C/S模式, B/S模式显著的特点是浏览的跨平台操作,方便使用,考虑到跨平台的优势,目前越来越多的平台功能特性放在B/S架构下^[11]。

第三章 显微图像处理平台系统设计

这一图像处理平台使用了基于 RESTful 架构的开放 API 体系，通过统一的 Web 服务展现给用户，借用 PaaS 平台完成部分存储和归档功能。借助浏览器获取相关的信息。借助树莓派较多的硬件接口，充分调用相关功能^[12]。

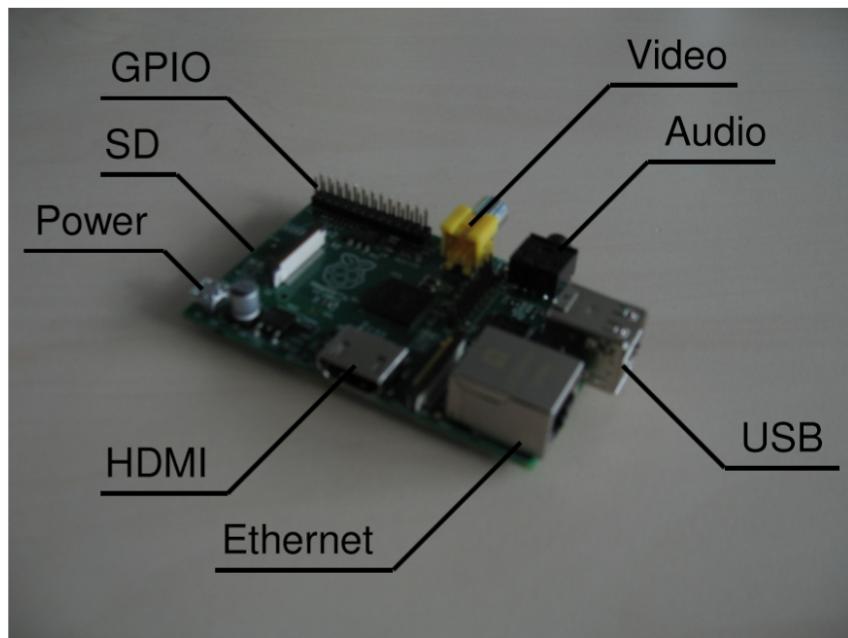


图 3.1: 树莓派实物和接口

3.1 树莓派平台简介

Raspberry Pi(中文名为“树莓派”，简写为 RPi，或者 RasPi/RPi)是为学生计算机编程教育而设计，只有信用卡大小的卡片式电脑，其系统基于 Linux。树莓派由注册于英国的慈善组织“Raspberry Pi 基金会”开发，Eben Upton 埃·厄普顿为项目带头人。2012 年 3 月，英国剑桥大学埃本·阿普顿 (Eben Epton) 正式发售世界上最小的台式机，又称卡片式电脑，外形只有信用卡大小，却具有电脑的所有基本功能，称为 Raspberry Pi 电脑板，中文译名“树莓派”。这一基金会以提升学校计算机科学及相关学科的教育，让计算机变得有趣为宗旨。基金会期望这一款电脑无论是在发展中国家还是在发达国家，会有更多的其它应用不断被开发出来，并应用到更多领域^{[13][14]}。

通过装载相应的 Linux 系统和相应的应用程序，RaspberryPI可以实现强大的通用能力，还具有廉价、体积小等优点。以硬件开源的嵌入式系统树莓派作为应用开发平台，通过TCP /IP 协议实现了数据查看的便捷化；便于联网的浏览器可以随时监看图片。

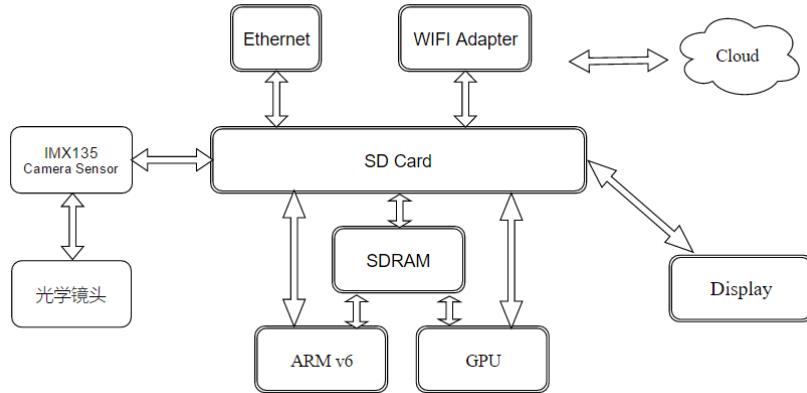


图 3.2: 显微平台数据流向

整体的数据流图如图3.2所示，拍摄物体经过光学镜头聚焦成像处理后，相应成像信息反映到IMX135的CMOS传感器中，经过CPU和GPU的处理后，存储在SD卡中，后联网存储备份归档在云中，同时还输出桌面显示，通过浏览器的方式完成图像的浏览和输出。

3.2 图像传感器接口驱动程序

3.2.1 接口驱动简介

V4L(Video4Linux)是为市场现在常见的电视捕获卡和并口及USB口的摄像头提供统一的编程接口。同时也提供无线电通信和文字电视广播解码和垂直消隐的数据接口。V4L2(Video for Linux two)，是V4L的改进版。V4L2从编程的角度来看，作为Linux内核中关于视频设备的框架系统，对于上层应用提供了一套统一的接口来访问底层的视频设备，起到抽象底层硬件的差异的作用。向下面向视频采集设备的驱动开发定义了一套开发规范，同时提取出公共代码避免代码冗余，将所有的视频采集设备的驱动程序都纳入其的管理之中。V4L2不仅给驱动程序编写者带来极大的方便，同时也方便了应用程序的编写和移植，在远程会议、可视电话、视频监控系统和嵌入式多媒体终端中都有广泛的应用。^[15] Linux系统V4L2的能力可在Linux内核编译阶段配置，默认情况下都有此开发接口。V4L2从Linux 2.5.x版本的内核中开始出现。

应用程序通过V4L2接口采集视频数据分为五个步骤：

首先，打开视频设备文件。

其次，进行视频采集的参数初始化，通过V4L2接口设置视频图像的采集窗口、采集的点阵大小和格式；

第三，申请若干视频采集的帧缓冲区，并将这些帧缓冲区从内核空间映射到用户空间，便于应用程序读取/处理视频数据；

第四，将申请到的帧缓冲区在视频采集输入队列排队，并启动视频采集；

第五，驱动开始视频数据的采集，应用程序从视频采集输出队列取出帧缓冲区，处理完后，将帧缓冲区重新放入视频采集输入队列，循环往复采集连续的视频数据；

第六，停止视频采集。

第七，关闭视频设备文件。

简要查看下对结构体 $v4l2_format$ 的情况，其由type和联合体fmt构成，来描述视频设备当前行为和数据的格式。 $v4l2$ 的struct展示如下：

代码 3.1: V4L2 struct format Code

```

1  /**
2   * struct v4l2_format - stream data format
3   * @type:    type of the data stream
4   * @pix:     definition of an image format
5   * @pix_mp:   definition of a multiplanar image format
6   * @win:     definition of an overlaid image
7   * @vbi:     raw VBI capture or output parameters
8   * @sliced:   sliced VBI capture or output parameters
9   * @raw_data:  placeholder for future extensions and custom formats
10  */
11 struct v4l2_format {
12     enum v4l2_buf_type type;
13     /* V4L2_BUF_TYPE_VIDEO_CAPTURE */
14     union {
15         struct v4l2_pix_format pix;
16         /* V4L2_BUF_TYPE_VIDEO_CAPTURE */
17         struct v4l2_pix_format_mplane pix_mp;
18         /* V4L2_BUF_TYPE_VIDEO_CAPTURE_MPLANE */
19         struct v4l2_window win;

```

```

20      /* V4L2_BUF_TYPE_VIDEO_OVERLAY */
21      struct v4l2_vbi_format vbi;
22      /* V4L2_BUF_TYPE_VBI_CAPTURE */
23      struct v4l2_sliced_vbi_format sliced;
24      /* V4L2_BUF_TYPE_SLICED_VBI_CAPTURE */
25      __u8 raw_data[200];
26      /* user-defined */
27 } fmt;
28 };

```

3.3 显微图像预处理

3.3.1 预处理图像算法研究分析

预处理的目的是为了消除过度曝光、不合理的对比度、图像扭曲以及噪音等因素。在图像输出的部分，由于之前所说的因素，图像最终往往难以清晰。为了弥补，保证处理算法对指纹图像具有足够的鲁棒性，图像增强是其中需要的一点。图像增强是对图像采用一定的算法进行处理，使其缺陷的结构清晰，进而突出和保留固有的缺陷图像而避免与噪声混淆。^[16]

线性插值

线性内插值算法描述如下：对于一个目的像素，设置坐标通过反向变换得到的浮点坐标为(i+u,j+v)（其中i、j均为浮点坐标的整数部分，u、v为浮点坐标的小数部分，是取值[0,1]区间的浮点数），则这个像素得值 $f(i+u,j+v)$ 可由原图像中坐标为 (i,j)、(i+1,j)、(i,j+1)、(i+1,j+1) 所对应的周围四个像素的值决定，即：

$$f(i + u, j + v) = (1 - u)(1 - v)f(i, j) + (1 - u)v f(i, j + 1) + u(1 - v)f(i + 1, j) + uv f(i + 1, j + 1)$$

其中 $f(i, j)$ 表示源图像 (i, j) 处的像素值，以此类推。如图所示。

$$f(x, y_1) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{11}) + \frac{x - x_1}{x_2 - x_1} f(Q_{21})$$

$$f(x, y_2) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x - x_1}{x_2 - x_1} f(Q_{22})$$

中值滤波法

中值滤波法是一种非线性平滑技术，它将每一像素点的灰度值设置为该点某邻域窗口内的所有像素点灰度值的中值。中值滤波是基于排序统计理论的一种能有效抑制噪声的非线性信号处理技术，中值滤波的基本原理是把数字图像或数字序列中一点的值用该点的一个邻域中各点值的中值代替，让周围的像素值接近的真实值，从而消除孤立的噪声点。方法是用某种结构的二维滑动模板，将板内像素按照像素值的大小进行排序，生成单调上升（或下降）的二维数据序列。由此二维中值滤波输出为 $g(x,y) = \text{medf}(x-k,y-l), (k,l \in W)$ ，其中， $f(x,y)$ ， $g(x,y)$ 分别为原始图像和处理后图像。 W 为二维模板，通常为33，55区域，也可以是不同的形状，如线状，圆形，十字形，圆环形等。

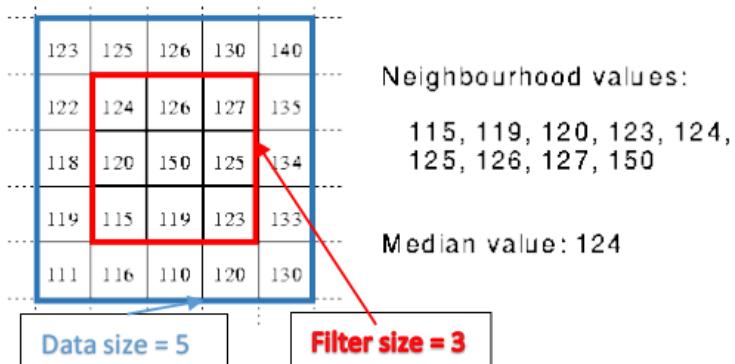


图 3.3: 中值滤波三三法

通过上面的介绍，我们简单的了解了图像预处理中的滤波算法。通过比较两种滤波算法，在 ARM 中进行图像预处理时，可以看出，虽然中值滤波算法复杂度更大，但其滤波的效果更佳，同时树莓派中的 GPU 运算速度又很高，因此采用中值滤波进行图像预处理，可以在 GPU 内部进行数学运算，这对于提高图像处理能力提供了巨大帮助，因此在本项目中，由于均值滤波会使图像的边缘信息丢失，不利于显微系统精度检测的提高，因此采用了中值滤波算法对该系统的图像做了预处理。

3.4 实时图像处理算法分析

由于本系统是靠显微镜来保证对准精度,进而提高测量精度的,因此,受显微镜观测视场的限制,无法直接观察大尺寸范围的物体。为了实现大尺寸样品的观测,就需要采用图像拼接技术,将采集到的多幅图像进行拼接,在进行图像融合时,设定一个图像局部清晰度的判定标准,本文中采用图像灰度均匀值作为标准,选择采集的比较高清晰的显微图像,尽量减少原图质量对融合效果的影响,再对显微图像进行区域分块,尽量将

图像清晰和模糊的区域分割开来。将观测视场临近的两幅图像分割出来的清晰区域块进行带权值的像素融合，以此类推，扩展到将所有的视场相邻的显微图进行融合，从而得到一个较大视场内的显微图像。^[17]

图像分割在对象定位、对象识别、视觉跟踪、图像检索和图像理解等各种应用中属于至关重要的步骤，其准确率和执行效率直接影响到后续步骤中目标特征提取和识别任务的效果。根据实现策略不同，图像分割算法主要有阈值法、模板匹配、聚类、活动边缘模型、图切割模型和区域增长等几种。这几种算法在特定的应用领域都有各自的优势特点，但均很难在分割正确率和执行效率两个衡量图像分割方法优劣的核心指标上同时都取得最佳性能，往往只能侧重其中之一，通常都是在两者之间寻找某种折衷。这里选用阈值分割来衡量匹配的阈值节点^[18]。

在对光学薄膜表面缺陷检测时，把图像中的缺陷进行阈值分割十分必要，一幅图像通常由前景区域和背景区域组成。图像分割的目的是将背景区域和前景区域分开，提高特征提取的准确性，同时节省处理时间，从而提高整个系统的性能。若选取的分割阈值过高的话，图像缺陷会被掩盖在背景中，会导致图像表面缺陷的漏检；相反，若阈值选取太低的话，一部分背景就会被误判为图像存在的缺陷。图像的阈值分割利用了图像中要提取的目标物与其背景在灰度特性上的差异，把图像视为具有不同灰度级的两类区域（目标和背景）的组合，选取一个合适的阈值，以确定图像中每一个像素点应该属于目标还是背景区域，从而产生相应的二值图像^[19]。

3.5 上位机图像显示处理

树莓派预搭载的编程开发环境是 Python 语言。Python 是一种面向对象、直译式计算机编程语言，已有近 20 年的发展历史，包含了一组完善而且容易理解的标准库，能够轻松完成很多常见的任务，语法简捷、清晰，且具有垃圾回收功能，能够实现内存的自动管理，常用于处理系统管理任务和网络程序编写，也适合完成各种高级任务。Python 经常被用于 Web 开发，Python 对于各种网路协定的支援很完善，因此经常被用于编写服务器软件、网路蠕虫。第三方函式库 Twisted 支援非同步线上编写程式和多数标准的网路协定（包含客户端和服务器），并且提供了多种工具，被广泛用于编写高性能的服务器软件。另有 gevent 这个流行的第三方库，同样能够支持高性能高并发的网络开发。^[20] 在显示上主要考虑下面的几种框架或库^[21]。

Django Django 是一个开放原始码的 Web 应用框架，由 Python 写成。採用了 MVC 的软体设计模式，即模型 M，视图 V 和控制器 C。使用 Django 最大的好处就是包括资料

库的连接、表单、登入系统、管理界面等等建构网站应该要有的元件都是 Django 本身的一部份，虽然乍看之下失去了一些弹性，但是却可以加快开发效率，简化组件复用工作量。

Qt Qt是一个跨平台的C++应用程序开发框架。广泛用于开发GUI程序，这种情况下又被称为部件工具箱。也可用于开发非GUI程序，比如控制台工具和服务器。Qt使用标准的C++和特殊的代码生成扩展（称为元对象编译器（Meta Object Compiler, moc））以及一些宏。

GTK+ GTK+最初是GIMP的专用开发库（GIMP Toolkit），后来发展为Unix-like系统下开发图形界面的应用程序的主流开发工具之一。GTK+是自由软件，并且是GNU计划的一部分。GTK+的许可协议是LGPL。GTK+使用C语言开发，但是其设计者使用面向对象技术。也提供了C++（gtkmm）、Perl、Ruby、Java和Python（PyGTK）绑定，其他的绑定有Ada、D、Haskell、PHP和所有的.NET编程语言。

考虑到目前多类型的跨客户端的平台化趋势，注重系统的可迁移，决定在B/S架构的基础上使用默认的编程开发环境Python，并使用Django框架降低开发难度，不考虑CGI的从头书写，和本地的系统C/S平台，如需要直接在系统平台内操作系统上的文件管理查看即可^[22]。

3.6 本章小节

本章主要简单介绍了树莓派平台的基本结构，常见应用。分析了驱动程序V4L2的特性和使用，了解图像预处理的方式，包括线性插值、中值滤波和高斯滤波等。了解了图像处理中的聚类和分割。最后对图像显示的几种常见的方法做了介绍，最后选定使用Django的框架来进行快速的开发。

第四章 显微成像系统的检测实验

通过搭建实验检测平台，对光学薄膜的表面缺陷通过进行观测检测验证。对照本显微成像检测系统的原理框图搭建实验平台，通过显微成像实验平台，实验完成对照明光源的位置设置、显微图像采集格式的设置、树莓派接口通信的调试，以及对图像数据采集传输与控制进行测试，验证了光学薄膜表面缺陷检测系统具备可行性。

4.1 薄膜表面缺陷检测测试

4.1.1 实验仪器介绍和安装

该平台分为两部分安装，一部分为实验装置的安装，后对系统烧录后进行驱动安装和图像测试，测试成功后安装对应的Django框架和相应程序。

硬件基础环境搭建 首先组装实验装置平台，把设计的显微镜头和设计的 CMOS 驱动板通过 CS 口连接起来，CMOS 驱动板的数据输出采用 FFC 软排线连接，组装好后的镜头如图 5.1 所示。显微系统的照明光源采用环形白色 LED 的反射式照明，根据外界不同的环境光照，可以调节照明 LED 灯的亮度，以适应不同透明度的薄膜的照明要求。采集不同光照强度下的显微图像，选取具有最佳对比度的图像作为最终需要识别处理的图像。

其次是树莓派系统的启动。首先通过 SD 卡将 Raspbian 系统写入树莓派中，插入 SD 卡后，系统会自动启动。使用ssh客户端软件或者命令行通过路由连接树莓派，startx后进入 pi 的登录界面，输入账号和密码，Raspbian 的默认账号和密码分别为 pi 和raspberry。

树莓派摄像头的配置及启动调试：首先进行摄像头的硬件设置，RPi 固件和 raspi-config 已为摄像头更新，接着通过执行apt-get 系列升级和补丁命令来进行系统更新，然后重启树莓派即可。最后通过相关摄像头命令sudo raspi -config 使pi 摄像头的CSI接口生效，选择 Enable Camera 回车。摄像头使能后，在树莓派官网上根据需求下载相应camera驱动和二进制包进行测试。

软件系统设置（未完成） 需要进行Django的安装和Server机的安装，

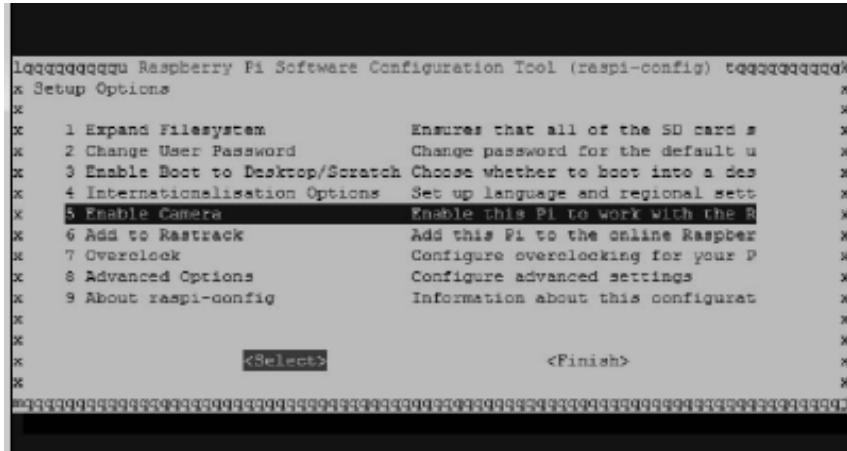


图 4.1: 安装系统界面

搭建的实验系统平台如图所示，连接显微镜头和 CMOS 采集板，采用反射式的照明方式，打开上位机软件界面，并调节好显微镜焦距，观测镜片表面的薄膜，如图所示。

4.1.2 缺陷检测结果分析（未完成）

在显微成像系统测试中，选取了实验室最常用的光学透镜的表面薄膜，观测了增透膜和增反膜表面的缺陷程度，对表面出现的缺陷进行了分析。调整不同的光照方式，对同一片增透膜进行了多次检测，以消除光照对检测误差的影响，通过采集不同的薄膜显微图像，并对传输上来的图片进行了图像预处理，采用均值滤波算法滤除高斯噪声，并对图像进行二值化处理，对采集的多幅图像进行比较选择，排除对焦不同等差异条件。采集到的薄膜表面的显微图像原始数据。在上位机上对显微图像采用中值滤波进行图像降噪预处理，处理结果如图所示。采用阈值分割处理算法对预处理过的图像做缺损检测处理，选定标准完整的薄膜表面图像作为标准背景图，将显微系统采集到的带有缺损的薄膜表面图像与之对比，对缺陷图像进行阈值分割处理，并截取放大显示识别出的典型的缺陷形状，选取记录结果如下所示。

在对薄膜的缺陷检测实验中，CMOS 相机采集的速度在 1080P 格式下可以达到每秒 30 帧，通过 CSI 接口传到显示器，实现了对薄膜的实时检测和识别，还能够通过网口传到局域网实现远程缺陷检测。相比传统的缺陷检测系统，整个缺陷检测系统硬件上采用了性价比的 ARM 处理器，选用手机上常用的 CMOS 图像传感器 IMX135，并通过 CSI-2 串行接口输出图像，实现了高速实时的检测特点。整个检测系统结构紧凑，Raspbian 操作系统可更新升级，实现了微型化智能化的检测特点。

4.1.3 本章小节

本章搭建了基于树莓派处理器的显微成像系统平台，选用不同光照强度的光源进行照明，测试了实验室的光学薄膜表面的缺陷，验证了该系统的实时检测性能，并对采集的薄膜表面的图像进行处理，识别出缺陷的类型和形状，达到了系统的总体设计要求。

第五章 总结与展望

5.1 论文工作总结

对光学薄膜表面存在的缺陷进行检测，采用传统显微检测的方法不再适应现代化工艺的要求。本文设计出一种基于 ARM 的显微成像系统。

(1) 对本文的研究原理和背景做基本的阐述，希望解决的问题。并查阅相关显微成像和光学薄膜的国内外发展现状，在此基础上进一步改进，介绍本文的主要结构和目前安排。

(2) 设计显微成像系统硬件部分，对显微成像的光学原理做出阐述，并进行目镜物镜的选型。图像传感器功能和类型性能比较选型，数据传输接口的比较，显示处理方式的比较和选择，设计出一个较完整的低耦合化方案。

(3) 对显微图像处理平台进行系统设计，查阅研究了相关CSI的接口在树莓派的驱动程序，介绍V4L2框架的应用和使用。并分析图像算法的在预处理和处理上的研究，包括线性滤波、中值滤波、高斯滤波，边缘检测等方法，并研究了显微图像的相关拼接算法，设计了阈值分割和像素投影算法。最后在Web上展示处理后的图像，同时可以查阅前期归档后的图像。

(4) 设计显微成像的实验测试系统，验证系统可行性，并依据照明光线变量验证可以达到的系统精度，满足系统的测试要求指标。

5.2 未来发展展望

本显微成像对薄膜的检测系统还处于初级研究阶段，但本显微成像检测系统还存在着诸多的不足，未来还需对系统进行进一步的深入研究。改进方向体现在如下几点：

(1) 在显微成像的 CMOS 相机方面，对图像的采集速度还有一定提升的空间，后期还可以将帧频提升至 60 帧每秒，以提高实时检测方面的精度。对相机镜头的加工工艺也有进一步提升的空间。

(2) 在对图像处理进行算法分析方面，文中的图像的滤波插值算法和融合分割方法虽然取得了预期效果，但是还存在进一步提升的空间，可以进一步提升算法的智能化，加强数据的处理和算法的效率。

(3) 图像web显示界面还有一定的优化空间，包括后期的数据归档和存储都可以改进，如定期上传不常用数据到云中归档，定期调用数据进行数据迁移。

(4) 在显微图像缺陷的模式识别方面，对光学薄膜表面缺陷图像模式处理的智能化识别还需要进一步研究，可以针对图像缺陷的生成机制进行深入研究，对不同类型的缺陷进行更好优化显示和识别，借此提高该检测系统对光学薄膜缺陷图像的识别率。

致谢

时光飞逝，转眼间本科阶段的学习和生活即将划上句号，4年的本科学习虽然辛苦，但能够学到知识却使我倍感快乐！在论文撰写完成之际，深深地感谢这段时间里给予我关心和帮助的许多人。

我首先要感谢我的导师刘杰涛讲师，在攻读学士学位期间，刘老师在某某等方面都给予我无私的教诲、悉心指导和无微不至的关心，他时时关注我的课题研究进度，并敦促我及时完成我的毕业论文。刘老师治学态度严谨、学识渊博，这种严谨的做学问的态度对我未来的学习、工作和生活都将产生深远的影响。感谢我的学长某某某，某某某，某某某等人陪我一起走过什么鬼阶段，大家在一起上课学习交流的时光让我终生难忘。更要感谢他们在学术上给予我的帮助和指导，使我能够顺利地完成论文。

感谢实验室全体同窗们的热情帮助和支持，感谢——等人共同营造的和谐、浓厚的学术氛围，和他们在一起学习使我感到无比快乐，和他们的讨论使我受益匪浅。感谢我的其他同学和其他朋友，正是因为有了各位的扶持和陪伴，我的生活才能如此的丰富、充实。

感谢在座的各位领导和同事在此期间给予我充分的关心、帮助、理解和支持，对此表示由衷的感谢。感谢物理与光电工程学院的很多老师在我学习期间和课题研究过程中给予我的热心指导和无私帮助，谢谢你们！

在我本科学习期间，要特别感谢我的家人尤其是我的父母给予我的无私帮助，正是他们对我的理解、支持和关心，才使我能够静下心来地学习、轻松地生活，良好地完成学业。

寥寥数语难以充分表达我的感激之情，最后在这里祝大家万事如意、身体安康。

参考文献

- [1] 王向军 and 王峰. 显微精密成像与微型机械尺寸检测技术. 光学精密工程, 9(6):511–513, 2001.
- [2] 王云新, 王大勇, 赵洁, 李艳, and 万玉红. 基于数字全息显微成像的微光学元件三维面形检测. 光学学报, 31(4):109–114, 2011.
- [3] 王云新, 王大勇, 杨怡姝, 赵洁, 欧阳丽婷, 肖向茜, and 戎路. 数字全息技术在生物医学成像和分析中的应用. 中国激光, 41(2):209002–1, 2014.
- [4] Robert C Kain. Micro-imaging system, 1998.
- [5] 姚骏恩. 电子显微镜的现状与展望. 电子显微学报, 17(6):767–776, 1998.
- [6] 熊平. Ccd 与 cmos 图像传感器特点比较. 半导体光电, 25(1):1–4, 2004.
- [7] 顺祥, 海兴, 物理学, 劲松, and 光学. 物理光学与应用光学. 西安电子科技大学出版社, 2000.
- [8] 王华英, 王大勇, 赵洁, and 谢建军. 用于高分辨率成像的简单高效的数字全息技术. *Chinese Optics Letters*, 6(3):165–167, 2008.
- [9] 苏祥林, 陈文艺, and 闫酒洒. 基于树莓派的物联网开放平台. 电子科技, (09):35–37+41, 2015.
- [10] 刘媛, 张伟, and 王知学. 基于 b/s 和 c/s 架构的嵌入式远程监控系统. 仪表技术与传感器, (10):39–41, 2008.
- [11] 闫石. 基于嵌入式B/S架构的智能家居远程监控系统开发. 硕士, 2014.
- [12] Frederico Valente, Carlos Viana-Ferreira, Carlos Costa, and Jose Luis Oliveira. A restful image gateway for multiple medical image repositories. *Information Technology in Biomedicine, IEEE Transactions on*, 16(3):356–364, 2012.
- [13] Raspberry Pi. Raspberry pi. *Raspberry Pi 1 HDMI 13 Secure Digital 34 Universal Serial Bus 56 Python (programming language) 84*, page 1, 2012.
- [14] Sonal Jain, Anant Vaibhav, and Lovely Goyal. Raspberry pi based interactive home automation system through e-mail. In *Optimization, Reliability, and Information Technology (ICROIT), 2014 International Conference on*, pages 277–280. IEEE.

- [15] Lu Yinli, Yu Hongli, and Zhang Pengpeng. The implementation of embedded image acquisition based on v4l2. In *Electronics, Communications and Control (ICECC), 2011 International Conference on*, pages 549–552. IEEE.
- [16] J.R. Jensen. *Introductory digital image processing: A remote sensing perspective*. 1986.
- [17] 段宗涛, 沙爱民, and 张燕妮. 高性能嵌入式图像处理系统研究. *微电子学与计算机*, 25(6):38–41, 2008.
- [18] 刘松涛 and 殷福亮. 基于图割的图像分割方法及其新进展. *自动化学报*, (06):911–922, 2012.
- [19] 许晓丽. 基于聚类分析的图像分割算法研究. 博士, 2012.
- [20] 汪鑫 and 彭雨薇. 基于树莓派的网络监控系统的研究与实现. *硅谷*, 7(14):25–26, 2014.
- [21] 李文胜. 基于树莓派的嵌入式linux开发教学探索. *电子技术与软件工程*, (09):219–220, 2014.
- [22] 李龙棋, 方美发, and 唐晓腾. 树莓派平台下的实时监控系统开发. *闽江学院学报*, (05):67–72, 2014.

附录 A 数据

这里是附录的数据部分，作为测试；来看个公式编号对不对，如式(??)所示。

A.1 代码

代码 A.1: C code

```
1  /**
2   * struct v4l2_format - stream data format
3   * @type:    type of the data stream
4   * @pix:     definition of an image format
5   * @pix_mp:  definition of a multiplanar image format
6   * @win:    definition of an overlaid image
7   * @vbi:    raw VBI capture or output parameters
8   * @sliced:  sliced VBI capture or output parameters
9   * @raw_data: placeholder for future extensions and custom formats
10  */
11 struct v4l2_format {
12     enum v4l2_buf_type type;
13     /* V4L2_BUF_TYPE_VIDEO_CAPTURE */
14     union {
15         struct v4l2_pix_format pix;
16         /* V4L2_BUF_TYPE_VIDEO_CAPTURE */
17         struct v4l2_pix_format_mplane pix_mp;
18         /* V4L2_BUF_TYPE_VIDEO_CAPTURE_MPLANE */
19         struct v4l2_window win;
20         /* V4L2_BUF_TYPE_VIDEO_OVERLAY */
21         struct v4l2_vbi_format vbi;
22         /* V4L2_BUF_TYPE_VBI_CAPTURE */
```

```
23     struct v4l2_sliced_vbi_format sliced;
24     /* V4L2_BUF_TYPE_SLICED_VBI_CAPTURE */
25     __u8 raw_data[200];
26     /* user-defined */
27 } fmt;
28 }
```

Java code

```
1 /**
2  * Program: Hello world;<br>
3  * Time: 2015/12/04.
4  * @author Stick Cui
5 */
6 public class JavaTest{
7 /**
8  * The main methord.
9  * @param args The parameter when the methord is called.
10 */
11 public static void main(String[] args){
12     System.out.println("Hello world!");
13 }
14 }
```
