



智慧动物实验室系统
AIOT ANIMAL LABORATORY SYSTEM

中国高校计算机大赛 网络技术挑战赛

“生物智联”

——基于 AIoT 的智慧动物实验室管理系统

所在赛道与赛项：A

二〇二三年八月

摘要

动物实验室是对动物进行饲养、观察和记录的场所，在医学、生物学研究中具有广泛的应用。传统的动物实验室采用人工管理的模式，存在养殖效率低、数据监管难、人力成本高、安全隐患大等问题。针对这些问题，作品提出**“生物智联”——基于 AIoT 的智慧动物实验室管理系统**，以 AIoT(人工智能物联网)为核心，结合边缘计算、云计算、物联网、可视化、人工智能等技术，为智慧实验室提供解决方案。本作品“生物智联”通过 Web 网页和手机小程序两种方式呈现，实现五个功能模块：集中式监控大屏、实验室用户管理、环境监测与可视化、养殖仓视频监控、动物体征智能识别。最终达到动物实验室的数据智能采集、设备智能运维、体征智能预测、安全智能监管、集中式动态地图管理的效果，完成自动化、智能化、共享化和物联化的管理模式转变。整个系统全方位保障动物实验室的行为、管理、环境和数据等全景应用，满足现代实验室高效运营和动物福利保护的多重需求，致力于促进动物实验室数字化技术的革新和智慧化建设。

关键词：动物实验室；AIoT；边缘计算；机器学习；可视化

目录

一、 目标问题与意义价值	1
1.1 研究背景	1
1.2 问题分析	2
1.3 系统功能	3
1.4 意义与应用价值	5
1.5 本章总结	6
二、 设计思路与方案	7
2.1 设计思路	7
2.2 系统概要设计	7
2.3 方案详细设计	9
2.4 本章总结	13
三、 方案实现	14
3.1 开发平台及工具	14
3.2 功能模块实现方案	14
3.2.1 用户管理模块方案实现	14
3.2.3 环境监测与可视化模块实现	23
3.2.4 养殖仓视频监控模块实现	26
3.2.5 动物体征智能分析模块	33
3.3 本章小结	42
四、 应用和测试效果	43
4.1 Web 网页端应用效果	43
4.2 小程序应用效果	48
4.3 测试报告	49
4.3.1 硬件电气特性	49
4.3.2 软件鲁棒性	50
4.3.3 软件运行	50
4.3.4 模组运行速度	51

4.3.5 安全性.....	51
4.3.6 部署方便性和可用性.....	52
4.4 本章小结	52
五、 创新与特色	53
5.1 “联”： 使用边云协同实现实验室互联	53
5.2 “大”： 使用大数据分析处理海量数据	53
5.3 “智”： 使用智能化技术赋能解放人力	53
5.4 “全”： 使用自上而下的全面设计方案	54

一、目标问题与意义价值

1.1 研究背景

在过去的疫情冲击下，人民对生命健康日益重视，市场对药物研发、动物实验和检测相关的需求猛增。医学实验特别是药效及药物安全性评价必须依赖于高品质的实验动物。作为用于饲养、繁殖实验动物的场所，动物实验室的监管在实验动物的品质和实验数据的准确性方面发挥重要作用。

然而，现今的许多动物实验室仍采用传统的人工管理方式，管理模式、理念和制度相对滞后，很容易导致人、财、物的管理混乱无序。资源信息的共享也无法有效进行，数据的统计和上报变得困难。随着现代化实验室建设水平的提高，实验室管理工作的复杂性和艰巨性大大增加，对工作的规范性和高效性提出了更高要求。传统的实验室管理模式已经无法满足现代科研人员的需求，甚至成为科研进展的巨大阻力，暴露出信息滞后和失真、使用效益差、管理效率低等问题。因此，对传统实验室进行智能化改造已经成为大势所趋。

综上所述，构筑融合数字科技与生物安全技术创新的智慧动物实验室，对实验室管理的自动化、智能化、物联化以及共享化具有至关重要的意义。通过在多个维度全面保障实验室的行为、管理、环境和数据等全景安全性，科研人员能够借助简明操作，解决繁琐的人工工序、调控环境安全状态、追溯与控制设备运用等问题。此举有望令实验室显著提升其安全性、便捷性和效率性，为科研探索的进程提供坚实支持，实现实验室的安全性、环保性和资源共享性，以促使行业的发展模式逐步向着精耕细作的阶段性演进，继而实现成本的降低和效率的提升目标。

本作品致力于开发“生物智联”——基于 AIoT 的智慧动物实验室管理系统。通过设备自动化操作，实现自动喂养、消毒杀菌、通风换气，降低人工操作的错误率，减少接触。通过智能数据采集与分析，提高数据处理效率。通过集中式云联大屏，共享各地实验室数据，避免信息孤岛。通过机器学习与深度学习算法，智能分析实验动物特征信息，便于实验对象的挑选。为动物实验室带来数字化技术的革新和开放式创新机制，加快智慧动物实验室的建设。

1.2 问题分析

根据传统生物实验室管理的现状分析，目前存在问题如下：

(1) 人工养殖造成环境不稳定性、效率低、成本高、安全隐患大：

人工无法确保实验室的消毒、通风、喂食和喂水等操作在恒定的时间和条件下进行，从而保持实验环境的稳定性和一致性。如果人工操作存在不准确或不及时的情况，可能会导致实验结果的变异或失真。缺乏自动化设备和系统的支持，增加了管理和培训的时间和金钱成本。实验室人员需要直接接触实验动物，进行喂水、喂食等操作，存在因接触传染病病原体或被动物咬伤造成感染的风险。

(2) 养殖仓环境数据采集实时性差、可靠性低、无法回溯：

传统的人工监管下，对实验数据的采集和记录主要依赖于人工操作和手动输入。这种方式不仅需要大量人力，效率低，还可能存在数据记录不准确、遗漏、篡改和难追踪的问题，从而影响实验数据的可靠性、可重复性和可追溯性。

(3) 人工轮询观察动物生长情况，筛选对象工作量大、准确性低：

传统方式依赖人员轮询密切地观察实验动物生长情况，无法自动进行定时巡检并记录动物的生长情况和体征，需要人为记录动物体重，挑选实验动物对象。当出现异常情况或紧急事件时，及时采取相应措施变得困难，可能导致安全风险的增加。

(4) 多地区实验室分散的数据记录带来的信息孤岛问题：

传统实验室中记录通常是分散的，每个实验室信息无法及时共享和整合，在跨团队或跨区域实验室的合作中存在信息断层和不一致。因为每个实验室会有不同的工作方式和数据收集方式，导致数据记录没有被统一整合。这会给研究人员在分析结果时带来困难。



图 1-1 传统实验室动物房管理方式现状

1.3 系统功能

针对以上问题,本团队设计了基于 AIoT 的智慧动物实验室管理系统——“生物智联”,使用物联网,边缘计算,云计算,机器学习,大数据分析与可视化等技术,旨在实现实现动物实验室管理的自动化、智能化、物联化、共享化,在全方位保障实验室的行为、管理、环境、数据等全景安全的同时,为实验室管理人员提供简便,集中,高效的管理平台。系统功能模块图见图 1-2:

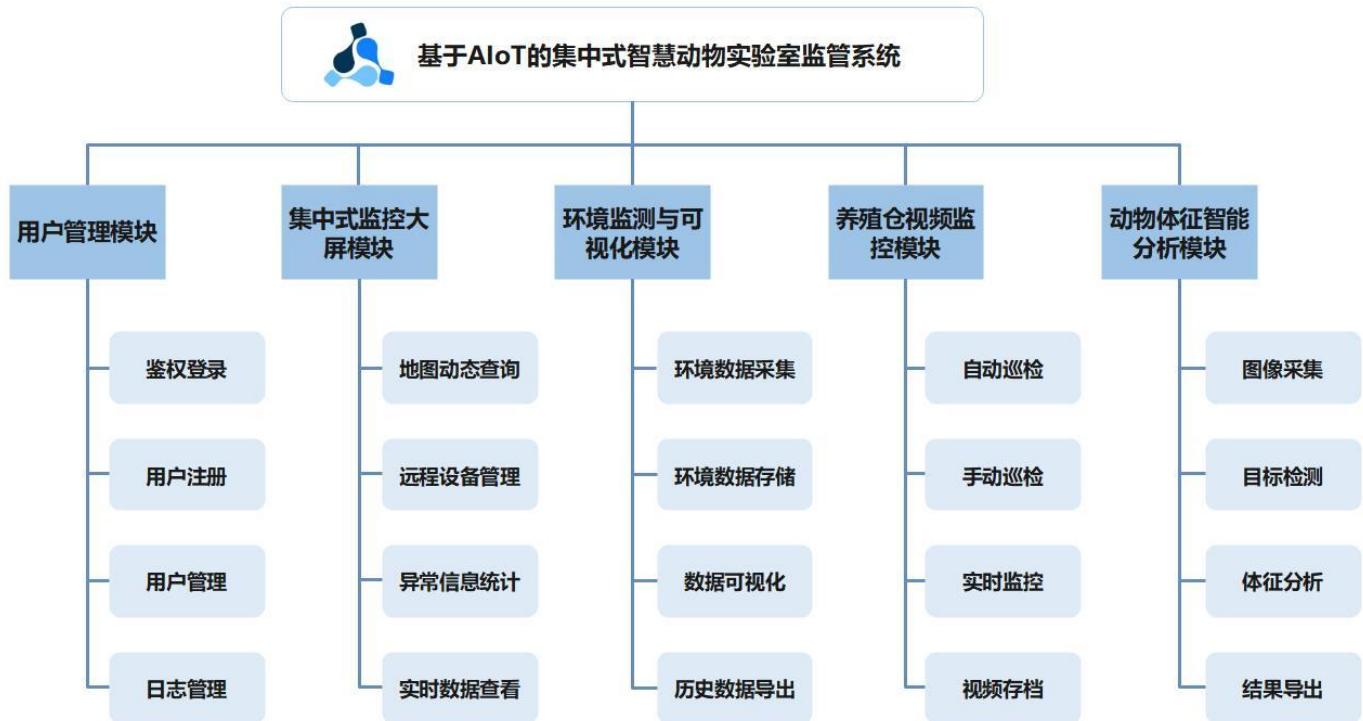


图 1-2: 系统功能模块图

本系统分为五个模块，每个模块功能如下：

● 模块一：用户管理模块：

- 鉴权登录：输入账号、密码进行身份认证。
- 用户注册：给新到实验室人员进行账户权限分配。
- 用户管理：对用户信息等进行增删改查操作。
- 日志管理：记录用户、设备上下线情况、操作记录。

● 模块二：集中式监控大屏模块：

- 地图动态查询：点击地图，选择指定实验室的数据查看。
- 远程设备管理：
 - ✧ 自动喂养：控制实验室喂水，喂食设备开关。
 - ✧ 杀菌消毒：打开紫外线消毒灯，对养殖仓杀菌。
 - ✧ 通风换气：打开风扇通风，为养殖仓换气。

- 异常信息统计：显示当前各地实验室异常报警信息。
 - 实时数据查看：显示当前各地实验室运行、设备在线情况、环境温湿度。
- 模块三：环境采集与可视化模块：
 - 环境数据采集：采集养殖仓温度、湿度、CO₂浓度、光照强度数据并传至云服务器。
 - 环境数据存储：在云端数据清洗并存储加密。
 - 历史数据导出：支持一键导出各时段所有环境数据。
 - 数据可视化：可视化各时段数据。
- 模块四：养殖仓视频监控模块：
 - 自动巡检：轮询所有养殖仓状态。
 - 手动巡检：点击观看指定养殖仓状态。
 - 实时监控：网页远程观看实验室监控。
 - 视频存档：保存每个养殖仓视频。
- 模块五：动物体征智能分析模块：
 - 图像采集：采集养殖仓中动物生长图片。
 - 目标检测：检测图片中动物位置、大小。
 - 体征分析：分析动物体重、颜色、健康指标。
 - 结果导出：导出所有动物分析表。
 - 智能建议：通过将分析结果询问 ChatGPT 以获取有针对性的养殖建议。

1.4 意义与应用价值

本作品提出的“生物智联”智慧动物实验室管理系统，可以实现实验室管理的自动化，智能化，共享化。摆脱传统方式完全依赖人工管理，使实验室

更加安全、便捷和高效，为科研探索进程提供支持。

基于 AIoT 的智慧动物实验管理系统可以为以下多种用户人群提供便利：

- ❖ **科研工作者**：科学研究对实验室的管理需求较高，传统实验室需要专人管理且可能存在数据记录不完整等情况。智能化的实验室通过自动化和远程监控帮助科研工作者更好地管理实验运行过程和结果，提高实验可重复性和数据质量。
- ❖ **相关专业学生**：课程的实践操作需要实验室的资源支持。由于实验室资源紧张、时间安排和操作安全隐患等原因，可以采用智能化无人值守实验室在课下为学生提供更充足的实验机会，且开放时间更灵活，同时还可以避免人员密集的情况，减少病毒感染传播的风险。
- ❖ **制药行业工作者**：研究过程涉及危险品和复杂的实验设备的处理操作，因此对实验室的环境和设备监测控制的精细程度要求较高。部分实验中涉及到的物质可能会对人体造成伤害，采用无人值守实验室可以降低工作人员的风险，并且通过自动控制确保研究过程的稳定性。

1.5 本章总结

本章主要介绍了随着人们对生命健康的重视，市场对药物研发、动物实验和检测相关的需求猛增的背景。分析了目前许多动物实验室仍采用传统的人工管理方式，由此带来的管理混乱无序，以及数据统计和上报的困难等问题。针对这些问题，本作品提出了建设数字科技与生物安全技术创新融合的智慧动物实验室的方案。智慧动物实验室管理系统旨在使用机器学习，大数据分析与可视化，边缘计算，云计算，通过设备自动化操作，智能数据采集与分析，云联大屏数据共享，和机器学习等方式提高实验室管理的效率，降低错误率，避免信息孤岛等问题。

另外，本章还详细解析了“生物智联”智慧动物实验室管理系统的各个功能模块，以及该系统的意义和应用价值，包括为科研工作者，相关专业学生，制药行业工作者等提供更安全、便捷和高效的实验室管理方案。

二、设计思路与方案

2.1 设计思路

本作品设计了基于 AIoT 的智慧动物实验室管理系统，该系统对上述问题的解决方案设计思路如下：

(1) 针对人工养殖造成环境不稳定性、效率低、成本高、安全隐患大的问题：

本作品设计实验室设备自动化控制系统：以 Arduino 为主控单元，借助多种传感器与硬件设备如：电机、水泵、风扇、紫外线消毒灯、蜂鸣器等，实现远程控制实验室设备，完成 Web 端手动或定时定量自动喂养、消毒杀菌、通风换气，远程报警等工作，以减少对专业人员的依赖和与动物的接触。

(2) 针对养殖仓环境数据采集实时性差、可靠性低、无法回溯的问题：

本作品使用三轴双目摄像头远程监控与记录养殖仓，并形成动物体征 AI 模型：以树莓派为主控单元，辅助激光测距传感器、光电门、步进电机、三轴模组、双目摄像头等，可实时远程监控养殖仓内部状态，同时提供自动巡检和手动巡检功能，并储存视频与截图。并形成动物体征 AI 模型自动更新机制，减轻后期维护工作。建立“样本标注——模型自动训练——模型自动分发机制：将巡检中截取图片标注输入到训练好的体征分析模型中，进行实验鼠体征分析。

(3) 针对人工轮询观察动物生长情况，筛选对象工作量大、准确性低的问题：

本作品设计实现环境数据智能采集系统：以 Arduino 为汇聚节点，多种环境采集传感器为采集节点，如：温湿度传感器，CO₂ 浓度传感器，光照强度传感器等搭建实验室无线传感网络，采集数据并上传至阿里云，进行数据可视化与分析导出。

(4) 针对多地区实验室分散的数据记录带来的信息孤岛问题：

本作品建立集中式地图大屏，汇聚各地实验室数据：将各地实验室数据将数据集成到一个统一的平台上，并通过数据大屏进行可视化展示。这样可以打破各个部门之间信息孤岛的壁垒，使得相关信息能够在一个界面上综合展现。

2.2 系统概要设计

本系统从层次角度，自下而上分为四层：感知层，网络层，平台层，应用层。

本系统系统架构模块图见图 2-1:

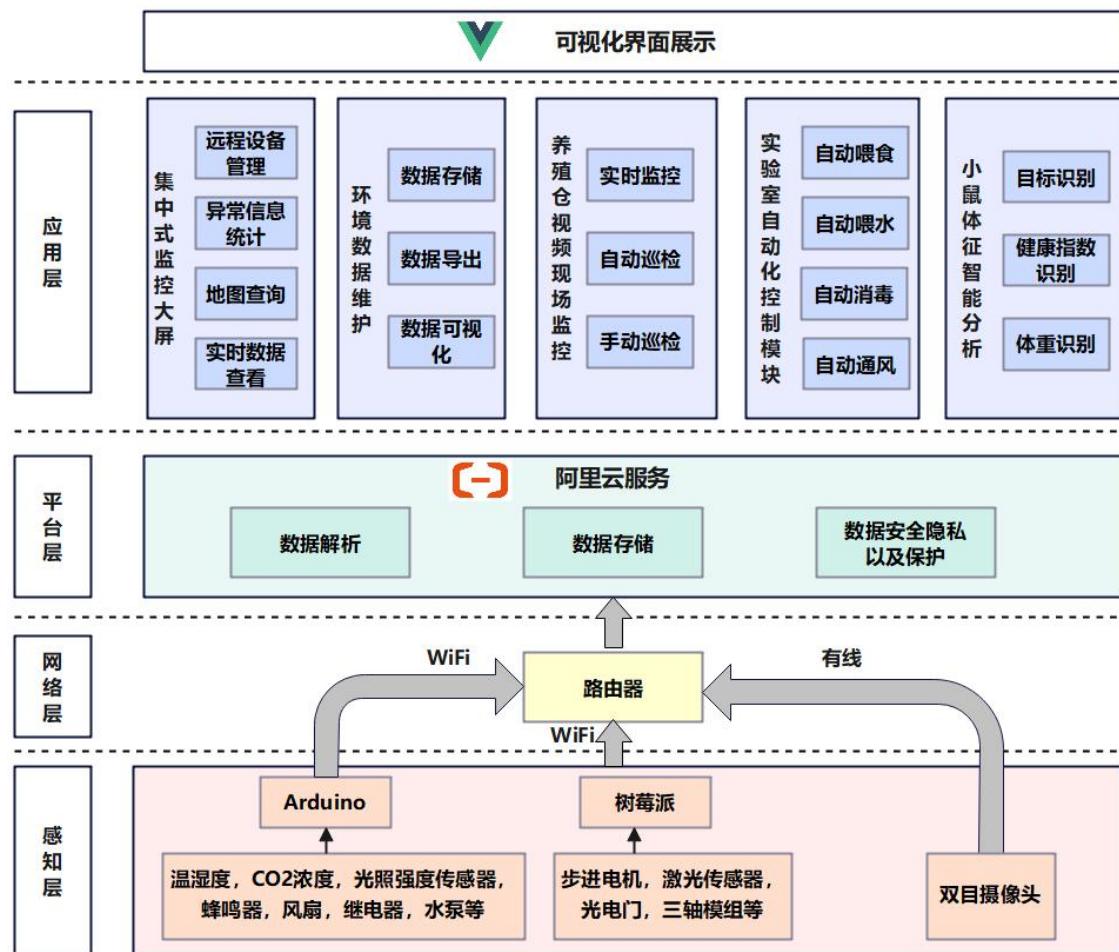


图 2-1: 系统架构模块图

感知层: 使用传感器技术，无线传感网络和嵌入式等物联网技术完成实验室环境信息采集，设备自动化，养殖仓巡检，视频采集。

网络层: 使用无线 WiFi 结合有线方式将数据与视频通过实验室路由器网关上传至阿里云。

平台层: 属于数据存储层，本作品使用阿里云服务器和阿里云数据库服务为网络层，应用层提供数据交互服务，也实现数据的安全隐私及保护。

应用层: 本作品用 Web 网页和手机小程序两种方式呈现。包括五个模块：集中式监控大屏模块、环境数据采集与可视化模块、养殖仓现场监控模块、设备自动化控制模块、动物体征智能分析模块。实现动物实验室的数据智能采集、设备智能运维、体征智能预测、安全智能监管、集中式动态地图管理，实现了全方位的动物实验室监管。

本系统的数据流程图由五大模块组成，数据流程图见图 2-2:

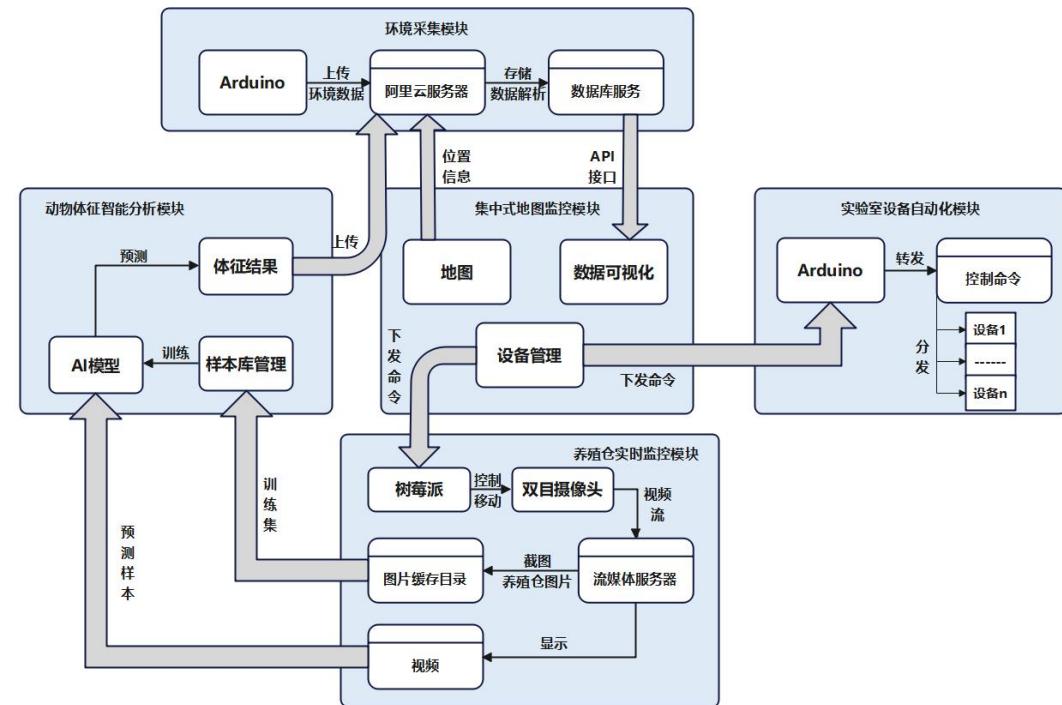


图 2-2: 数据流程图

2.3 方案详细设计

2.3.1 感知层设计

本系统感知层由 Arduino、树莓派、双目摄像头三大部分组成，如图 2-3 所示：树莓派负责驱动三轴模组的运动，通过传送带带动摄像头的移动，到达指定养殖仓位置。Arduino 负责汇聚环境数据，控制自动喂水喂食，风扇，紫外线消毒开关。双目摄像头负责采集养殖仓监控视频。

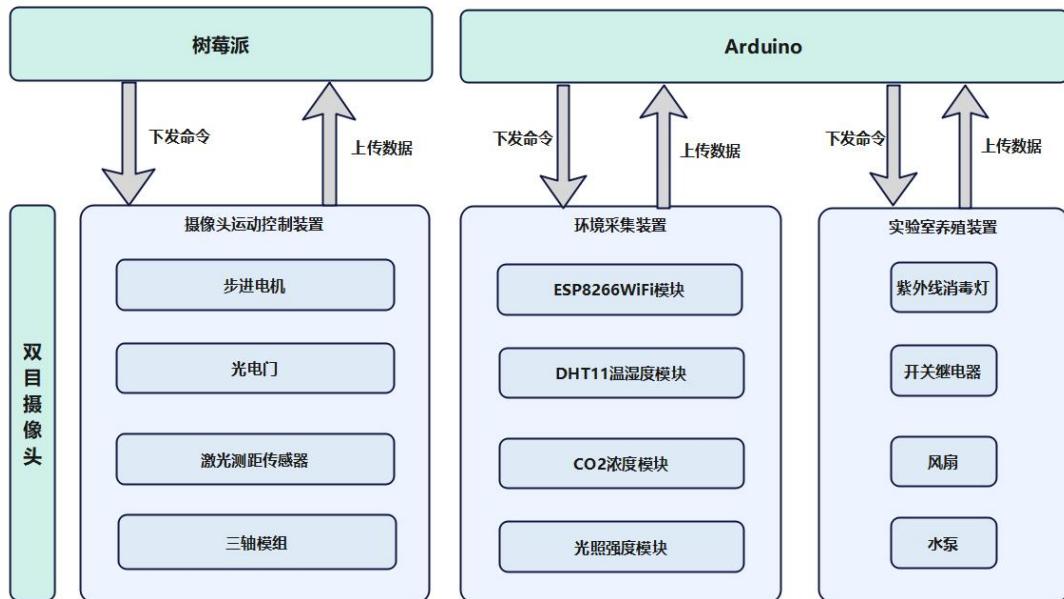


图 2-3：系统感知层流程图

针对上述设计，本作品选取以下硬件，硬件选型表格如表 2-1 所示：

表 2-1：硬件选型表

硬件名称	用途	供电电压
双目摄像头	采集养殖仓视频	POE 供电
DHT11 温湿度传感器	采集养殖仓温湿度	5V
RBY-CO2 CO2 传感器	采集养殖仓 CO2 浓度	5V
光照强度传感器	采集养殖仓光照强度	5V
蜂鸣器	异常报警	5V
紫外线消毒灯	消毒杀菌	12V
风扇	养殖仓通风换气	5V
光电门	模组限位	5V
继电器	控制电源开关	5V
DM860H 步进电机	带动传送带移动	43V

HX711 压力传感器	定量喂食/喂水	3V
电磁阀	控制自动喂水	24V
ESP8266-vl3 WiFi 模块	Arduino 使用 wifi 连接	5V
ATK-MS53L0M 激光测距模块	测距，量程 2 米	5V
树莓派	控制摄像头运动	5V
Arduino Uno	环境采集与上传、命令下发	5V

2.3.2 网络层设计

系统网络层架构图见图 2-4:

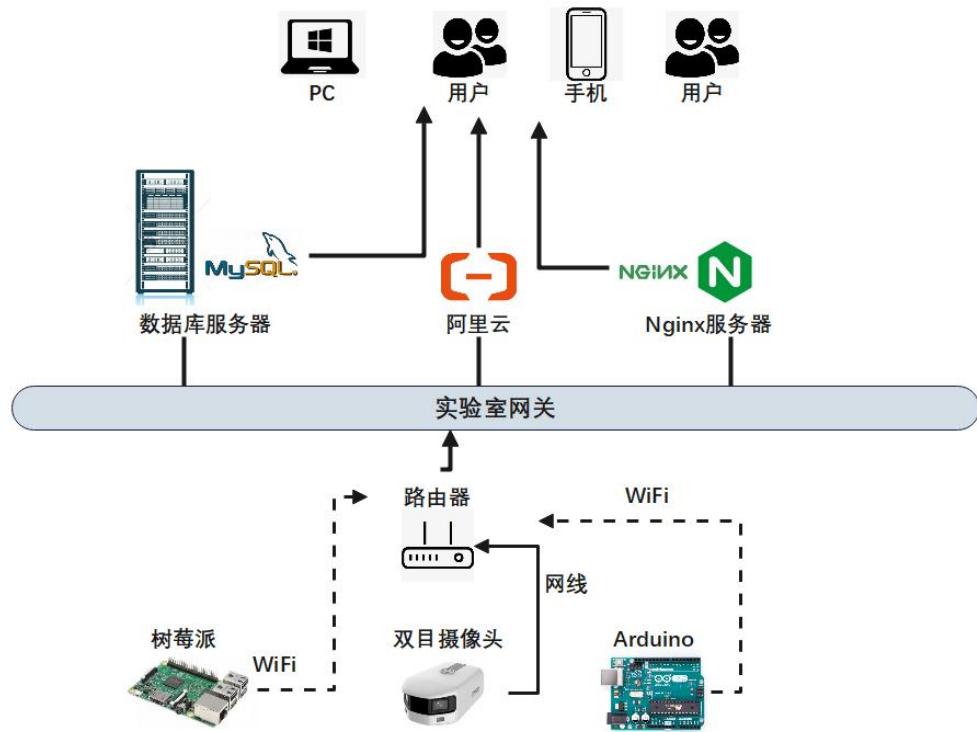


图 2-4: 系统网络层架构图

如图所示，Arduino、树莓派使用 WiFi 技术接入实验室网关，再将设备接入阿里云平台，远程云网规、云配置、云巡检、云运维。数据上传至阿里云数据库服务。应用服务器使用 nginx 进行负载均衡，保障可用性。

2.3.3 平台层设计

本作品平台层以“边云协同”为核心，使用了阿里云提供的边缘计算，云计算等服务，下面是使用阿里云作为中间平台层在项目中体现的优点：

- **降低延迟：**边缘设备处理数据的能力更强，可以在本地对数据进行实时处理和响应，减少了数据传输到云端和返回结果的时间延迟。
- **提高带宽利用率：**通过边云协同，可以将边缘设备上产生的大量原始数据进行预处理和筛选，只将关键数据或摘要信息发送到云端进行进一步分析和存储，减少了数据传输的数据量，提高了带宽利用率。
- **高效利用资源：**云端具有较强的计算能力和存储资源，可以承担复杂的模型训练、深度学习等计算密集型任务，而边缘设备则可以处理实时的简单计算任务，使得整个系统资源得到了更好的利用。
- **提升数据隐私与安全性：**边云协同将一部分敏感数据在本地进行处理和存储，减少了数据在网络传输中的风险，提升了数据隐私和安全性。

2.3.4 应用层设计

应用层由用户端小程序和管理端 Web 网页两部分组成。

用户可登录养殖仓环境监测小程序，观察到不同养殖仓当前实时温度，湿度，CO₂ 浓度，光照强度。控制养殖仓中喂食、喂水、消毒、通风设备开关。

本作品使用微信小程序原生平台进行小程序的开发，通过 wxml 进行页面布局，利用 wxss 对内容外观进行编辑。使用 js 进行前后端的交互。通过 wx.request 向服务器 提交 post 请求，实现与数据库的交互。通过 wx 平台提供的函数、接口实现小程序的基础功能。

实验室管理人员可登录智慧动物实验室系统 Web 网页，进入用户管理模块，可以管理用户信息，查看日志。进入集中式监控大屏，查看各地实验室运行情况，异常报警信息。进入养殖仓实时监控页面，管理员可以控制双目摄像头运动，实现自动巡检、手动巡检，在网页上实时查看每个养殖仓的情况，并将视频保存在云端。进入环境数据可视化页面，查看各时段环境数据，导出环境数据表格。进入动物体征分析页面，导入对应图片，可以得到动物体重、颜色、健康情况等体征数据。

Web 网页应用架构图见图 2-5:

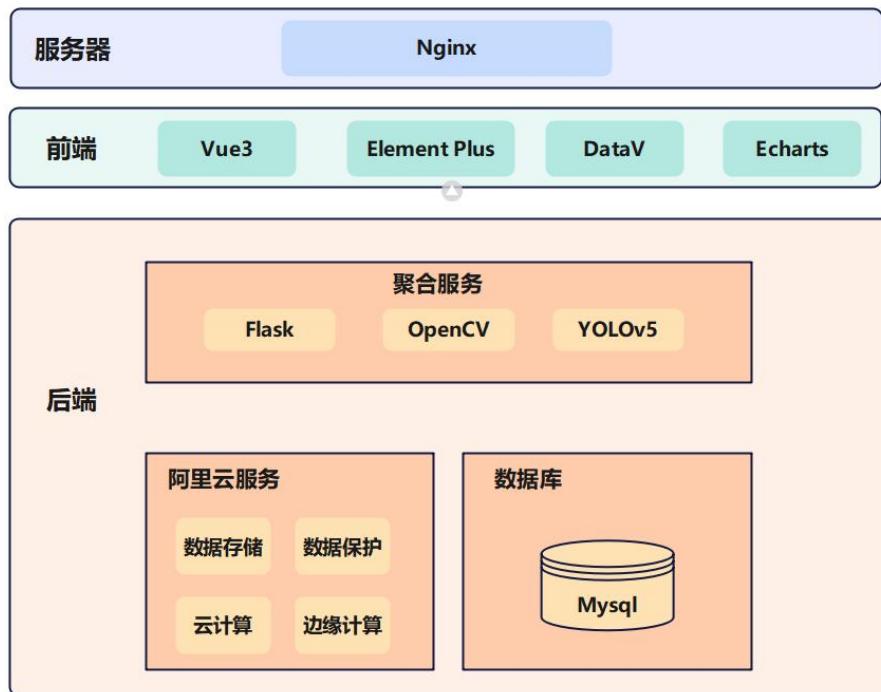


图 2-5: Web 网页应用架构图

Web 网页端前端使用 Vue3 + Element Plus + DataV + Echarts，后端使用 Flask + 阿里云微服务，数据库使用 MySQL，搭建前后端分离架构框架。使用 axios 库通信，设计 RestfulAPI 进行前后端数据交互。

2.4 本章总结

本章主要讨论了基于 AIoT 的智慧动物实验室管理系统的整体设计思路和方案。系统设计采用了层次化的架构，分为感知层、网络层、平台层和应用层。感知层负责信息采集和设备自动化，网络层通过 WiFi 和有线传输的技术将数据和视频流上传至阿里云，平台层使用阿里云服务提供数据交互服务，应用层则通过 Web 网页和手机小程序展现数据和管理设备。

三、方案实现

3.1 开发平台及工具

- 操作系统: Windows ,Linux
- 数据库: MySQL
- 云平台: 阿里云
- 编程语言: Python, C++
- 开发工具: Visual Studio Code, Arduino IDE

3.2 功能模块实现方案

本作品实现的功能可以分为下面五个模块: 用户管理模块, 集中式监控大屏模块, 养殖仓视频监控模块, 环境监测与可视化模块, 动物体征智能分析模块。

3.2.1 用户管理模块方案实现

用户管理模块实现的功能有: 鉴权登录, 用户注册, 用户管理, 日志管理。

系统账号数据表主要是记录系统使用人员的账号信息, 整个系统的所有账号信息保存在一张表内, 数据表结构见表 3-1。

表 3-1: 系统账号数据表

字段	类型	主键
账号	字符串(12)	是
密码	数字和字符(10)	是
账户类型	字符串(1)	是
备注	字符串(60)	否

在所有字段中, “账号”属性保存的是用户账户的用户名。“密码”属性存储账户的密码。“账户类型”属性的字符的值只能为“1”或者“0”, 其中, 为“1”时, 表明该账户为管理员账户, 可以登录系统中的用户管理页面。若为“0”, 说明只是普通账户, 不能访问相关页面。“备注”属性可以为空, 用来简要备注用户信息说明。

(1) 鉴权登录与用户注册

鉴权登录功能的流程图见图 3-1:

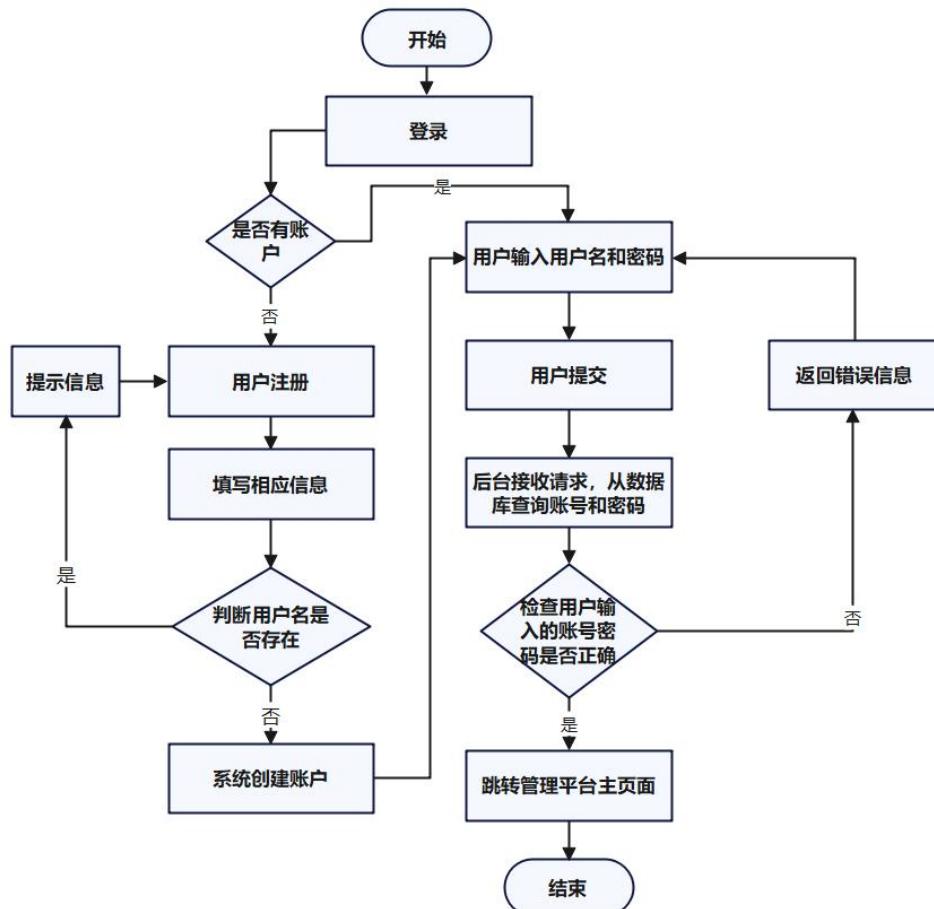


图 3-1：鉴权登录功能流程图

鉴权登录: 在用户提交登录申请之后, 后台会从数据库的系统账号数据表中获取所有的登录账号和密码信息, 并封装成 List, 然后将从前端获取的用户输入的账号、密码信息与 List 进行对比, 看是否存在符合的账户。如果正确, 则将登录账户的用户信息保存到 Session 中, 如果不存在, 则提示账号密码输入错误并重新返回登录界面。

用户注册: 用户若无账号, 需点击注册选项进入注册页面, 按照注册流程填写用户名和密码来创建账号, 之后选择用户创建。后台会从数据库的系统账号数据表中获取所有的登录账号, 并封装成 List, 然后将从前端获取的用户输入的账号与 List 进行对比, 搜索是否存在重复的账户。若存在则提示“用户名已存在”, 返回用户注册步骤。若不存在则系统创建账户, 将账户和密码保存到数据库中。

用户可使用创建好的账户进行登陆操作。

(2) 用户管理

用户管理功能的流程图见 3-2:

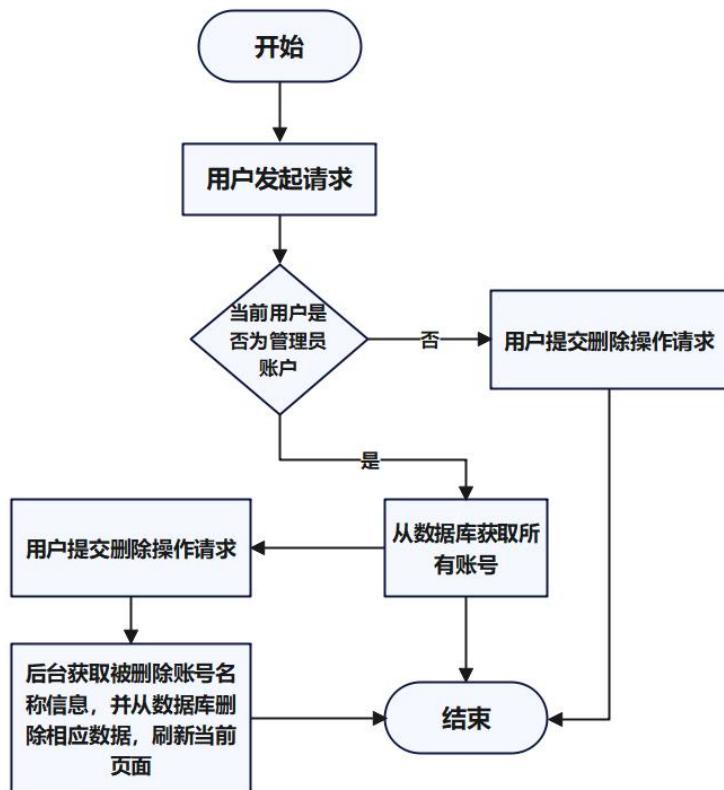


图 3-2: 用户管理功能流程图

用户申请访问用户管理界面之后，后台会从数据库中查询所有的管理员账户的账号名称信息，然后从 Session 中获取当前登录的用户名信息，将已登录的用户名与管理员账户名称信息进行对比，如果管理员账户信息的队列中存在当前登录的用户名，说明当前登录的用户为管理员用户，允许访问用户管理界面。如果不存在，则弹窗提示当前用户不具有权限，禁止访问并跳转主页面。管理员进入用户管理界面之后能查看并管理数据库中存储的所有账户信息。当管理员在某个用户信息之后提交了删除请求，后台会处理相关请求，获取被删除用户信息并删除数据库中对应的所有信息，完成用户删除操作。

(3) 日志管理

日志管理功能的流程图见 3-3:

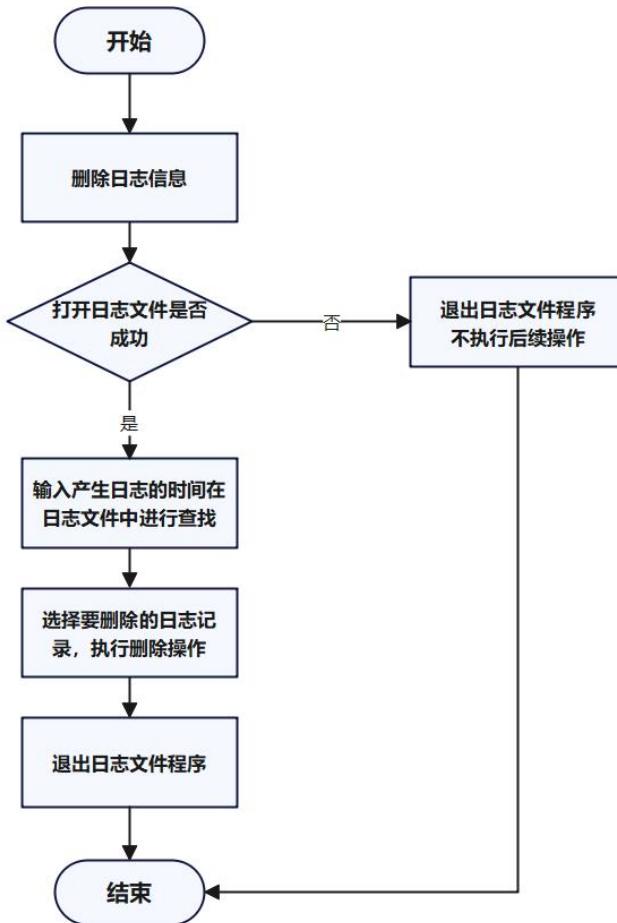


图 3-3：日志管理操作流程图

管理员完成登陆操作，进入管理员操作界面。管理员申请删除日志信息服务后，系统打开日志文件程序。管理员输入时间查找所需删除的日志记录，后台从 Session 获取日志时间，后台查找对应时间的日志记录并返回。管理员选择并执行删除操作，后台会处理相关请求，获取被删除日志信息并删除数据库中对应的所有信息，完成日志删除操作。

3.2.2 集中式监控大屏模块方案实现

实验室集中式数据监控大屏以“边云协同”为核心，使用云计算、边缘计算、大数据分析、智能处理等技术，将原本各地分散的、独立的边缘实验室全景数据能够在云端海量汇聚，实现在中心大屏进行数据展示、数据共享、集中管理。实验室中心数据具有高扩展性、高适应性，能够避免“信息孤岛”，解决在跨团队或跨区域实验室的合作中存在信息断层和不一致问题。

下面是实验室中心大屏实现的功能，功能模块图见图 3-4：

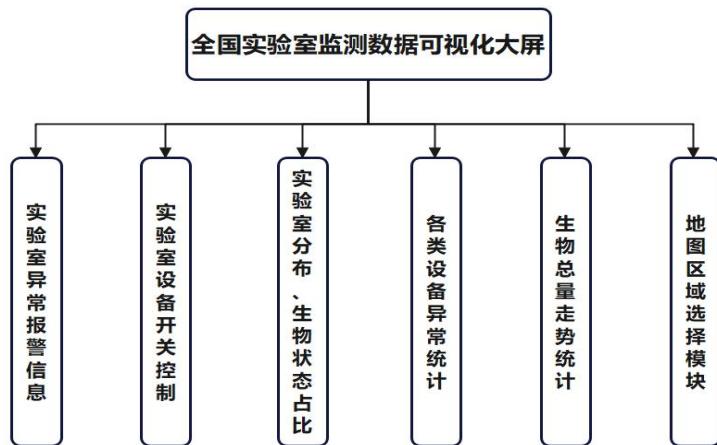


图 3-4: 功能模块图

实现的功能如下：

- 各省份分布的实验室情况散点标记映射到地图
- 实时统计全国各省份的实验室异常报警信息
- 控制实验室所有设备开关与异常数量统计
- 查看全国实验室数量分布与生物健康状态分布
- 分析各月份实验室生物总量走势，全面反应实验室养殖状况

针对上述实现功能，本作品设计实验室中心大屏布局，尽可能在紧凑的屏幕空间内展示高密度的信息。设计各种图表类型，增加各视图的可理解性。见图 3-5：

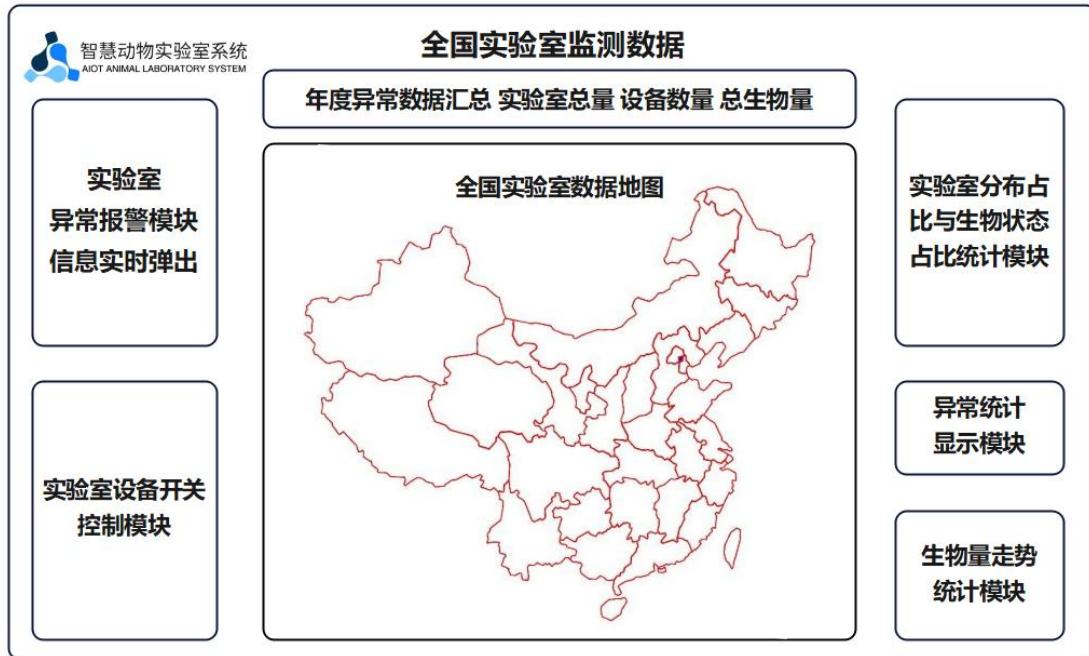


图 3-5：实验室集中式数据监控大屏布局

本作品按照设计稿开发了一个实验室集中式数据监控大屏原型系统，最终实现效果见图 3-6：

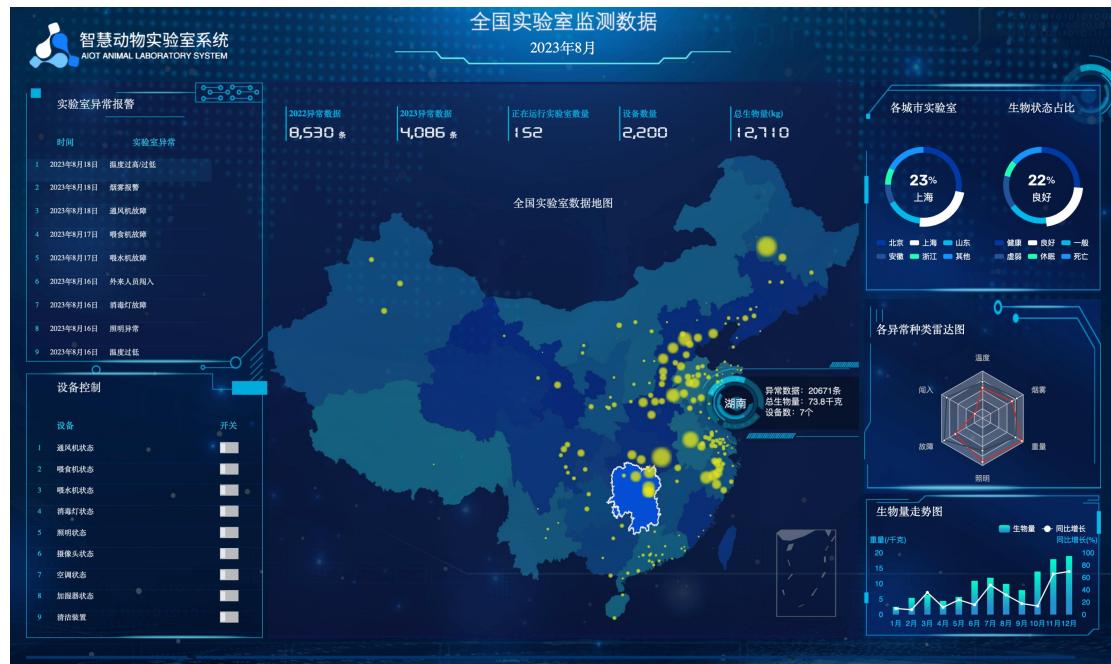


图 3-6：集中式数据监控大屏

实验室集中式数据监控大屏原型系统可以分成下面几个视图：

- 实验室地图分布视图：

该视图效果见图 3-7：



图 3-7：实验室地图分布视图

该视图每隔两秒动态高亮显示各省份实验室总体情况，使用散点标记，将各个实验室地理位置映射到地图上，反应实验室数量分布情况。

➤ 实验室数据总览视图：

该视图效果见图 3-8：

2022异常数据	2023异常数据	正在运行实验室数量	设备数量	总生物量(kg)
8,530 条	4,086 条	152	2,200	12,710

图 3-8：实验室数据总览视图

该视图展现实验室关键数据总览，包括 2022 年异常数据条数，2023 年异常数据条数，正在运行实验室数量，设备数量，总生物量。

➤ 实验室异常报警视图，见图 3-9：



图 3-9：实验室异常弹出视图

该视图使用动态轮播表，实时更新报警日志，方便管理员快速查看，

- 实验室设备控制视图，见图 3-10：



图 3-10：实验室设备开关视图

该视图使用控制栏开关，能够远程全局控制全国各地实验室的喂食，喂水，消毒，通风设备开关。

- 城市、生物状态比例视图，见图 3-11：



图 3-11：各城市实验室与生物状态数据占比

该视图使用饼图反应各城市实验室数量占比，生物健康情况（健康，良好，一般，虚弱，休眠，死亡）占比。

➤ 各异常种类雷达视图，见图 3-12：

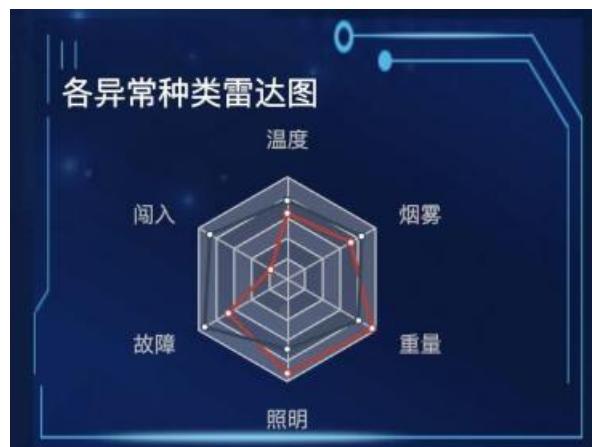


图 3-12：各异常种类占比视图

该视图使用雷达图反应实验室各异常（温度，烟雾，重量，闯入，故障，照明）数量占比。

➤ 实验室生物量走势图，见图 3-13：

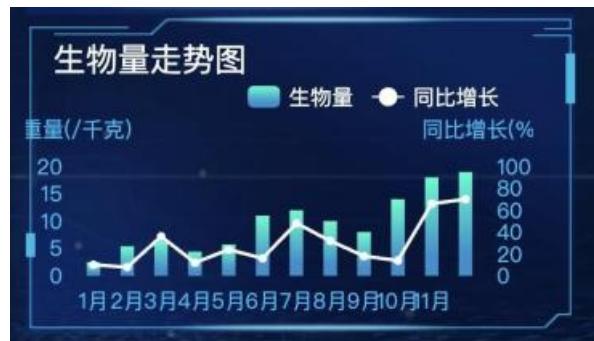


图 3-13: 实验室生物量走势图

该视图使用折线图和条形图复合图标，条形图表示每个月份生物量，折线图表示月份生物量同比增长趋势。

3.2.3 环境监测与可视化模块实现

本作品在环境监测与可视化模块实现实时采集各个实验仓环境数：温度、湿度、CO₂浓度、光照强度数据并上传至阿里云服务器进行数据处理与分析，在Web网页前端进行数据可视化。

下图为环境监测与可视化模块实现的流程图见图 3-14:

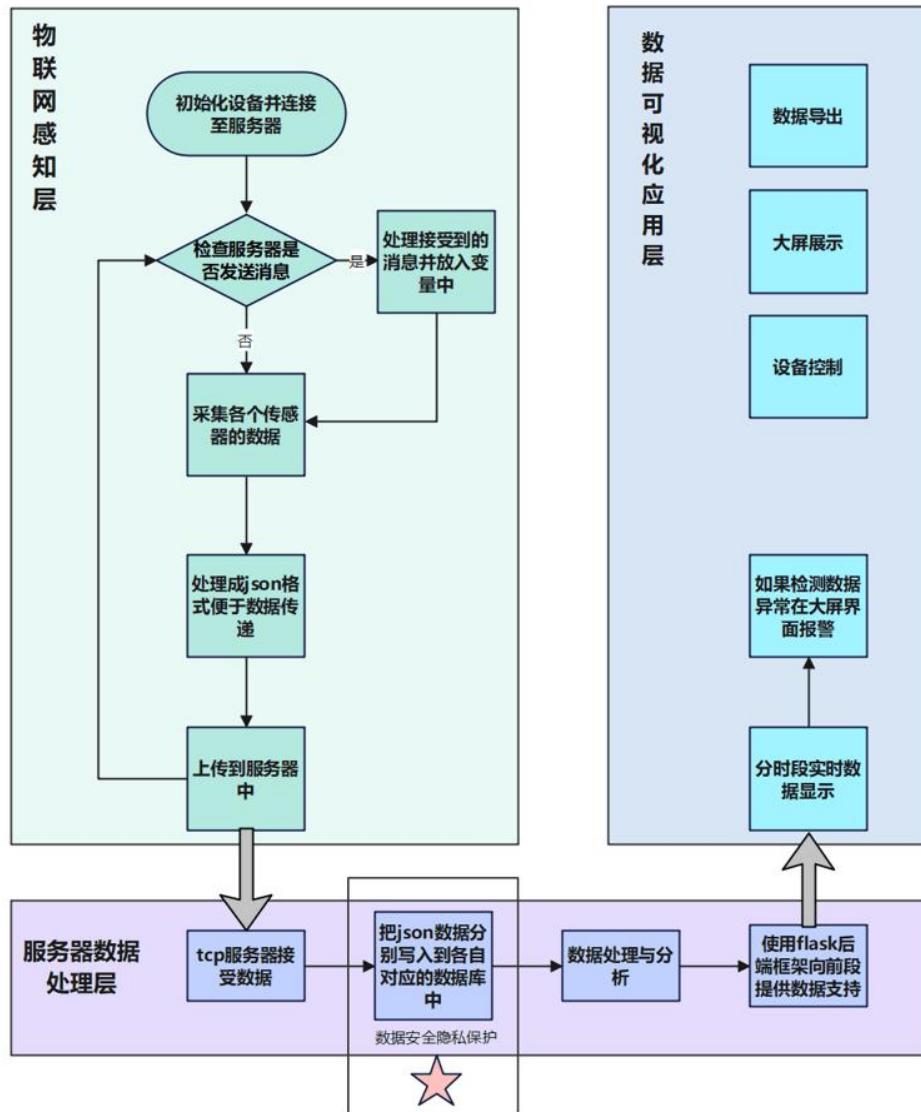


图 3-14:环境监测与可视化模块流程图

该模块通过三层——物联网感知层，服务器数据处理层以及数据可视化应用层组成。底层通过数据的采集以及一些信息的处理发送到阿里云服务器上，转存到阿里云数据库中，对数据进行了隐私化以及安全性保护。并对数据进行一系列处理，最后通过 flask 后端向前段可视化应用层传递数据。在应用层中，本作品实现了数据的分时分段展示以及数据异常的报警。最后通过应用层可以反向控制底层的紫外线，风扇等设备。以及数据的表格导出功能。

在前端，数据的可视化是通过 Echarts 呈现出来的。通过向运行在阿里云的后端请求数据。后端再从数据库中取出相应时间段的数据并以 json 格式返回给前端。前端接收后经过图形化处理，最终在 Web 上呈现出数据，如图 3-16 所示。

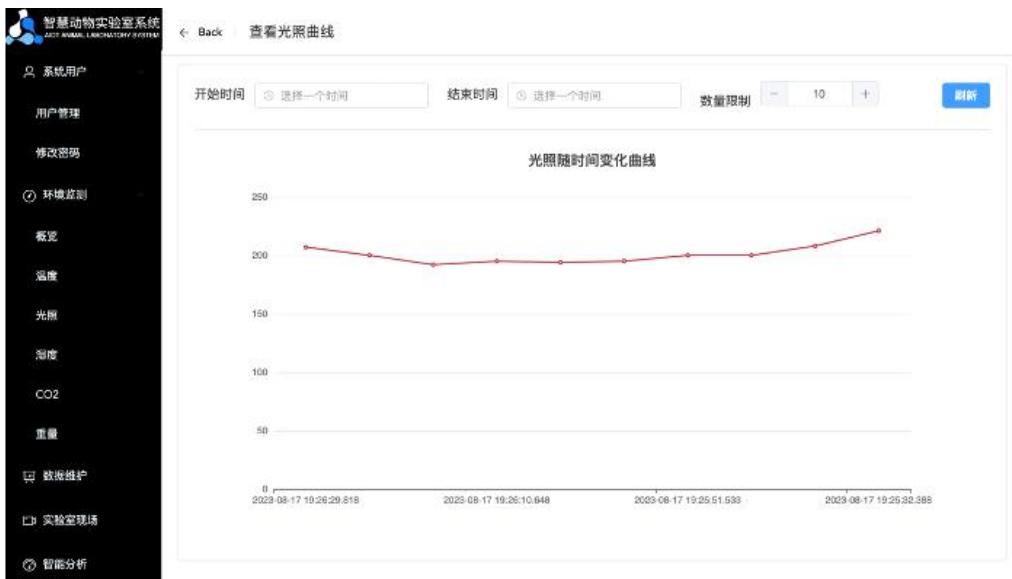


图 3-16: 环境数据展示图

而数据的导出功能主要是通过用户给定的数据时间范围和所需要获得的环境参数类别。从数据库中获取相应环境参数不同时间的数据。然后后端把这个数据变成表格的形式发送到前端以供用户下载。

接下来是环境数据库设计，如表 3-2 所示：

表 3-2：环境数据库设计

属性中文名	属性	数据类型	含义说明
光照	Light	Varchar(255)	光照强度
二氧化碳	CO2	Varchar(255)	CO2 浓度
湿度	Humidity	Varchar(255)	湿度数值
温度	Temp	Varchar(255)	温度数值
重量	Weight	Varchar(255)	喂食重量
时间戳	Stamp	Varchar(255)	数据上报时间
数据标识	Id	Int	主键用于标识

在所有字段中，光照属性是通过光照传感器采集上来的小鼠仓内的光照强度。二氧化碳属性是指小鼠仓内的二氧化碳浓度。温度和湿度分别是为了检测小鼠的

生活环境是否适宜，他们都是可变字符串。重量属性是目前食物的重量。而时间戳属性是反应此数据上报的时间。最后一个数据标识是为了标识唯一的属性，作为该数据模式的主键。

Arduino 连接的硬件有 ESP8211 WiFi 模块，DHT11 温湿度模块，光敏传感器，CO2 浓度传感器，风扇，紫外线灯，电磁阀，电子秤，电机，继电器等。实现环境数据采集，自动喂养，通风换气，消毒杀菌。Arduino 的硬件连线见图 3-15：

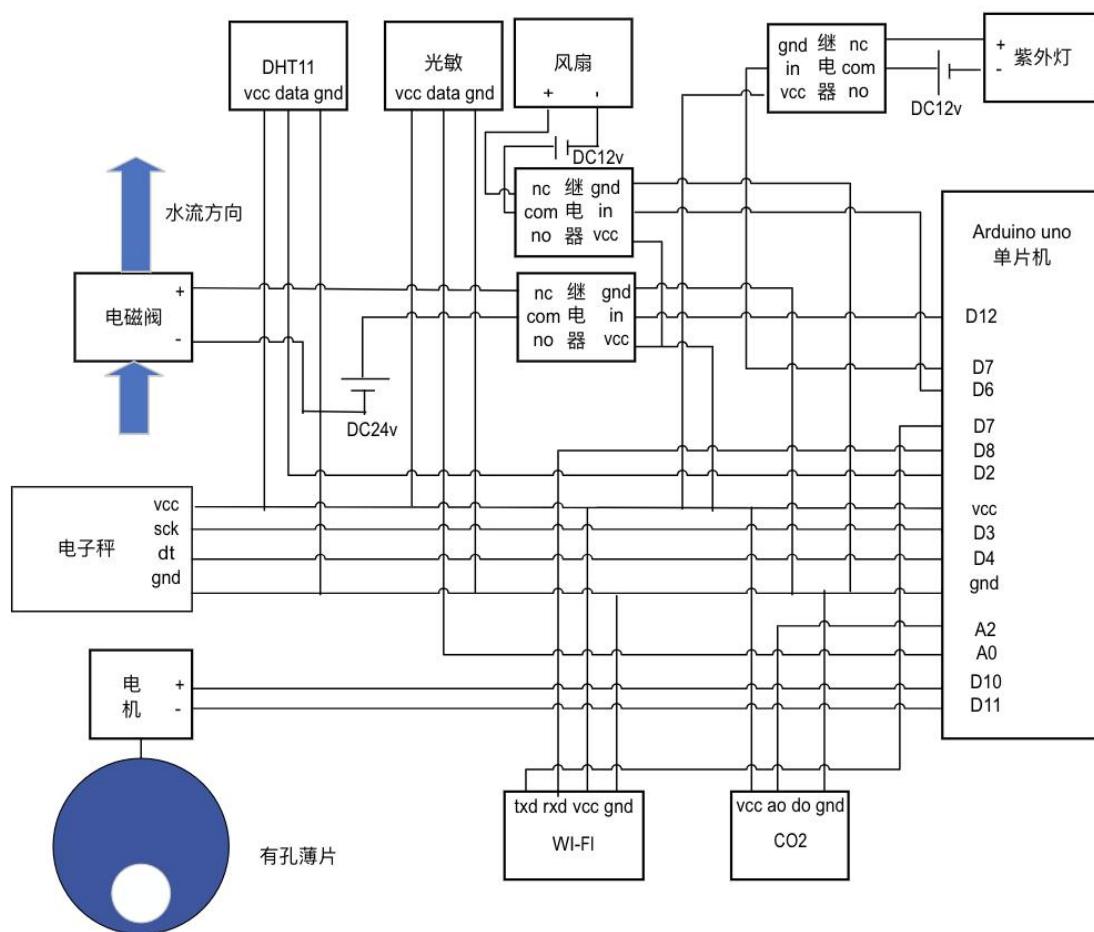


图 3-15:Arduino 硬件接线图

3.2.4 养殖仓视频监控模块实现

该模块实现的功能有自动巡检，手动巡检，实时监控，视频存档。养殖仓视频监控模块流程图见图 3-17：

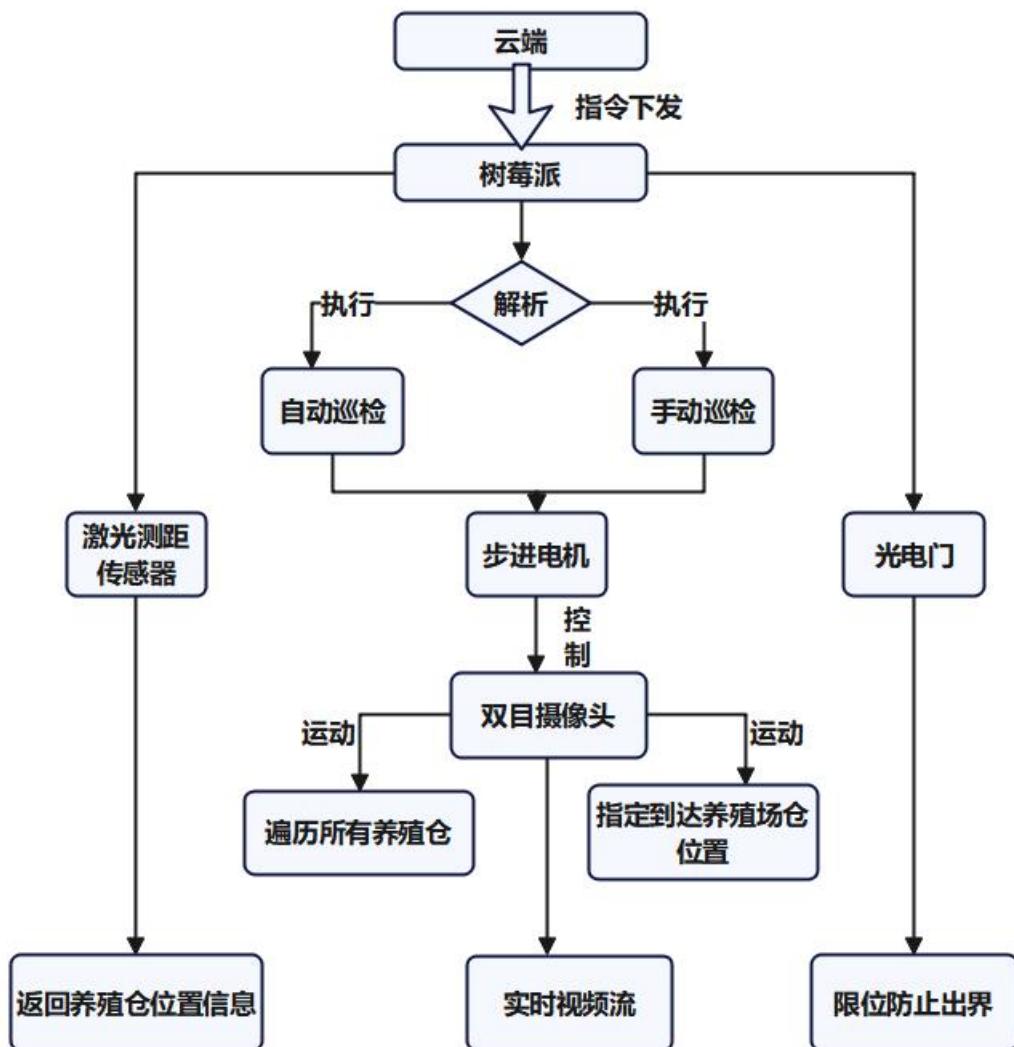


图 3-17：养殖仓视频监控模块流程图

该模块由树莓派为主控单元，辅助激光测距传感器获取位置信息，光电门限制摄像头运动避免出界。流程具体表现为：树莓派接收云端下发的指令，解析后执行不同功能，通过控制步进电机转动带动传送带移动，控制双目摄像头运动。若命令为自动巡检：摄像头自动遍历所有养殖仓，停留在某一养殖仓一段时间后，将采集视频上传至云端。若功能为手动巡检：树莓派获取到云端发来的位置指令后，驱动步进电机转动，控制传送带带动摄像头到达指定养殖仓位置。

树莓派硬件接线图见图 3-18：

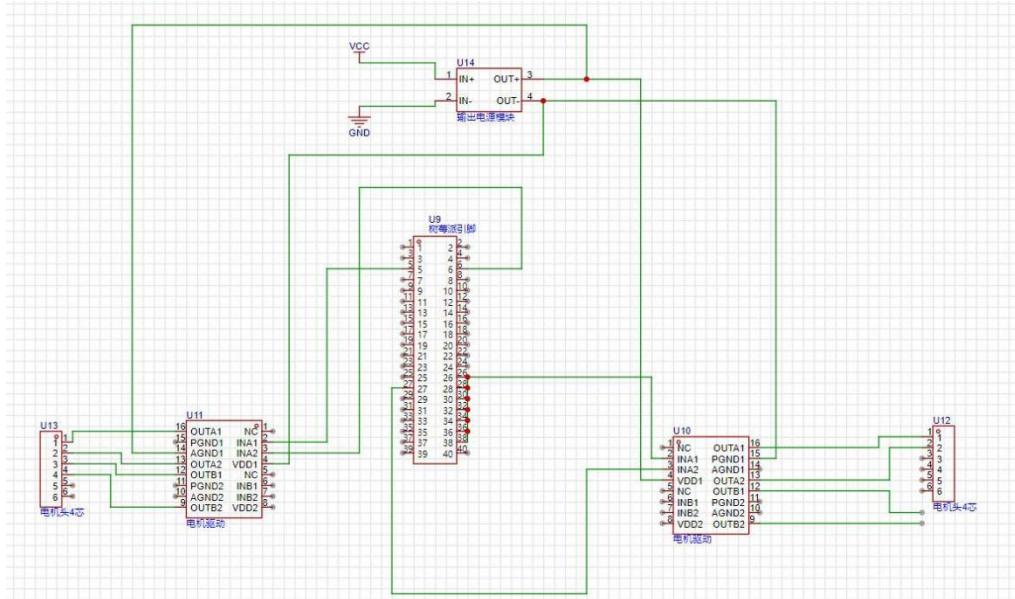


图 3-18: 树莓派硬件连线表

下面是养殖场视频监控模块实现过程中需要解决的四个重要问题:

1. 树莓派怎样稳定地控制步进电机转动?
2. 树莓派怎样高效地控制摄像头运动?
3. 树莓派怎样和上层应用双向通信?
4. 树莓派怎样规划巡检路径?

➤ 问题一的解决办法可以总结为: 步进电机脉冲产生方式选择

树莓派有两种方法可以产生脉冲信号, 一种是通过 PWM (脉冲宽度调制) 技术产生脉冲, 另一种是通过高低电平交替的方式产生脉冲。

树莓派通过 PWM 技术产生的是一个周期信号, 并不能直接控制输出固定数量的脉冲。并且树莓派上的 PWM 输出通常受限于最大输出频率和分辨率等因素, 因此可能会出现精度、稳定性和输出波形质量等方面的问题。相比于 PWM 技术, 使用高低电平交替产生脉冲信号的方式可以更灵活地控制脉冲数量、时间长度和频率等参数, 可以根据实际需求进行自由组合和调整。所以在本作品中本作品使用高低电平交替的方式产生脉冲数可控的脉冲信号。

➤ 问题二的解决办法可以总结为: 多线程程序设计

由于在程序中要并行执行 socket 双向通信与上位机保持实时通信、执行完成双目摄像头运动控制功能并且在双目摄像头执行运动至特定位置的命令时, X 轴和 Y 轴的步进电机需要同时工作, 所以设计了多线程实现并发执行多个任务。

也可以通过多进程编程实现并行执行多任务。但是相比于多线程，多进程编程中每个进程都有独立的地址空间，进程之间的切换会有较大的开销。而多个线程可以在同一进程中并发或并行地执行，共享同一个地址空间和其他系统资源，因此线程更轻量级，也更容易管理和切换，线程之间切换的开销小。

树莓派端程序设计使用 C++语言，使用 Linux 的 Posix 协议实现多线程。程序运行后，主线程中先开辟 ServerSocket 线程等待上位机（PC 端）连接树莓派，当连接建立，此时并不能实现实时的双向通信，只能实现同步通信，需要等待发送（或接收）结束才能继续接收（或发送）信息，于是在于 PC 端建立 TCP 连接后，在 ServerSocket 线程中开辟另一个线程 RecvThread 专门用来实时接收上位机的命令。主线程完成双目摄像头的运动控制功能。当前程序共 3 个线程，当树莓派端接收到上位机发送的“从当前位置运动到 x 号养殖舱位置时”，为达到快速响应，此时运动轨迹为斜线，故此时在主线程中再次开辟 2 个线程 myThread_x 和 myThread_y 分别完成 X 轴和 Y 轴步进电机的同步控制，此时程序共 5 个线程，当此命令执行结束，主线程自动销毁这两个临时线程。

线程同步设计：

线程同步是为了解决多线程并发访问共享资源时可能出现的竞态条件和数据一致性问题。线程同步在本程序中也用于协调多个线程之间的执行顺序，来实现线程间的协作和同步。线程同步是为了保证多线程并发访问共享资源的正确性和一致性，确保程序的执行结果符合预期，并避免竞态条件和数据不一致性问题。

为了解决这些问题，需要进行线程同步操作，确保在任意时刻只有一个线程可以访问共享资源，从而避免竞态条件和数据一致性问题。常用的线程同步机制包括互斥锁、信号量、条件变量等。在此程序中本作品同时使用了互斥锁和条件变量来实现线程同步。

➤ 问题三的解决办法可以总结为：socket 网络通信

上位机与树莓派通信使用 IPv4（AF_INET）协议，传输使用 TCP 协议（SOCK_STREAM）。上位机作为客户端，下位机作为服务端。在树莓派端执行程序时，在 ServerSocket 线程中：使用 socket() 函数创建一个套接字对象 listen_fd，并调用 bind() 函数绑定 IP 地址和端口号，再用 listen() 函数将套接字转换成监听状态。在 PC 端创建一个客户端：使用 socket() 函数创建一个套接字对象，并调

用 `connect()` 函数连接到服务器提供的 IP 地址和端口号。在服务器进程中，使用 `accept()` 函数等待并接收客户端的请求，一旦有客户端连接到服务器，`accept()` 函数就会返回一个新的套接字文件描述符，用于后续与该客户端进行通信。树莓派端接收到 PC 端的连接套接字后，关闭 `listen_fd` 套接字，然后开辟 `RecvThread` 线程用于发送消息。

➤ 问题四的解决办法总结：路径规划

养殖架为 7×5 结构，X 方向 7 个养殖仓，Y 方向共 5 个养殖仓，共 35 个养殖仓。基于养殖架建立 XOY 平面坐标，第一层第一个坐标为 $(0,0)$ 。巡检方式分为自动巡检和手动巡检，手动巡检包括上、下、左、右移动和指定位置运动。摄像头自动巡检移动路径为“S”字形，指定位置运动路径为斜线。对养殖架建立坐标系如下图 3-19。使用相对路径导航算法完成双目摄像头的运动控制。

自动巡检路径为“S”字形，手动巡检包括上、下、左、右、暂停以及运动到指定养殖仓（坐标）功能。下位机接收到“上”、“下”、“左”或“右”单步命令时，双目摄像头移动一个养殖仓的距离。若要执行移动到某个具体的养殖仓，下位机在收到命令后，将养殖仓编号分解成坐标位置，然后分解成 X 轴运动方向的距离以及 Y 轴运动方向和距离，通过多线程的方法，使 X 轴和 Y 轴的步进电机同时转动，从而时摄像头通过斜线路径移动到确定位置。

步进电机控制双目摄像头自动巡检、手动控制运动方案设计：

程序运行开始，先等待上位机连接，上位机连接之后给下位机发送命令，包括自动巡视指令、上下左右指令、暂停指令、到具体编号养殖仓指令以及结束指令。程序执行时：

- (1) 先将初始化位置的全局变量(`lastxpos, lastypos`) 和(`xpos, ypos`)置零；
- (2) 等待上位机发送命令；
- (3) 读取上位机发送的指令，每次发送的命令包括命令时间和命令，对命令解析如果指令为“auto”，则执行自动巡视步骤
- (4) 自动巡检。如果是“move up”，执行步骤
- (5) 向上移动一个单位。如果是“move down”，执行步骤
- (6) 向下移动一个单位。如果是“move left”，执行步骤
- (7) 向左移动一个单位。如果是“move right”，执行步骤
- (8) 向右移动一个单位。如果是“book x”，执行步骤
- (9) 运动到指定位置。如果收到“continue”命令，则执行步骤
- (10) 从自动巡检中断的地

方继续巡检，如果是“finish”，则执行（11）退出程序；

（4）自动巡检时，先将计数器 index 置零，每移动一个单位距离至下一个编号，“index++”且更新(lastxpos,lastypos)，然后立即向上位机发送当前坐标位置和养殖仓编号，再判断是否一次巡检结束，若一次巡检已经完成，则跳到步骤（2）执行，否则，读取上位机命令，若无新指令，继续从当前记录的“index”号养殖仓向下一个养殖仓运动并巡检，若有新命令，跳到（3）执行；

（5）向上移动一个单位距离，更新(lastxpos,lastypos)，跳到步骤（2）；

（6）向下移动一个单位距离，更新(lastxpos,lastypos)，跳到步骤（2）；

（7）向左移动一个单位距离，更新(lastxpos,lastypos)，跳到步骤（2）；

（8）向右移动一个单位距离，更新(lastxpos,lastypos)，跳到步骤（2）；

（9）分解成 X 和 Y 方向运动，开辟 X 和 Y 方向的运动线程，从当前养殖仓移动到“x”号养殖仓（x 为养殖仓编号），更新(lastxpos,lastypos)，跳到步骤（2）；

（10）先从当前位置(lastxpos,lastypos)运动到暂停时的养殖仓的下一个编号位置“index”，然后继续巡检，每巡检一个养殖仓和步骤（4）执行相同的判断操作；

（11）收到结束命令，结束程序运行。

基于养殖架建立 XOY 坐标系，坐标系如下图 3-19 所示：

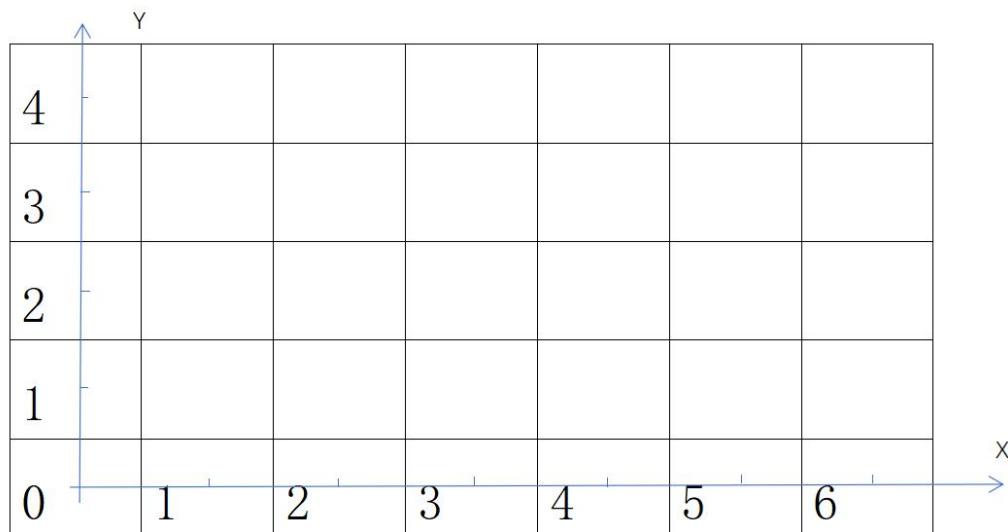


图 3-19：基于养殖架建立 XOY 坐标系

如果接收到自动巡视指令，则按照如下路线进行巡检，见图 3-20：

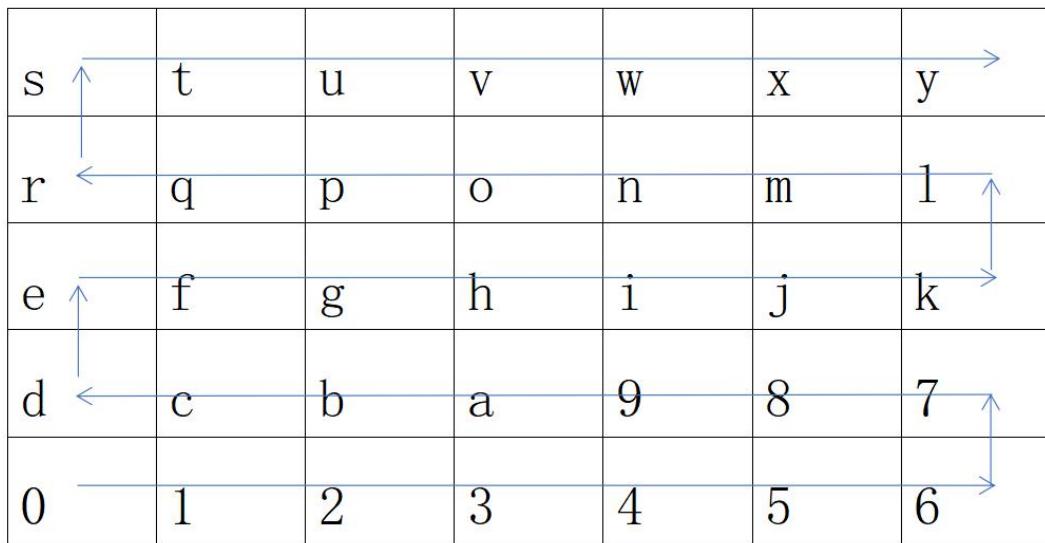


图 3-20: 巡检路线图

解决上述四个问题后，总结本作品实现养殖仓监控摄像头运动控制的程序流程图见图 3-21：

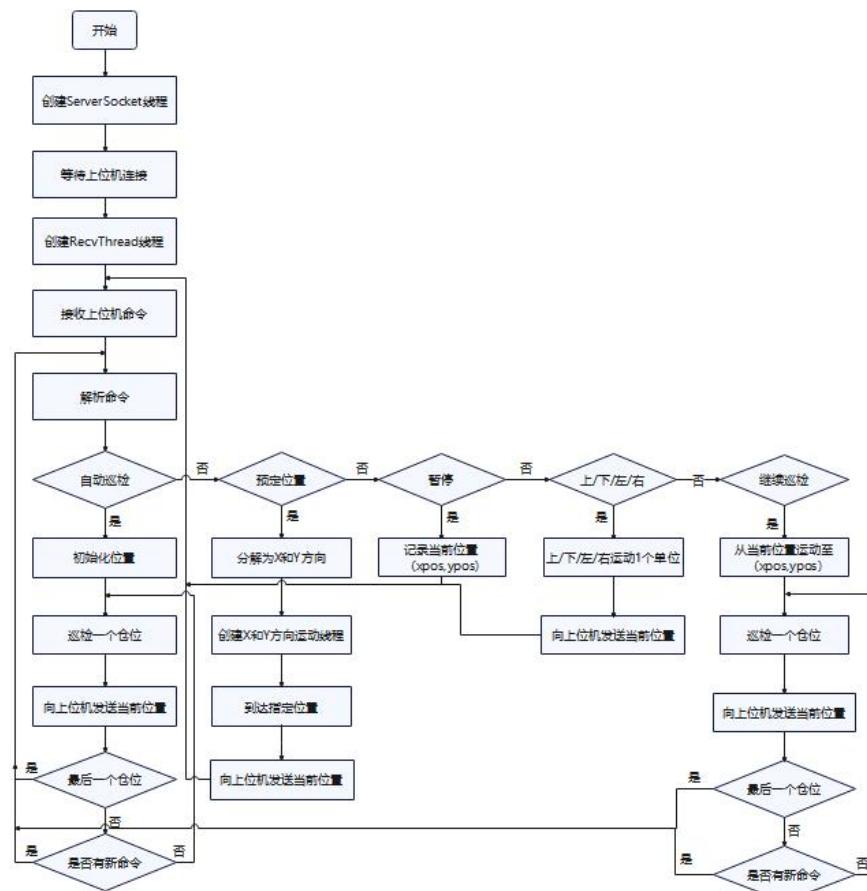


图 3-21: 养殖仓监控摄像头运动控制程序流程图

3.2.5 动物体征智能分析模块

该模块实现了在视频中识别实验鼠，使用 YOLOv5 模型对仓鼠进行目标检测，在 Web 网页端实时显示摄像头画面中进行抓拍，生成的图片会保存在本地。然后在 Web 网页端可以上传图片进行智能分析，调用训练好的 MLP 模型，根据图片中的仓鼠面积预测仓鼠物种，仓鼠的颜色与健康状态以及仓鼠的质量，最后将数据导出，便于实验人员挑选合适的实验对象。

该模块实现流程图见图 3-22：

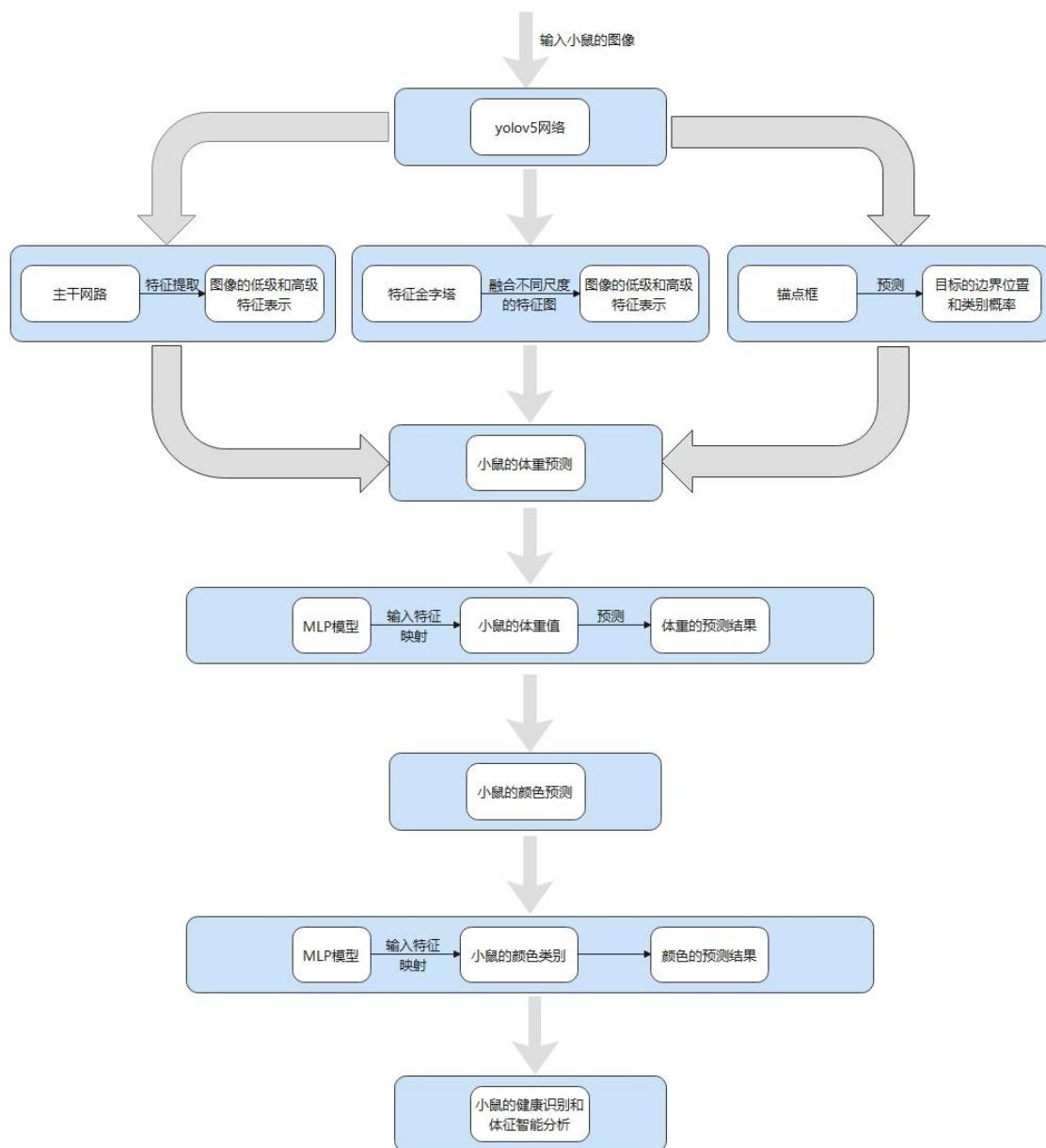


图 3-22：动物体征智能分析实现流程图

下面是该模块的实现过程:

➤ 图像数据采集:

本作品在实验室内部搭建图像采集装置：将两只仓鼠放在同一个仓鼠笼子（1400*430*1500mm）里，使用型号为 DS-2CD3T67WDP2V2-L，序列号为 L30296668 的双目相机，在拍摄后图片深度等于 3 的距离下对装有两只仓鼠的笼子进行录像。通过在 Web 网页端对双目相机进行录制视频操作，视频保存在本地。然后使用解码工具对视频进行抽帧处理，333 毫秒截取一张图片，图片以 jpg 类型存储到指定路径。对采集到的图像进行标记和分类，得到有效数据集共 2998 张，在开源工具 LabelImage 下进行人工标注，见图 3-23，标签为仓鼠-白色/黄色-健康。

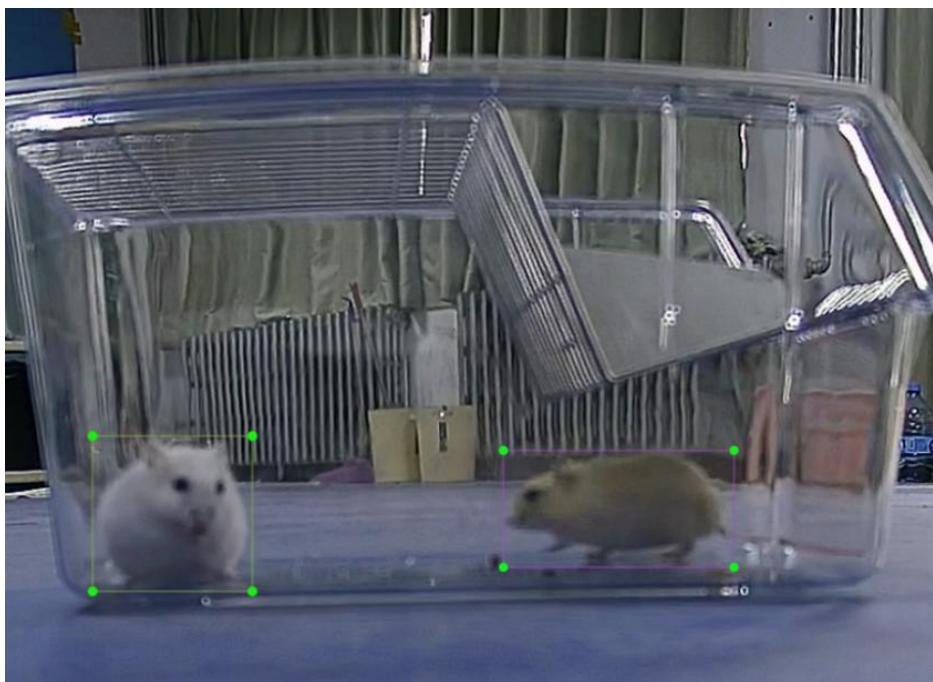


图 3-23：动物体征智能分析实现流程图

标注后选择生成后缀为.xml 的文件，文件中记录有双目相机视野的宽度，高度，深度以及标记目标区域的坐标信息。双目相机直面不同方向采集到的数据按照约 7: 2: 1 的比例随机划分到了训练集（2100 张），测试集（671 张）和验证集（227 张）。

➤ 算法实现:

本实验中采用的目标检测网络是 YOLOv5s 将整张图像划分为若干个网格，每个网格预测出该网格内物体的种类和位置信息，然后依据预测框与真实框之间

的 IoU 值进行目标框的筛选，最终输出预测框的类别和位置信息。输入端采用了 Mosaic 数据增强方式，采用了 4 张图片，随机缩放、随机裁剪、随机排布的方式进行拼接，使模型在更小的范围内识别目标。目标识别使用到的损失函数有 BCEBlurWithLogitsLoss、FocalLoss 和 QFocalLoss。通过将这些函数组合使用，优化模型性能，准确识别图片中的对象，并将其定位到图片中的具体位置。本实验中目标识别使用到的损失函数有 BCEBlurWithLogitsLoss、FocalLoss 和 QFocalLoss。通过将这些函数组合使用，优化模型性能，准确识别图片中的对象，并将其定位到图片中的具体位置。

YOLOv5 网络结构见图 3-24：

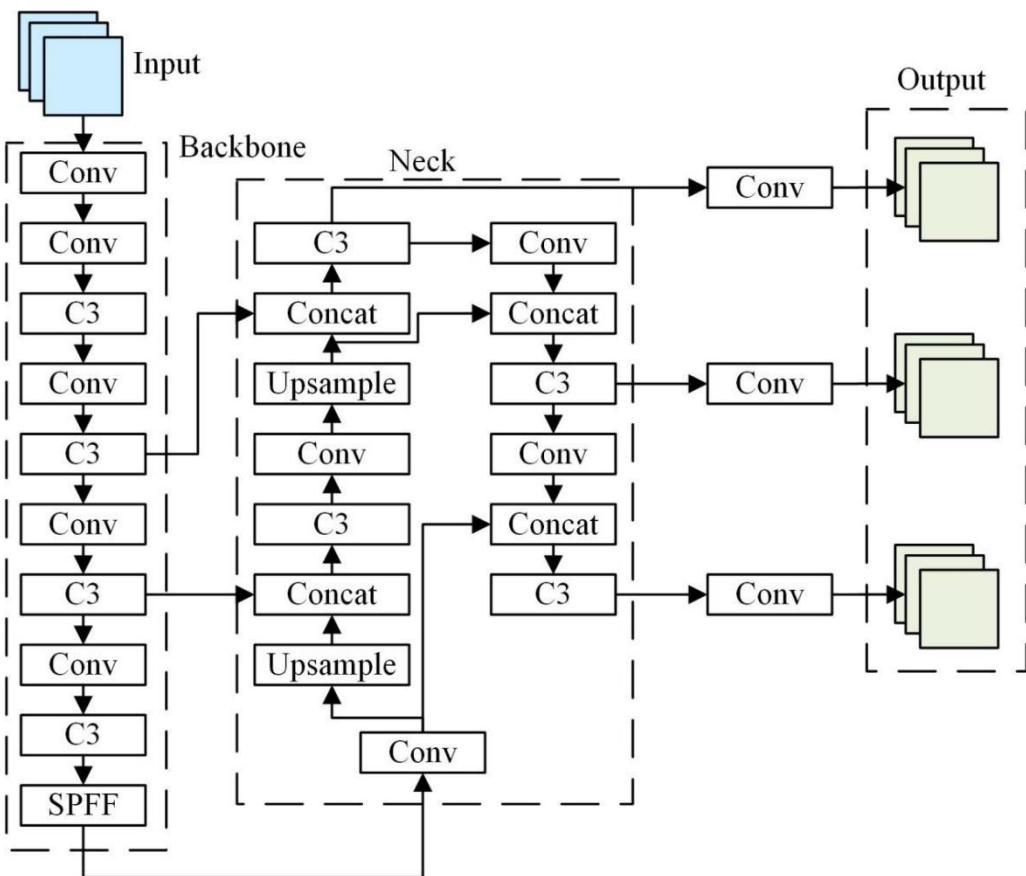


图 3-24: YOLOv5 网络结构图

在 PyCharm、Python 和 Pytorch 环境下，对采集的近 3000 张仓鼠图片进行处理，开发程序对仓鼠图片进行检测，通过训练和比较，最终得到较优的仓鼠目标检测模型。对算法的效果进行分析，首先要对训练中绘制的图片进行分析。

绝大多数的图片中都是两只仓鼠，所以首先应该分析混淆矩阵，见图 3-25。混淆矩阵的每一列代表一个类预测的实例，而每一行代表一个实际的类的实例，

即每一行是预测类别，每一列是真实类别。通过混淆矩阵可以方便地看出机器是否将两个不同的类混淆了。

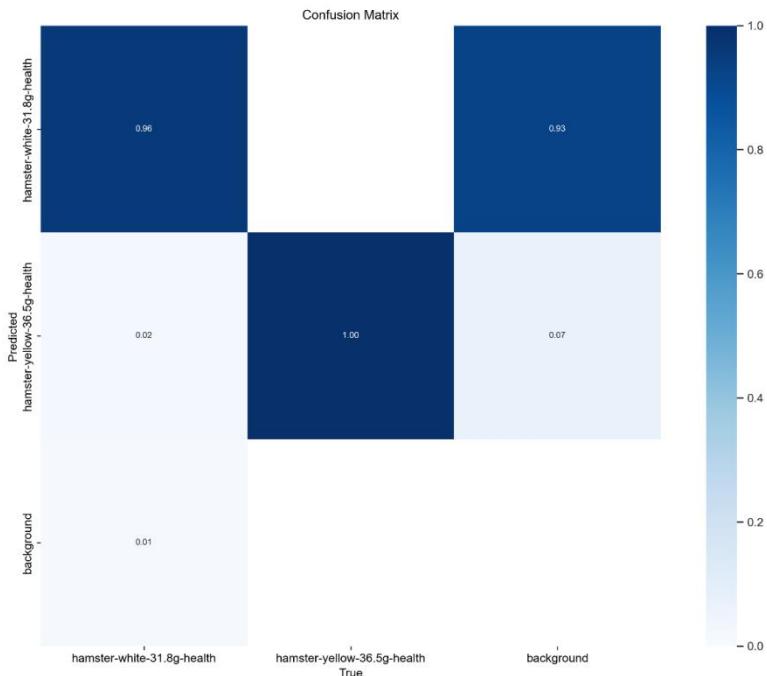


图 3-25: 混淆矩阵图

由上图可以看出，白色仓鼠被预测为白色仓鼠类别的概率是 0.96，而黄色仓鼠被预测为黄色仓鼠类别的概率是 1.00，但同时可以看出白色仓鼠被预测为背景的概率是 0.93。

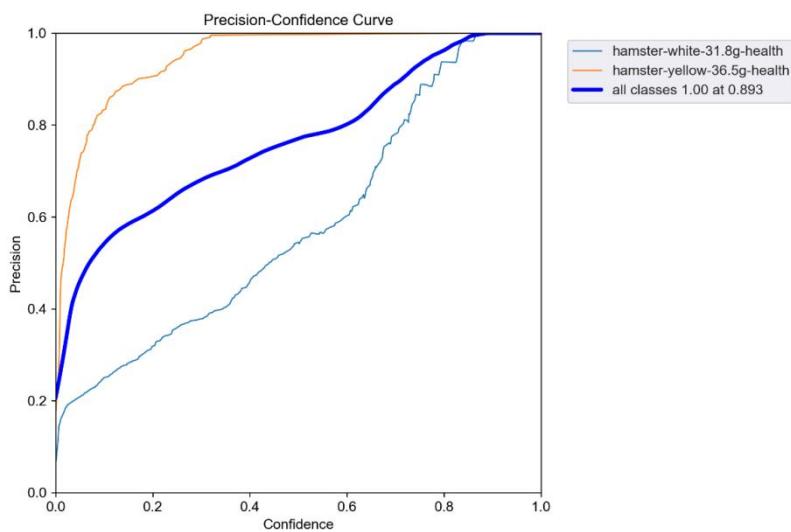


图 3-26: P_curve 图像

P_curve 图像（单一类准确率，即预测为 positive 的准确率）也是准确率

和置信度的关系图，见图 3-26。当设置置信度为某一数值的时候，各个类别识别的准确率。当置信度越大的时候，类别检测越准确，只有当 confidence 很大，才更容易被判断是某一个类别。这样的缺点就是会漏检一些置信度低的类别。那么接着就要分析 R_curve 图像（召回率，真实为 positive 的准确率，即正样本有多少被找了出来），见图 3-27。R_curve 是召回率（查全率）和置信度的关系图，即当设置置信度为某一数值的时候，各个类别查全的概率。当置信度越小的时候，类别检测的越全面。

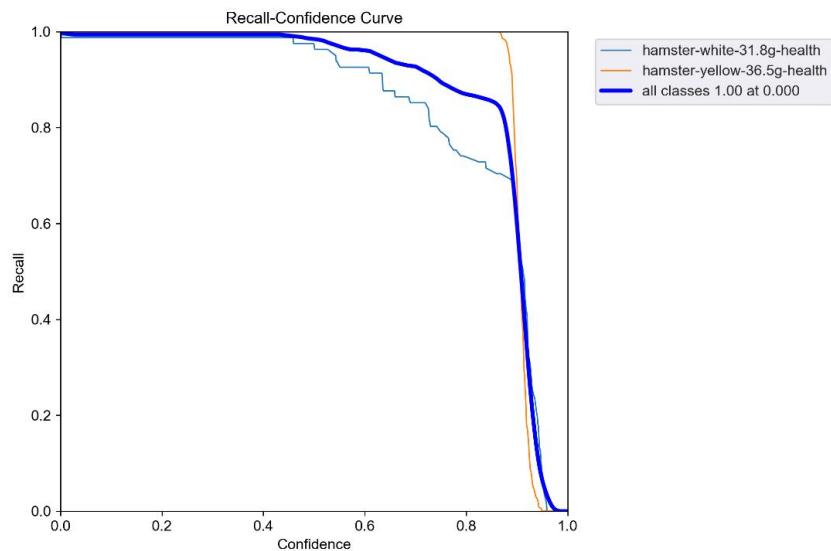


图 3-27: R_curve 图像

PR_curve 是准确率和召回率的关系图，见图 3-28，mAP 是 Mean Average Precision 的缩写，即均值平均精度。

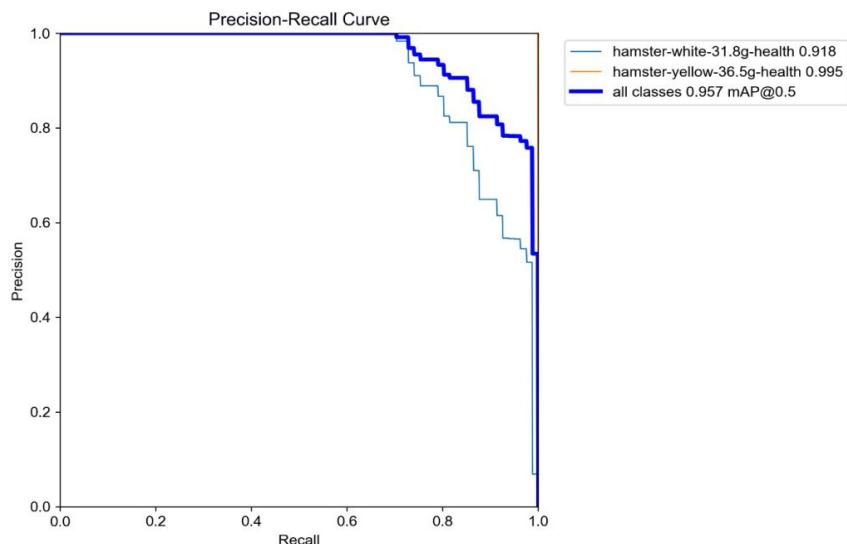


图 3-28: R_curve 图像

在 PR_curve 图中可以看出，精度越高，召回率越低。这与期望是不相符的，要求期望曲线接近 (1,1) 点，即希望 mAP 曲线的面积尽可能接近 1。可见本作品也不能拿这张图片来作为衡量指标。在分类问题中，许多模型的最终测评方法是 F1 分数 (F1-score)，它是精确率和召回率的调和平均数，最大为 1，最小为 0，见图 3-29：

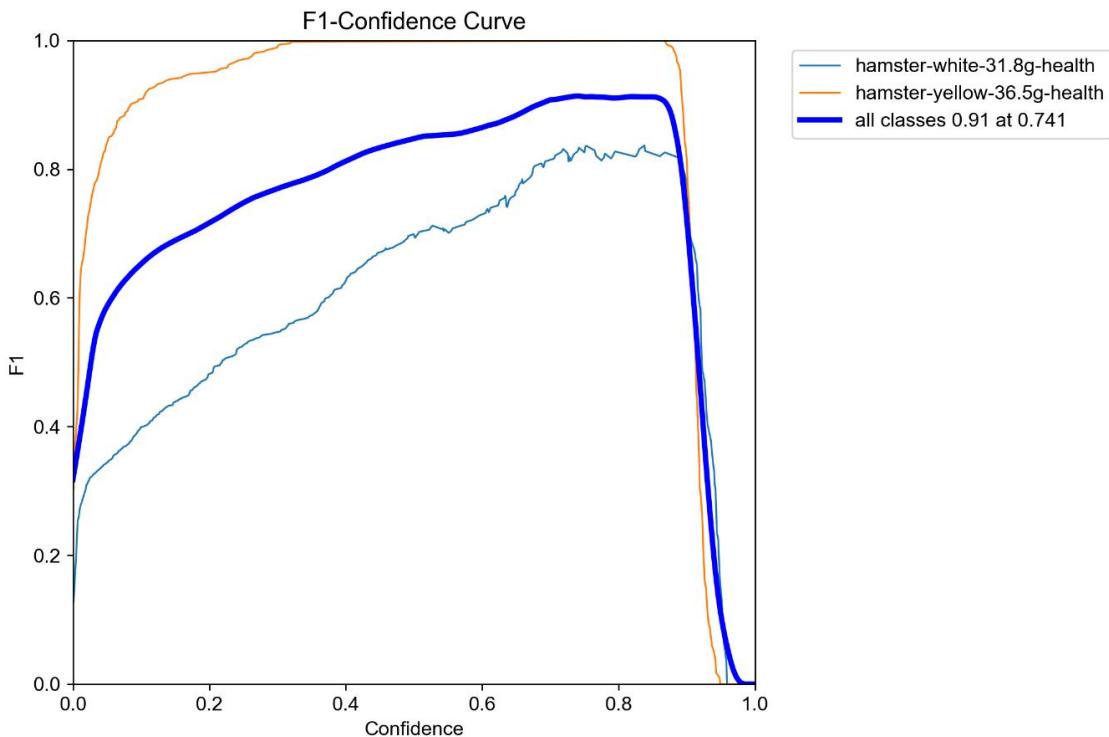


图 3-29：F1-score 图像

对于白色和黄色仓鼠的分类，综合了 Precision 和 Recall 的一个判断指标，F1-Score 的值是从 0 到 1 的，1 是最好，0 是最差。F1Score 的公式为 (3-1)。

$$F1Score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (3-1)$$

分析完评测标准，然后要对标签图片进行分析。首先是 labels 图片，见图 3-30：

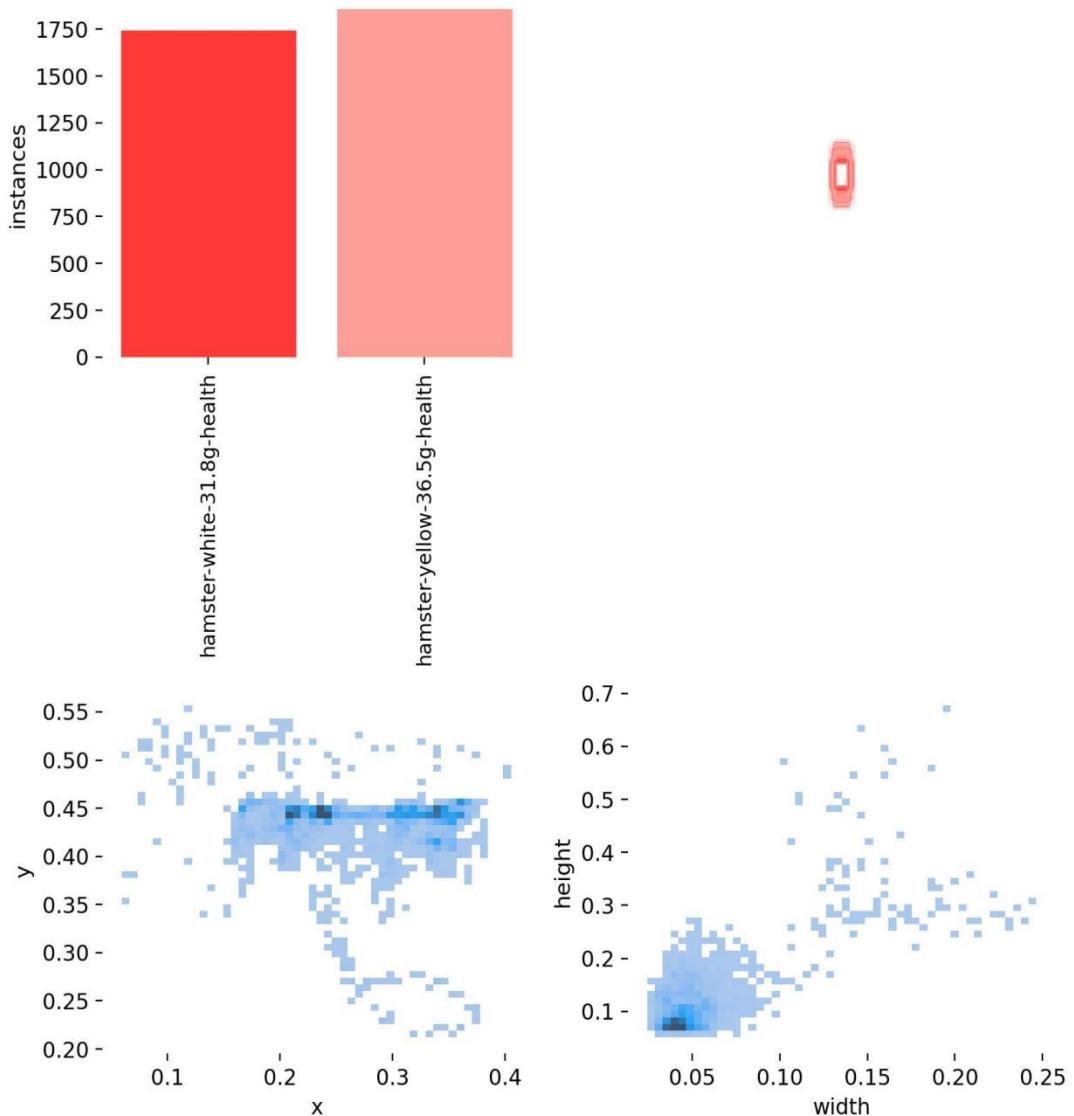


图 3-30: labels 图像

第一张图为训练集的数据量，每个类别分别有多少张图片。第二张图片是检测框的尺寸和数量。第三张图是 center 点的位置，可以看到数据集中仓鼠的位置随机分布，但是大多集中在 (0.2-0.4,0.4-0.45)；第四张图是 label 的高和宽，仓鼠一般相对于整张图片比较小，所以看到的样本大多分布在 (0-0.05,0-0.1)。

损失函数是用来衡量模型预测值和真实值不一样的程度，极大程度上绝对了模型的性能。主要有三种损失：定位损失 box_loss，预测框与定位框之间的误差；置信度损失 obj_loss，计算网格的置信度；分类损失 cls_loss，计算锚框与对应的标定分类是否正确。设置训练轮数为 100，批次为 16 进行训练。最终得到模型验证集目标检测 loss 均值收敛于 0.01，验证集分类 loss 均值稳定在 0.05 以下，

精度保持在 0.95，召回率虽有波动，但是起伏不大，保持在 0.85 以上，结果见图 3-31。

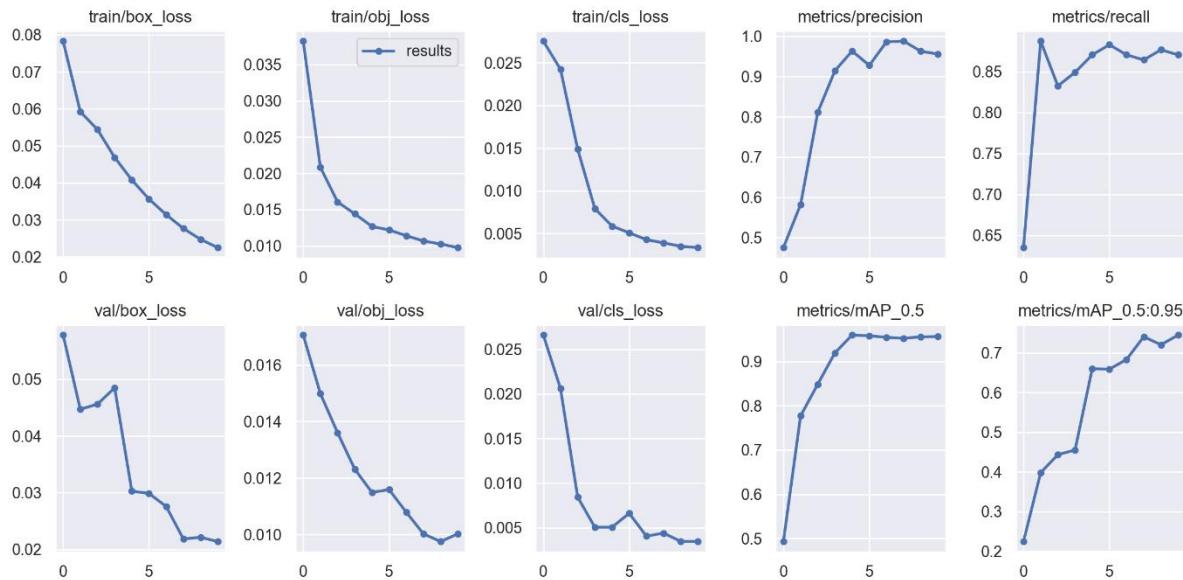


图 3-31：目标检测结果图

上面的数值会生成一个 csv 文件方便对数值进行查询，取 10 次结果如下（图 11），前三列和后三列分别是训练集的定位损失、置信度损失、分类损失和验证集的定位损失、置信度损失和分类损失；中间四列分别是精度（找对的正类/所有找到的正类）、召回率（真实为 positive 的准确率）、表示阈值大于 0.5 的平均 mAP、表示在不同阈值上的平均 mAP。

	train/box_loss	train/obj_loss	train/cls_loss	metrics/precision	metrics/recall	metrics/mAP_0.5	metrics/mAP_0.5:0.95	val/box_loss	val/obj_loss	val/cls_loss
0	0.078314	0.038246	0.027529	0.47496	0.63501	0.49312	0.22443	0.057792	0.01706	0.026608
1	0.059231	0.020833	0.02423	0.582	0.88714	0.77717	0.3984	0.044691	0.014983	0.020607
2	0.054397	0.016051	0.014879	0.8111	0.83241	0.8488	0.44357	0.045606	0.013603	0.0084282
3	0.046867	0.014409	0.007915	0.91463	0.84901	0.91967	0.45538	0.048477	0.012313	0.0050444
4	0.040769	0.012673	0.0058668	0.96337	0.87037	0.95996	0.66058	0.03026	0.01149	0.0050548
5	0.03561	0.01218	0.0050766	0.92777	0.88272	0.95825	0.65924	0.029855	0.011589	0.0066113
6	0.031385	0.011392	0.0042767	0.98632	0.87037	0.95479	0.68388	0.02756	0.010788	0.0040528
7	0.027655	0.010663	0.0038957	0.98807	0.8642	0.95277	0.74124	0.021825	0.010021	0.0043904
8	0.024717	0.010241	0.0034983	0.9628	0.87654	0.9558	0.72089	0.022122	0.0097508	0.0034623
9	0.022479	0.0097475	0.0033434	0.95598	0.87037	0.95658	0.74547	0.021375	0.010022	0.0034464

图 3-32：训练数值

最后需要看的就是 train_batchx 和 val_batchx_labels&val_batchx_pred。设置的 batchsize 是 16 故一次读 16 张照片，见图 3-33：

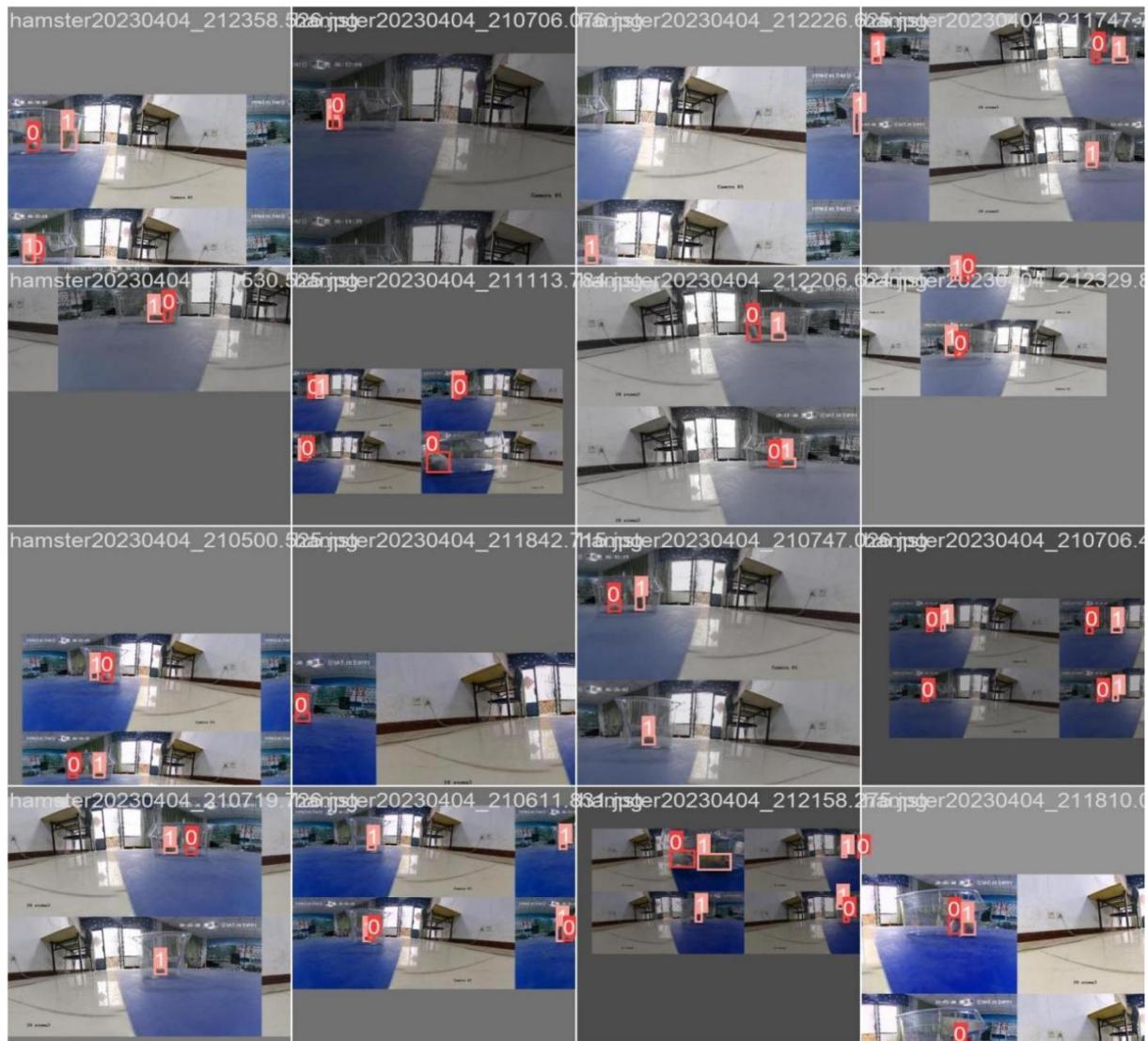


图 3-33: train_batch0



图 3-34: val_batch_labels



图 3-35: val_batch_pred

在 PyCharm 上输入新的拍摄图片进行模型测试, 输出得到了小鼠编号, 颜色, 体重, 健康情况四个指标。

在得到识别结果后, 通过将得到的数据语言化发送给 ChatGPT 询问养殖的建议。得到了相应的有针对性的建议。在这里使用的是 OpenAI 提供的相应接口。可以提供 prompt 指定 GPT 所专攻的领域, 然后使用 ChatCompletion.create() 函数即可得到 GPT 响应的结果。

3.3 本章小结

本章详细地阐述了实验室管理系统的方案实现。首先介绍了开发平台和工具的选择, 然后将系统的功能分为了集中式监控大屏模块、用户管理模块、环境监测与可视化模块、养殖仓视频监控模块、动物体征智能分析模块五个主要模块。

用户管理模块实现了用户的鉴权登录、用户注册、用户管理和日志管理, 确保了系统的安全性和数据的准确性, 同时为管理员提供了方便的操作界面。

集中式监控大屏模块利用云计算、边缘计算、大数据分析等技术, 将分布在不同地区的实验室数据进行集中管理, 展现在一体化的大屏幕上, 提高了数据的可视性和实时性, 解决了信息断层和不一致的问题。

养殖仓视频监控模块实现控制双目摄像头运动, 支持自动巡检、手动巡检、视频存档、实时显示等功能。

环境监测与可视化模块实现实时采集各个实验仓的环境数据, 包括温度、湿度、二氧化碳浓度和光照强度等, 然后上传至阿里云服务器进行数据处理与分析,

并在前端进行可视化展示，以便实时监控和调整实验室的环境状态。

动物体征智能分析模块能够识别和分析实验用的仓鼠，包括仓鼠的种类、颜色、健康状况以及质量。并给出合理的养殖建议。

四、应用和测试效果

4.1 Web 网页端应用效果

➤ 页面一：集中式监控大屏见图 4-1：



图 4-1：集中式监控大屏

用户可以点击地图中实验室查看某一实验室具体数据，以及统计的全国各省份的实验室异常报警信息，通过点击开关控制实验室所有设备开关，查看全国实验室数量分布与生物健康状态分布，分析各月份实验室生物总量走势。

➤ 页面二：欢迎页面见图 4-2：



图 4-2:欢迎页面

➤ 页面三：用户管理模块见图 4-3，图 4-4：

The screenshot shows the user management page of the system. The left sidebar has the same navigation menu as the welcome page. The main area displays a table of users:

No.	账户名称	备注	操作
1	张三	实验室主任	<button>删除</button> <button>编辑</button>
2	李四	实验室副主任	<button>删除</button> <button>编辑</button>
3	七大姑	实验室工作人员	<button>删除</button> <button>编辑</button>
4	八大姨	实验室工作人员	<button>删除</button> <button>编辑</button>

A blue '增加' (Add) button is located below the table. A watermark '© 基于AIoT的智慧动物实验室管理系统 2023' is at the bottom right.

图 4-3: 用户管理页面

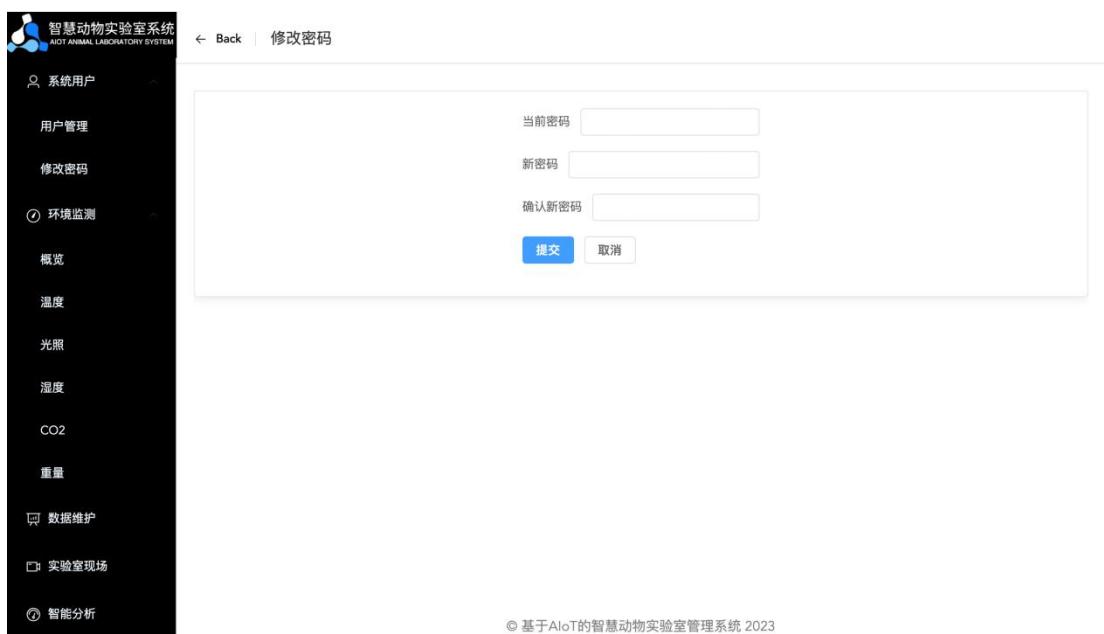


图 4-4: 修改密码页面

➤ 页面四：实验室环境数据概览页面见图 4-5：

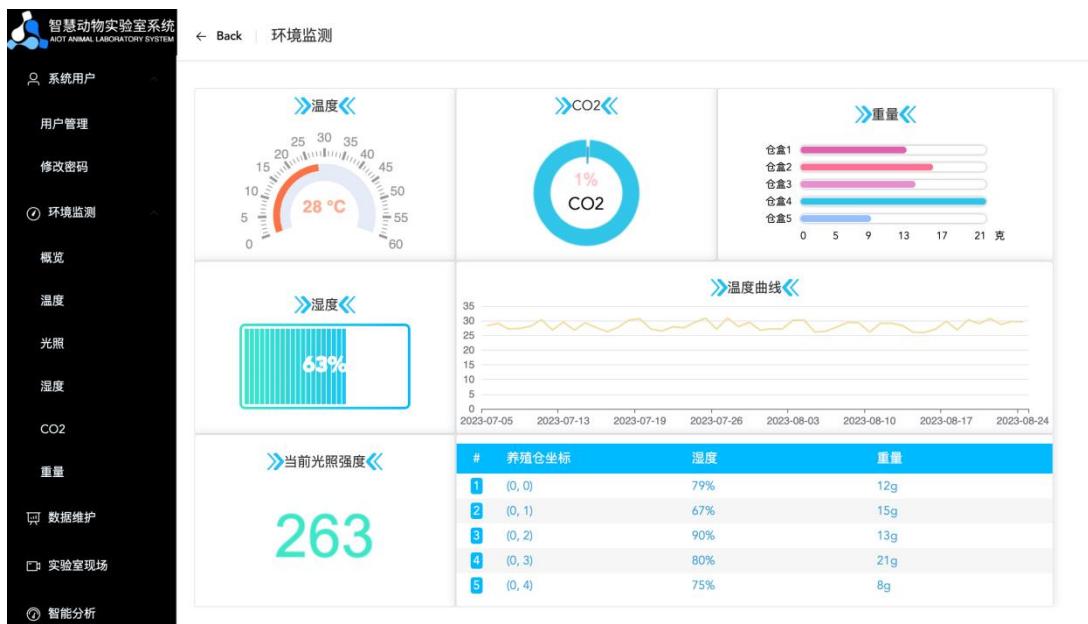


图 4-5:环境概览页面

环境监测概览页面展示当前实验室温度、湿度、光照强度，以及各个实验仓的数据动态刷新轮播图。

➤ 页面五，六为温度，光照数据曲线，见图 4-6，4-7：

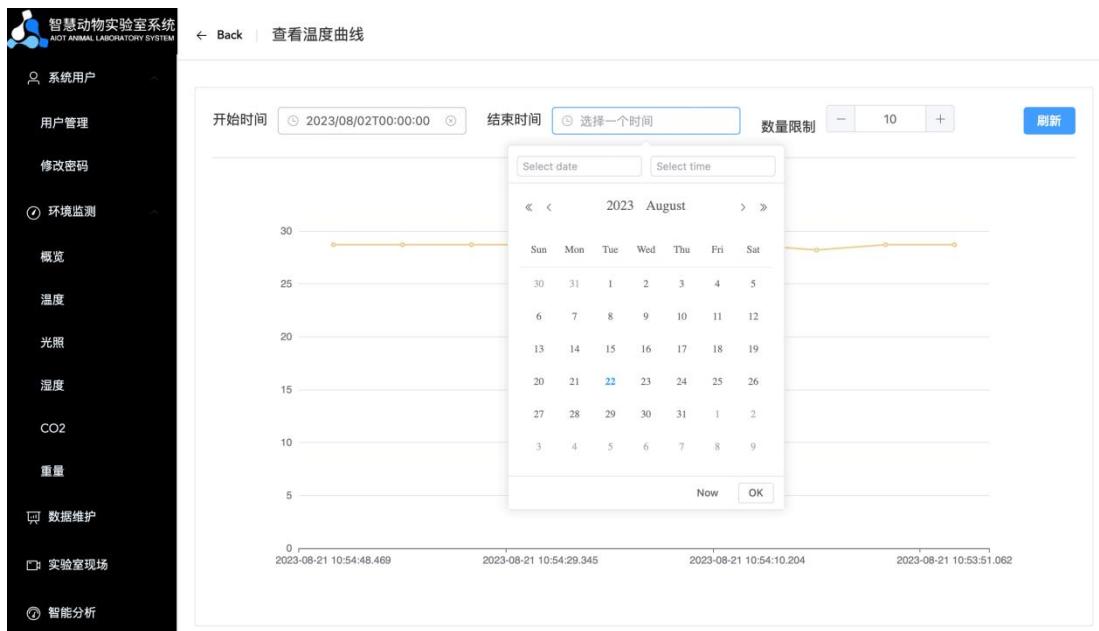


图 4-6:温度随时间变化曲线

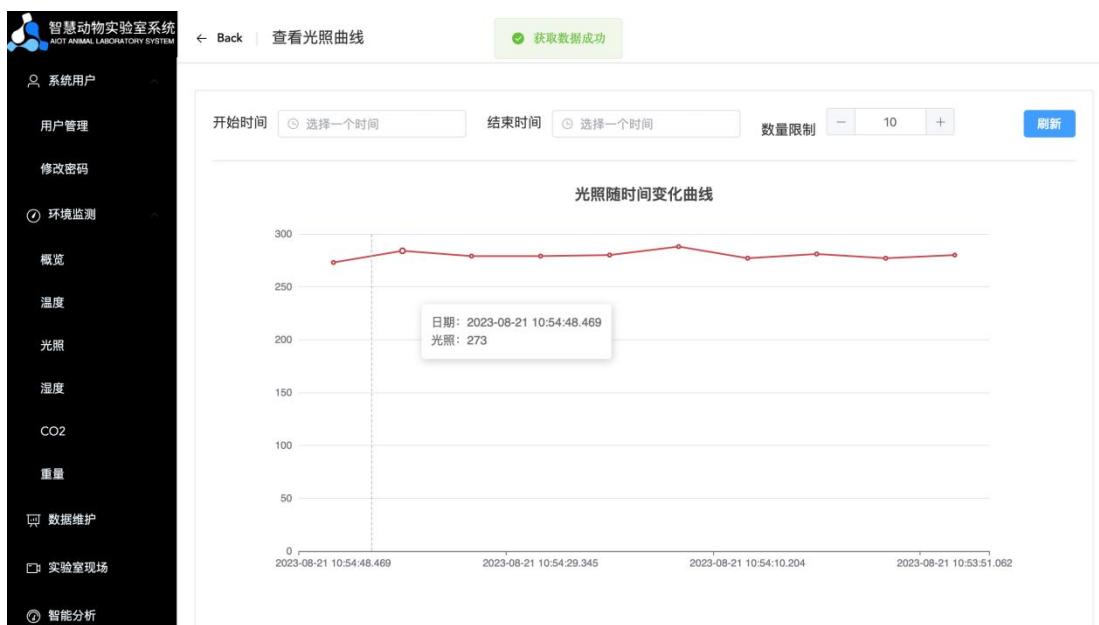


图 4-7:光照随时间变化

➤ 页面七：数据维护页面，见图 4-8：

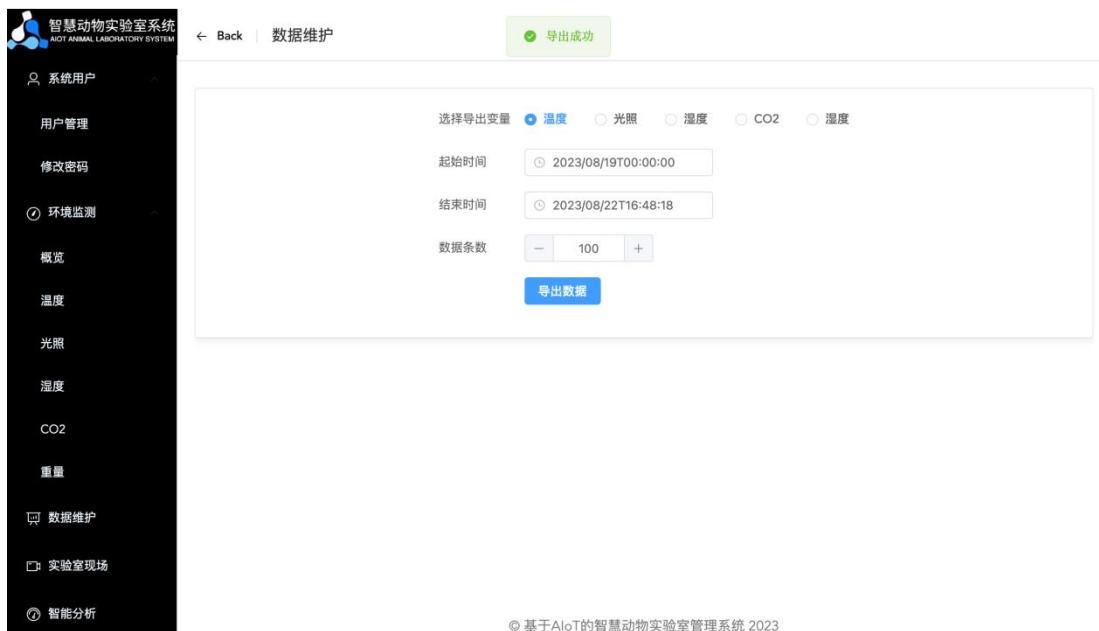


图 4-8:数据维护页面

用户可以选择导出变量，起始时间，结束时间，数据条数 导出数据表格用于实验数据分析。

➤ 页面八：实验室现场页面，见图 4-9：

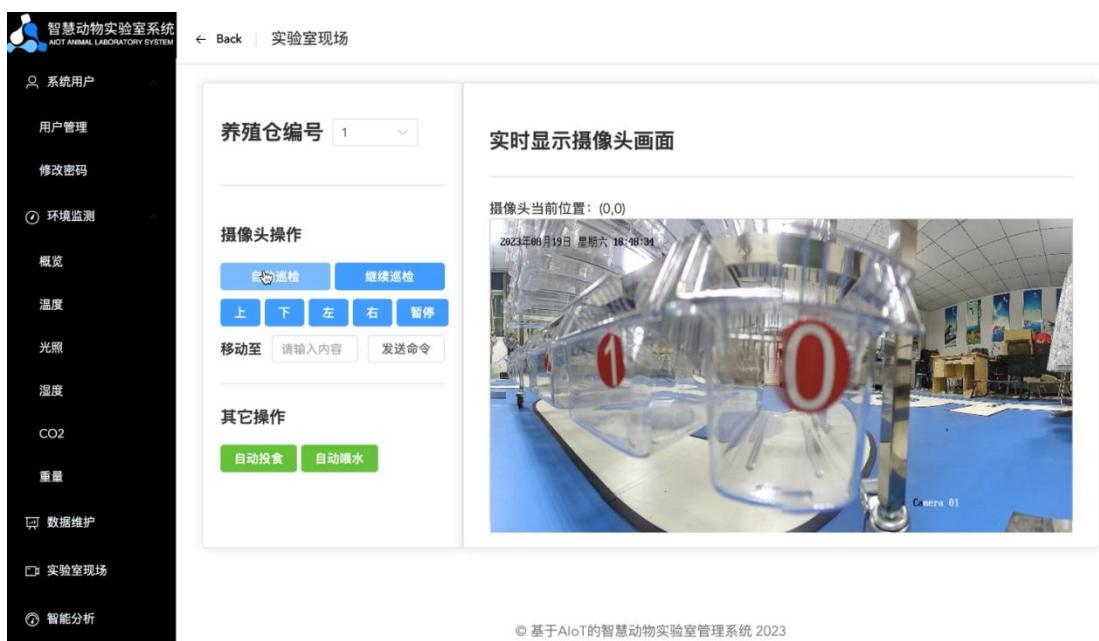


图 4-9:实验室现场页面

用户可以通过简单的点击，控制摄像头的运动，包括自动巡检，上，下，左，右，暂停，输入养殖仓序号。指定双目摄像头到达某一养殖仓位置并观看实时监控，实时在后台截图，保存到指定路径。该页面也提供投喂，喂水开关。

- 页面九：智能分析页面，见图 4-10：



图 4-10: 智能分析页面

用户可以导入小鼠图片，调用训练好的目标检测模型，得到分析后的小鼠体征数据，包括种类，颜色，体重，健康情况，以及测试情况。并通过询问 ChatGPT，得到养殖建议。

4.2 小程序应用效果

- 页面十：小程序登录和个人页面，见图 4-11：

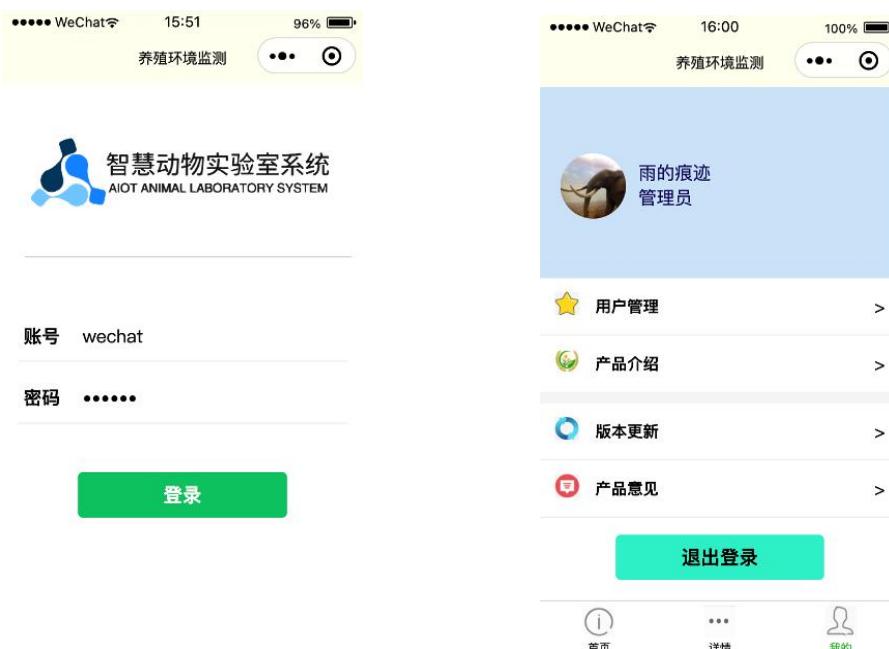


图 4-11: 小程序登录和个人界面

小程序支持多用户登录。并且在个人界面有用户管理，产品介绍版本，版本更新，产品意见等功能。

➤ 页面十一：小程序环境监测页面，见图 4-12：

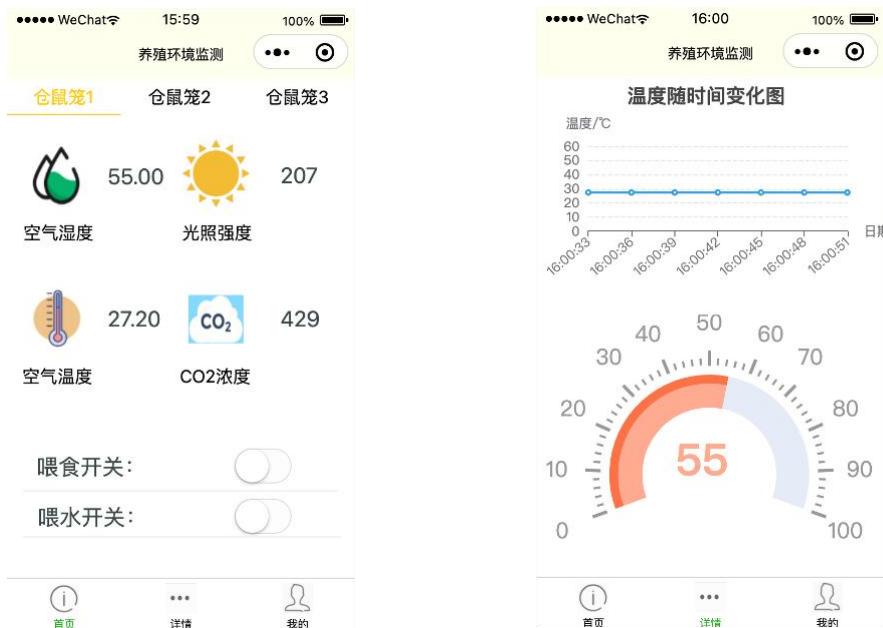


图 4-12：小程序环境监测界面

环境监测界面可以查看养殖场仓当前环境数据，如空气湿度，光照强度，空
气温度，CO₂ 浓度，以及一段时间的温度。

4.3 测试报告

4.3.1 硬件电气特性

表 4-1：硬件电气特性

特性名称	测试结果	备注
设备工作电流	6A	电机的稳定工作电流
最低工作电压	36V	低于此电压电机将不能运行
最高工作电压	42V	电机最大承受电压

4.3.2 软件鲁棒性

表 4-2: 软件鲁棒性

测试项目	测试过程	测试结果
自动巡检	让设备进行自动巡检 10 次	通过
路由器故障(网络断开)	巡检时关闭路由器电源	摄像头断开, 无法采集图像
路由器网络恢复	再次打开路由器电源	摄像头需要平台来进行重连
工作台断电	巡检时断开工作台的电源	电机无法运行
单片机断电	断开单片机的供电	上位机断开, 单片机不工作
环境错误	巡检时遮挡摄像头	采集的图像不完整

4.3.3 软件运行

在长时间的运行测试中，集中式地图大屏表现出了出色的性能。我们没有观察到明显的卡顿现象，而摄像头的实时显示画面一直保持了流畅性，这为实验室管理人员提供了实时而清晰的数据展示。

对于小程序的启动过程，我们发现初次启动可能需要一些数据加载的缓冲时间，然而这种启动时的延迟在后续的启动中几乎是不可察觉的。这意味着，用户在以后的使用中将能够享受到更快的响应速度，增强了整体的用户体验。

4.3.4 模组运行速度

(1) X 方向运行速度

表 4-3: X 方向运行速度

测试项目	运行速度
向右运动	0.029m/s
向右运动	0.030m/s
向左运动	0.046m/s
向左运动	0.045m/s

(2) Y 方向运行速度

表 4-4: Y 方向运行速度

测试项目	运行速度
向上运动	0.031m/s
向上运动	0.03m/s
向下运动	0.045m/s
向下运动	0.046m/s

4.3.5 安全性

在摄像头与路由器使用中需要上网，所以要测试其安全性，见表 4-5:

表 4-5: 安全性

测试项目	测试过程	测试结果
路由器用户登陆	使用错误的用户名和密码	不能登陆
摄像头用户登陆	使用非管理员用户登陆	不能修改参数

4.3.6 部署方便性和可用性

在新的电脑上测试网页服务，测试结果见表 4-6：

表 4-6：测试结果

测试项目	部署方式	部署用时	测试结果
网页访问	配置 NGINX 服务 器，启动 flask	10min	远程访问成功
摄像头访问	配置摄像头 IP	11min	网页访问视频成功
连接树莓派	树莓派开机后连接	<1min	连接成功，可以正常控制模组工作

4.4 本章小结

这一章主要介绍了实验室环境监控系统的应用效果和测试结果。应用效果部分包括 Web 网页端和小程序的具体界面展示和功能描述，例如集中式监控大屏、环境监测概览、用户管理、实验数据维护、现场实时监控、智能分析等模块。测试部分详细解释了硬件电气特性，软件鲁棒性，软件运行状况，模组运行速度，系统安全性，部署方便性和可用性等各方面的测试过程和结果，总体上验证了系统的工作性能和稳定性。

五、创新与特色

本作品提出的“生物智联”——基于 AIoT 的智慧动物实验室管理系统，实现了边云协同、大数据分析、设备自动化、目标检测、动物体征检测、异常检测、远程监控、地图管理等功能，从“万物互联”到“万物智联”，全方位保障实验室的行为、管理、环境和数据等全景安全，建设了数字科技与生物安全技术创新融合的智慧动物实验室，实现实验室管理的自动化、智能化、物联化、共享化。

本作品的创新与特色的可以从四个方面概括为四个字“联”，“大”，“智”，“全”。

5.1 “联”：使用边云协同实现实验室互联

本作品采用边缘计算、云计算等先进技术，实现了实验室联网的创新解决方案。这一方案能够实现集中式数据监控和分布式设备控制，将实验室设备、仪器和实验数据紧密地连接在一起，为实验室管理带来了巨大的便利和效益。通过在实验室内部部署边缘计算设备，例如传感器、智能终端等，可以实现对实验数据的实时采集和处理。这些边缘设备能够快速处理大量的数据，并将关键的数据传输到云端进行进一步的分析和存储。采用边缘计算技术，可以降低数据传输的延迟和带宽消耗，提高数据处理的效率和实时性，使得实验室监控和控制更加精确和可靠。

5.2 “大”：使用大数据分析处理海量数据

本作品应用大数据分析和可视化技术来处理海量的实验数据，从而获得更深入的洞察和更好的决策支持。实验室会产生大量数据，包括实验结果、仪器输出、环境参数、设备状态等多种类型的信息。传统的数据处理方法可能无法高效地处理和分析如此海量的数据，但利用大数据分析与可视化可以帮助实验室管理者进行资源规划和预测，通过历史数据和趋势分析，合理安排设备使用和人员调度，提高实验室的整体运营效率。将大数据分析的结果以可视化的方式呈现，如图表、图形、雷达图等，能够让实验室管理人员直观地理解数据的特征和趋势。

5.3 “智”：使用智能化技术赋能解放人力

本作品应用智能化技术，如人工智能、自动化、机器学习等技术，来实现实

验室的智能化管理和操作，从而解放人力，提高工作效率和质量。本作品利用物联网技术，实现实验室设备和仪器的自动监测和维护。通过将传感器和监测设备与智能系统连接，我们可以实时监测设备的运行状态和性能指标。本作品利用机器学习和模式识别算法，训练动物体征智能系统，使其能够自动分析动物健康情况，预测实验结果，确保实验的准确性。

5.4 “全”：使用自上而下的全面设计方案

本作品综合考虑各种因素，包括技术、市场、用户需求、可行性等，设计了从应用层，平台层，网络层，感知层自顶向下的全面方案。在应用层，我们深入了解动物实验室需求和市场趋势，准确把握传统实验室的期望和痛点，实现了小程序和 Web 网页多端协同的两种应用模式；在平台层，本作品选择阿里云平台，兼顾可扩展性和灵活性，以支持未来的功能拓展和系统升级；在网络层，们设计了高效可靠的网络架构，以支持实时数据传输和远程通信，选择了合适的网络协议和通信技术，并进行了网络优化和安全性的考虑，以确保数据的高速传输和保密性；在感知层，我们合理进行硬件选型，布置传感器节点和感知设备，我们可以实时采集环境数据、设备状态和用户行为等信息。这些数据为系统的智能决策和自适应调整提供了基础，以提供更智能、高效的服务和功能。