



安徽建筑大学
ANHUI JIANZHU UNIVERSITY

毕 业 论 文

题 目: 基于 NVIDIA-jetson 的智能门禁系统

姓 名: 许准

学 号: 18205010212

学 院: 电子与信息工程学院

专 业: 电子信息工程

指导教师: 邵慧

完成时间: 2022 年 06 月 01 日

摘 要

采用 NVIDIA-jetson Nano (2GB) 作为主控模块, 结合 STM32 单片机和相关传感器完成基于 NVIDIA-jetson 的智能门禁系统的设计, 实现实时人脸解锁、自动发送报警邮件、指纹和密码方式解锁、图形用户界面实时修改密码或指纹、无接触测量温度、实时语音合成及播报等功能。

通过索尼 IMX219 摄像头捕获实时视频流, 基于 Python 工具 face_recognition 完成人脸的实时检测与识别工作。在系统图像捕获范围内, 首次出现未经注册的面孔, 将截取该帧图像并以附件的形式自动发送至主人指定的电子邮箱内。除验证人脸方式外, 用户可手动选择验证指纹或数字密码方式完成门禁的解锁; 同时支持更改密码、新增指纹、删除指纹等功能。在锁定状态下可进行无接触体温测量; 使用 XFS5152CE 实现相关提示语的实时合成及播报功能。实验结果表明, 设计的系统识别人脸准确, 功能表现稳定、可靠。

关键词: NVIDIA-jetson; 邮件自动报警; 人脸识别; STM32; 语音合成

ABSTRACT

An intelligent access control system based on NVIDIA-jetson nano (2GB), STM32 and some other sensors is designed, which can realize multiple functions, such as unlocking in real-time face recognition, automatically sending alarm email, unlocking in fingerprint and password mode, real-time password or fingerprint modification in graphical user interface, contactless temperature measurement and real-time speech broadcasting.

The designed system captures real-time video stream with Sony IMX219 camera, and then the real-time face detection and recognition is conducted with face_recognition(Python). If an unregistered face appears for the first time within the image capture range, the system will extract a frame image of the unregistered face and automatically send it to the owner email. In addition to face verification, users can manually select fingerprint verification or digital password to unlock the access control system. The system provides the features to change the password, add and delete fingerprint, also supports contactless temperature measurement and voice broadcast with XFS5152CE. The experimental results show that the system is accurate in identification, and it is reliable and stable.

Key words: NVIDIA-jetson Nano(2GB); Automatic mail alarm; Face recognition; STM32; Speech synthesis

目 录

第一章 引言	1
1.1 课题背景与研究现状	1
1.2 研究目的及意义	1
1.3 论文结构组成	2
第二章 系统总体方案与硬件设计	3
2.1 总体方案	3
2.2 器件选型	5
2.2.1 主控选型	5
2.2.2 辅助控制模块选型	6
2.2.3 系统输入输出接口选型	7
2.2.4 摄像头选型	7
2.2.5 视频流输出接口选型	8
2.2.6 指纹采集模块选型	8
2.2.7 网络连接方案	9
2.2.8 实时语音合成芯片选型	9
2.2.9 舵机选型	9
2.3 系统逻辑功能部署	10
2.4 系统供电电源设计	10
第三章 系统软件设计	12
3.1 总体逻辑架构	12
3.2 jetson Nano（2GB）部分	12
3.2.1 烧录 JetPack 镜像	12
3.2.2 配置编译环境	13
3.2.3 face_recognition 库环境	13
3.2.4 邮件报警	15
3.2.5 建立人脸库	16
3.2.6 调用 CSI 摄像头获取实时画面	17

3.2.7 实时识别人脸	17
3.3 STM32 单片机部分	20
3.3.1 AS608 程序设计	20
3.3.2 MLX90614 程序设计	24
3.3.3 XCFS5152CE 程序设计	25
3.3.4 数字舵机模块程序设计	26
3.3.5 STM32 主程序设计	27
3.4 TJC4827X343 液晶屏部分	29
第四章 实验结果与分析	31
4.1 装配系统	31
4.2 jetson 电源配置测试	31
4.3 系统稳定性测试	32
4.4 确定人脸特征欧氏距离阈值	33
4.5 功能实现	34
第五章 总结与展望	38
5.1 总结	38
5.2 后续工作展望	38
致 谢	39
参考文献	40

第一章 引言

1.1 课题背景与研究现状

人脸识别技术^[1]是一门20世纪70年代出现的技术。人脸识别技术的发展可分为4个阶段。第一阶段为点对点对比完成识别；第二阶段为使用主成分分析法和特征统计进行人脸识别；第三阶段是在21世纪初，重点研究非受限的人脸识别；第四阶段为2014年至今，重在研究精准度更高的识别问题。国内外人脸识别技术发展迅猛，如目前百度人脸识别API，可实现对比人脸库进行1:N检索，可满足人脸门禁等应用场景。王佳颖等^[2]利用百度AI平台SDK设计了一种人脸识别学生考勤系统，提高了考勤效率。Marwa Mohammed^[3]等发明了一种基于纹理分析的高效人脸识别方法。

人脸识别具有快速性和便捷性。人流量大的公共场合^[4]如火车站^[5]、图书馆^[6]等常配备快捷的人脸识别门禁装置，便于人们快速、自主地进行身份验证。而对于私人场所，如家庭、宿舍、卧室等，对于此类门禁装置需要具有更高的安全性和私密性。

人脸门禁系统的设计常使用ARM架构处理器、FPGA等实现。范世朝等^[7]使用现场可编程的门阵列（FPGA）设计了基于FPGA和卷积神经网络的人脸识别系统。王浩^[8]利用K210芯片的KPU神经网络加速器实现了快速人脸检测。为提高性能，人们将NVIDIA-jetson平台^[9]引入门禁系统。周宇^[10]等利用NVIDIA-jetson开发平台实现了智能门锁的设计。利用OpenCV及相关人脸识别算法的部署实现了人脸的检测及识别，利用jetson开发板的输入输出引脚（GPIO）控制门锁开闭。胡佳玲等^[11]将改进型轻量级卷积神经网络Mobile部署在Jetson上，大幅减小运算量，提升识别速度。

门禁系统的开发过程利用嵌入式硬件系统结合人脸识别算法进行设计，可应用于快捷支付、企业日常考勤等场合，是人们的生活更为便利、安全。应用于公共场合的门禁系统、闸机系统已日趋成熟和完善，而应用于私人场合的门禁设计工作尚开展不足。因此，针对私人住所设计一套智能人脸门禁系统非常必要。

1.2 研究目的及意义

当今科技发展迅速，一套功能全面的智能化门禁系统能极大简化用户的出入与通勤，在一定程度上改变了人们的生活方式。相对于指纹验证等其他识别方式，人脸识别具有无接触、高效便捷等优势。人脸识别技术在门禁系统中的应用已非常广泛、普

遍，多用于大客流量场合。此类门禁系统内置算法成熟、识别速度快、准确，但维护成本高、安装复杂、功能单一、功耗高，不适于应用在私人场所。

由此，本文针对私人场所安全性与私密性需求，设计一种硬件人脸识别门禁装置，兼具预警等功能，其设计成本较低、易于安装与维护、功能多样。由于所需检索的人脸库较小，识别速度较快；仅需连接网络而无需任何在线配置操作，物联复杂度低。

1.3 论文结构组成

第一章，引言。介绍本系统设计相关的背景及相应来源、意义与必要性。阐述与总结设计思路。

第二章，系统总体方案与硬件设计。主要论述装置的各个功能及相应的实现方法；将各部分可选功能元器件的相关参数进行了对比、选择，完成了器件选型工作。设计了系统的电源模块，为除 jetson 以外的所有功能模块进行供电。

第三章，系统软件设计。对整个系统的开发设计工作进行了统筹与逻辑架构搭建，而后针对每一大模块的设计工作进行较详尽的阐述与分析，完成程序软件功能实现。

第四章，实验结果与分析。系统所有的逻辑功能均开发完成后，将硬件装配为一个完整的产品；对系统相关功能部件及整体的稳定可靠性进行了测试；对影响系统实际功能效果的部分重要参数进行实验与分析，并确定其最佳数值，最后完成系统功能的实验和分析。

第五章，总结与展望。对本次的设计工作进行客观总结；对今后的改进工作进行展望。

第二章 系统总体方案与硬件设计

2.1 总体方案

本设计应用场景为私人场所，功能要求丰富，对个性化功能要求更高。主要组成部分有 NVIDIA-jetson Nano(2GB)、无线网卡（局域网线亦可）、视频流实时 HDMI 触摸显示屏、CSI（信道状态信息）接口摄像头、STM32F103 单片机最小系统板、触摸液晶屏、红外温度测量传感器、指纹对比模块、语言合成芯片（模块）及其音频功率放大和播放电路、数字舵机、系统电源模块等。

系统的功能结构框图，如图 2.1 所示。

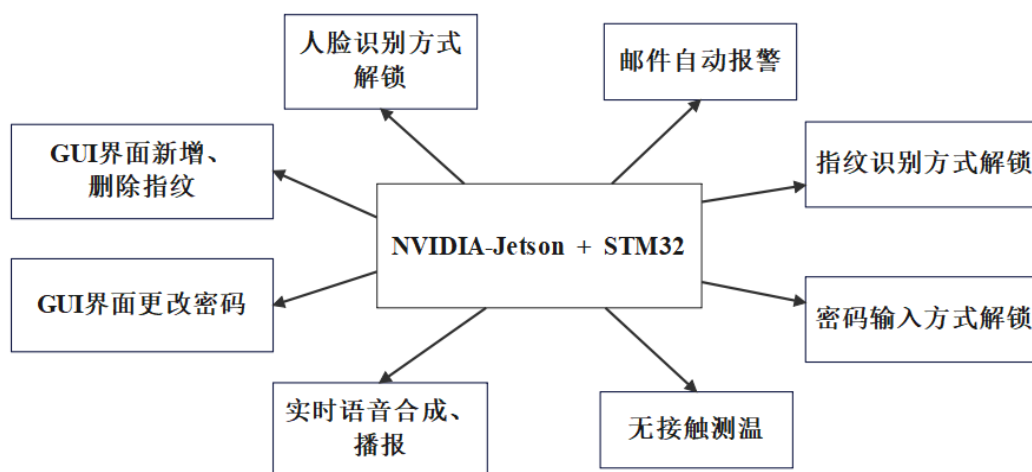


图 2.1 系统功能结构框图

主要实现以下功能：

- （1）智能门锁默认状态人脸识别：系统自动对进入摄像头图像实时采集范围内的人脸进行定位、对比，并用红色的 box 方框框出人脸。
- （2）智能自动邮件报警：对于摄像头图像采集范围内出现的人脸，若识别出人脸为已注册用户，则将会在 box 中显示该用户在系统中的姓名，如“XuZhun”，同时通过 jetson 的串口将该信息发送至 STM32 单片机；若识别出该人脸未注册，将分为以下两种情况：
 - a) 上电开机时间跨度内该人脸首次出现：系统将打上“Unknown”的标签于 box 左下方，并同时截取该帧带有陌生人脸的图像，借助 SMTP 自动发送邮件的协议将该帧图像以附件形式发送至已设定好的邮箱内。该封电子

邮件的主题是“**Intruders**”，邮件正文内容为该陌生人脸在系统中的排序（第几位）及该事件发生的具体时间，如“**No.2-Day07_17-30-11**”，附件中的图片亦以此命名。

- b) 上电开机时间跨度内该人脸再次（多次）出现：仅在人脸的 box 左下方打上“**Unknown**”的标签，不会再有其他响应。
- (3) 手动密码解锁：在无法识别人脸或人脸模块不工作的情况下，可使用密保密码手动解锁门禁。
- (4) 更改密保密码：在密保密码验证成功时可进入更改密码界面进行密保密码的更改。密保密码长度从 1 位到 12 位数字无限制，纯数字密码。
- (5) 验证指纹方式解锁门禁：在系统图形用户界面（**GUI**）内可选择使用验证用户指纹方式完成门禁的解锁。
- (6) 添加用户指纹功能：在密保密码验证正确的情况下（仅此一条件，指纹验证成功后并不可进行新增或删除指纹操作。）可进行用户指纹的增添。新增指纹时需要按两次指纹，系统将对比两次所得的指纹图像 **A** 和指纹图像 **B** 是否吻合，且该新指纹不能与指纹库中指纹相同，若相同则录入失败；若不相同且指纹 **A**、**B** 相吻合，则录入成功且赋予其指纹库编号；否则录入失败。
- (7) 删除用户指纹功能：同上条件下，可进行用户指纹的删除操作。一般，可删除编号范围在 1~12；编号为 0 的为主人指纹，无法在系统中删除，仅可通过 **PC** 端上位机直连串口的方式进行擦除与重新写入。
- (8) 实时语音合成与播报功能：该模块主要配合显示模块实现相关提示语播报提示功能。
- (9) 实时无接触高精速度测量温度功能：在系统 **GUI** 界面中可选择无接触测量温度操作。该功能由无接触红外测温模块实现，和单片机使用 **SMBus**（**System Management Bus**）总线协议进行数据的传输。

2.2 器件选型

2.2.1 主控选型

方案一：采用 Raspberry Pi(树莓派)^[12]做作为主控模块。Raspberry Pi 是系统基于 Linux 的 AI 开发平台，其功能强大，常用于各种嵌入式开发设计。但就图像处理而言，Raspberry Pi 在帧率提升、运算能力等方面稍显不足。

方案二：采用 OpenMV 作为主控模块。OpenMV 摄像头是基于 MicroPython 的开源机器视觉项目，是 32 位的单片机。OpenMV 结合各种类型的感光元件，可以完成多种机器视觉应用，比如物体检测、颜色识别等。但 OpenMV 模块在实时视频流人脸识别方面处理速度过于低下，处理后的视频流卡顿严重，无法满足本设计的硬件需求。

方案三：采用 Maix Bit 开发板作为主控模块。Maix Bit 开发板使用 RISC-V 内核的 AI 芯片 K210，其最大的优点是低功耗，但运算能力并不符合本设计的要求。

方案四：采用 NVIDIA-jetson 系列 AI 开发平台作为主控模块。jetson 是 NVIDIA 嵌入式计算平台的总称，根据算力、产品设计、外接传感器等需求，为各应用场景提供了解决方案。Jetson 系列产品，如图 2.2 所示。

Jetson AGX Orin 系列 高达 275 TOPS 15-60 瓦 100 毫米 x 87 毫米 售价 899 美元起 2022 年 7 月起售 了解详情 >	Jetson Orin NX 系列 高达 100 TOPS 10-25 瓦 70 毫米 x 45 毫米 售价 399 美元起 2022 年 9 月起售 了解详情 >	Jetson Nano 0.5 TFLOPS 5-10 瓦 45 毫米 x 70 毫米 99 美元 了解详情 >
Jetson AGX Xavier 系列 高达 32 TOPS 10-30 瓦 20-40 瓦 100 毫米 x 87 毫米 售价 899 美元起 了解详情 >	Jetson Xavier NX 系列 21 TOPS 10-20 瓦 45 毫米 x 70 毫米 售价 399 美元起 了解详情 >	Jetson TX2 系列 1.3 TFLOPS 7.5-15 瓦 10-20 瓦 尺寸 45 毫米 x 70 毫米起 售价 149 美元起 了解详情 >

图 2.2 NVIDIA-jetson 系列产品

其中，jetson Nano（2GB 版本）搭载四核 Cortex-A57 处理器、128 核 Maxwell GPU，可完美运行多种图像处理训练模型，且功耗较低（5W 模式）。可完美满足本设计的硬件需求。

表 2.1 NVIDIA-jetson Nano 与 Raspberry Pi 性能对比

模型名称	应用场景	架构	Nano	Pi 3	Pi 3 +
ResNet-50(224×224)	分类	TensorFlow	37	1.6	15
MobileNet-v2(300×300)	分类	TensorFlow	65	2.8	28
SSD ResNet-18 (960×544)	物体检测	TensorFlow	5	*	*
SSD ResNet-18 (480×272)	物体检测	TensorFlow	17	*	*
SSD ResNet-18 (300×300)	物体检测	TensorFlow	15	*	*
SSD Mobilenet-V2(960×544)	物体检测	TensorFlow	9	*	1.8
SSD Mobilenet-V2(480×272)	物体检测	TensorFlow	25	*	7
SSD Mobilenet-V2(300×300)	物体检测	TensorFlow	36	0.8	14
Inception V4(299×299)	分类	PyTorch	11	*	*
Tiny YOLO V3(416×416)	物体检测	Darknet	28	0.5	*
OpenPose(256×256)	姿态估计	Caffe	12	*	5
VGG-19 (224×224)	分类	MXNet	13	0.5	6
Super Resolution (481×321)	图像处理	PyTorch	20	*	0.6
Unet(1×512×512)	分割	Caffe	14	*	9

表 2.1 为 NVIDIA 给出了 jetson Nano 与 Raspberry Pi 的各版本在运行各种主流训练模型时的性能对比。其中*表示无法运行，数字的单位是 FPS，即帧率。不难看出，无论是分类识别、物体检测，还是图像处理方面，jetson Nano 的帧率表现均远高于 Raspberry Pi。Nano 支持主流的 TensorFlow、PyTorch、Keras 等 AI 算法和框架，尤其适合于人脸识别开发。

综上所述，本设计选用 Jetson Nano（2GB）作为主控模块，负责人脸识别部分。

2.2.2 辅助控制模块选型

方案一：使用传统 8 位 51 单片机。本文设计的系统功能全面，对串口资源、GPIO（通用输入输出接口）数量及定时器资源提出较高要求。即使增强型 STC12C5A60S2 单片机增加了双串口和 P4 接口，亦无法满足硬件资源需求。故 51 单片机无法满足本设计要求，不采用此方案。

方案二：使用 32 位单片机 STM32（F103）系列。F103 系列单片机具有 5 个双向串口，其中串口 1、2、3 为 USART（通用同步/异步串行接收/发送器）；定时器资源丰富（8 个）；型号 ZET6 更是拥有 144 个引脚。

综上，本设计选择使用 STM32F103 系列单片机最小系统作为系统辅助控制模块，负责锁的开闭控制及扩展功能。

2.2.3 系统输入输出接口选型

方案一：选用传统矩阵按键及 OLED 屏进行输入、输出操作。此方案虽原理简易，但实际编程复杂，需考虑机械防抖及软件防抖；函数封装困难、将会大量占用 GPIO，不利于开发其他功能。因此，舍弃此方案。

方案二：选用带有触摸功能的并口 LCD 液晶显示屏。此类屏幕驱动难度大、大量占用主控内部 FLASH 空间、函数封装困难、大量占用 GPIO；电阻触摸反应迟钝，无法满足本设计中密保密码的硬件设计需求。因此，舍弃此方案。

方案三：选用 TJC^[13]串口屏。串口屏是近年新兴的一种集输入、输出功能为一体的液晶显示屏，使用串口进行通信，单独上位机在线编程、烧录。串口液晶屏较少占用主控内存等资源；电容式触摸灵敏、便捷；指令简洁，可快速开发整洁的图形用户界面。因此，此方案可满足设计要求。

综上分析，选用 4.3 寸串口液晶屏 TJC4827X3_011C 作为本设计的输入输出接口。由于键盘需求，屏幕尺寸必须大于 3.2 寸，综合设计成本及实际硬件功能，选择 4.3 寸较合适。

2.2.4 摄像头选型

方案一：海康威视 USB 摄像头。该类摄像头像素高、成像清晰，但占用系统 USB2.0 接口且驱动困难、功耗较高；运行起来将占用一部分资源，从而拉低视频的帧率。因此舍弃此方案。

方案二：海康^[14]网络摄像头。通过随机 IP 地址将视频流传送至 NVIDIA-jetson，存在被恶意攻击的风险，且对网络连接环境提出较高要求，不满足本设计“识别人脸”的硬件要求。

方案三：CSI^[15]接口摄像头 IMX219。利用 jetson.utils 硬件器件库可直接驱动 15PinCSI 摄像头，较少占用系统内存资源，有助于提高视频流帧率；驱动电流小，功耗低。

由上分析，本设计选用 CSI 摄像头作为实时获取视频流的传感器。IMX219-77 Camera 的感光芯片为 Sony IMX219：焦距 2.96mm、分辨率为 3280×2464，可完全满足画面分辨率为 1024×600 设计硬件需求；对角视角场为 77 度，几乎无画面畸变（<1%），适合用于人脸识别。

2.2.5 视频流输出接口选型

方案一：选用传统 13.5 寸便携显示器（1920×1080）。其接口丰富、尺寸大、分辨率高，一般兼容 Windows 和 Linux 系统，适合开发途中调试使用。但对于成熟产品，屏幕尺寸过大，不便于安装。故，不采用此方案。

方案二：选用 7 寸 HDMI 接口电容触摸液晶屏。该显示屏兼容 Jetson 系列和 Raspberry Pi 系列视频输出，7 寸屏分辨率 1024×600 且为 IPS 材质，可视角度广；电容 10 点触摸，便于调试（可代替鼠标）；USB 5V 供电，电源电路兼容性强，设计难度相对较小；7 寸屏幕大小适中，配合 20PinHDMI 排线适合嵌入产品中；且 jetson Nano（2GB）仅可使用 HDMI 接口进行视频输出，不支出转接其他接口。

因此，综合考虑，采用方案二：选用 7 寸电容屏作为本设计的视频流输出接口。

2.2.6 指纹采集模块选型

方案一：采用电容式 FPC1020A 采集实时指纹。FPC1020A 反应灵敏、指纹存储容量大，但价格昂贵。

方案二：采用光学 AS608^[16]采集实时指纹。AS608 模块通过摄取指纹图片，利用内置指纹算法得到指纹特征。匹配指纹时，模块将所摄取的实时指纹特征与其内部已存入的（多个）指纹模板进行 1:N 对比，给出识别结果（通过或失败）。模块的分辨率 500dpi，录入指纹时间小于 1 秒，但识别时间随指纹库的增大而增加；供电电流小于 60mA，可直接挂载至单片机 3.3V 供电引脚。AS608 模块相较于电容式指纹识别模块，价格更低廉，也更适于本设计。

故本设计采用 AS608 模块进行实时指纹的采集与对比识别。

2.2.7 网络连接方案

方案一：使用 USB 无线网卡方式进行网络连接。由于 jetson Nano(2GB)未配置因特尔板载无线网卡，若需实现无线连接网络则需要配置 USB 网卡方式。此方式便捷，但通常稳定性稍差，可能出现断连等问题。本设计可采用绿联 650M 2.4/5G 双频 USB 无线网卡，其硬件参数如图 2.3 所示。

方案二：使用网线接口连接以太网，此种方式网络连接稳定、网络质量高；但缺点是冗余、仅可在局域网方式下，在某些特定场所无法使用。

综上，用户可根据网络环境自行选择或切换网络连接方式。

品牌: 绿联	型号: CM448	网卡插口: USB
成色: 全新	售后服务: 全国联保	颜色分类: 【小巧款】650M双频高速【...
适用网络类型: 以太网	生产企业: 深圳市绿联科技股份有限公司	传输速度: 其他
是否无线: 是	上市时间: 2021-05-06	

图 2.3 USB 网卡硬件参数

2.2.8 实时语音合成芯片选型

方案一：使用 SYN6288-A 模块实时合成语音。SYN6288-A 使用异步串口通信协议实现文本到语音转换（TTS），可使用常见的 TTL（Time To Live）电平驱动。

方案二：使用 XFS5152CE^[17]实现语音实时合成。科大讯飞 XFS5152CE 芯片（模块）支持 IIC（集成电路总线）、SPI（串行外设协议）、UART 三种通信协议进行数据传输；支持 GBK、ASCII 等多种语音编码方式；支持关键词识别；指令集简洁、合成速度快。

经对比，XFS5152CE 模块使用 USART 方式进行实时语音合成速度更快，发声更流畅、自然。因此，采用方案二。

2.2.9 舵机选型

方案一：选用模拟舵机。模拟舵机相对于数字舵机而言，线性度差、响应速度慢、控制电路较复杂。

方案二：选用 SG90/MG90S 小扭矩航模数字舵机。此舵机使用 5V 供电，反应灵敏，死区小；但扭矩过小，不符合本设计硬件需求。

方案三：选用 17kg 大扭力数字舵机 LD-1501MG。可使用 STM32 单片机的通用定时器 PWM 模式驱动；数字舵机响应灵敏、死区时间小；全金属齿轮，可避免电流过大而烧毁或因负载重量过大而引起齿轮损坏等情况；17kg 扭力可完全满足本系统拉动实际门锁的设计需求。

因此，本设计选用方案三，使用数字舵机 LD-1501MG 完成门锁开闭动作。

2.3 系统逻辑功能部署

确定各部分功能元器件及模块的选型后，即可对其进行逻辑方案部署。本系统的逻辑功能部署结构框图，如图 2.4 所示。

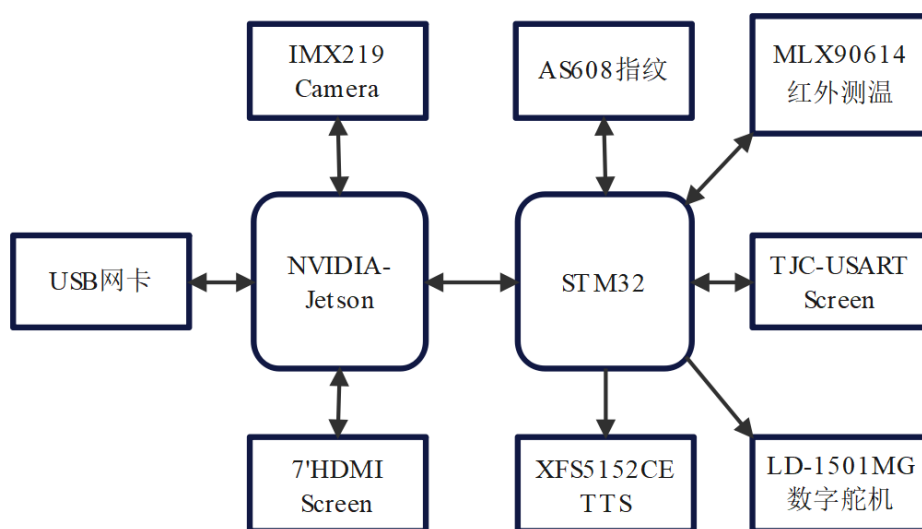


图 2.4 本系统逻辑功能部署

2.4 系统供电电源设计

因所用到的元器件较多，特别是大扭力数字舵机需要较大电流进行驱动，原最小系统板的 USB 电路无法满足本设计的供电需求，故需重新单独设计本系统的供电部分电路。

选用 Fulree 直流转直流（DC to DC）适配器进行电压转换。该 DC to DC 转换器的输出为 5V/3A（max），符合本设计的功率需求；该转换器的输入电压为 9~60V 直流电，可使用常用的 9V、12V 或 24V 电源适配器为其供电。

模块输出使用专用型电源端子线连接至各个功能模块，同时预留出排针及 USB 母座，便于扩展使用；输入使用常用 5.5mm DC 母座焊接；模块设有大功率开关，作为系统的电源总开关。图 2.5 为模块制作实物。

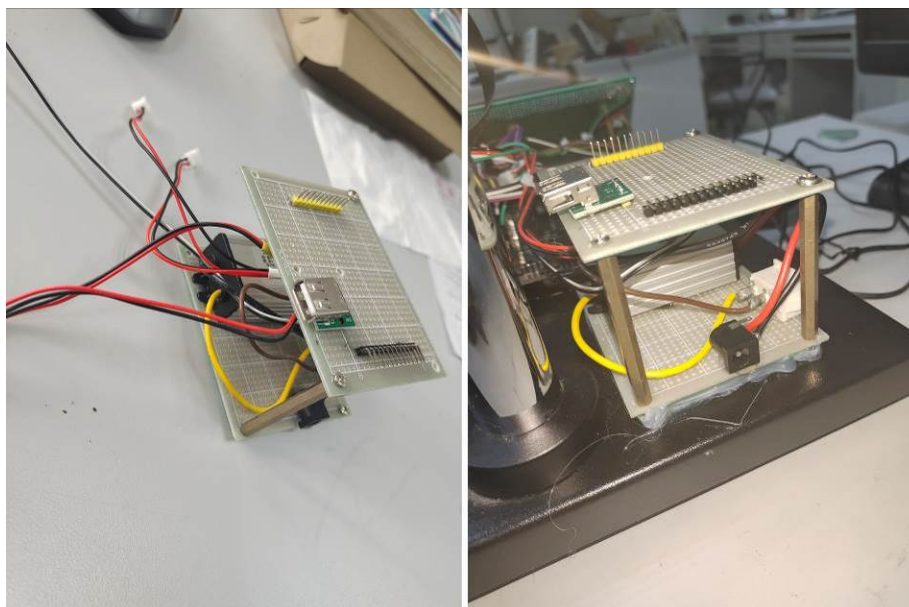


图 2.5 电源模块实物

第三章 系统软件设计

3.1 总体逻辑架构

本系统遵循“模块化编程”思想进行设计与开发。在软件设计上，可将系统开发工作分为对 NVIDIA-jetson Nano（2GB）编程、对 STM32 单片机编程以及对 TJC4827X343 液晶屏编程三个部分，三者之间均使用 UART 方式进行通讯（主控与 TJC4827X343 通过 STM32 单片机间接通信，并未直接通信）。使用一个拨动开关完成选择默认人脸模式或手动模式（指纹和密码方式）操作。

NVIDIA-jetson Nano（2GB）主要承担实时人脸的检测、识别及自动邮件报警等功能。使用 Python 语言进行开发设计。编译器软件为 NVIDIA-jetson 自带^[18]的 vim 编辑器。

STM32 单片机部分主要承担门禁的开闭以及指纹识别及更改、无接触测温等拓展功能。使用软件 MDK5（Keil-v5 MDK 版本）和 C 语言进行开发；交叉使用野火 CMSIS-DAP 仿真器和 ST-Link 仿真器进行仿真调试以及代码烧录。

TJC4827X343 液晶显示屏部分主要承担 GUI 界面的绘制、数字密码的输入与更改以及系统信息的输入输出功能。使用 TJC 公司配套图形化编程软件 USART HMI 进行开发；使用串口下载方式进行固件的刷新。

3.2 jetson Nano（2GB）部分

3.2.1 烧录 JetPack 镜像

在 NVIDIA 官方网站可下载最新版本 JetPack，版本为 4.51，名为 jetson-nano-2gb-jp451-sd-card-image。选用 64G SD 卡作为 Nano 的存储器，烧录 JetPack 系统镜像。在 Windows10 系统环境下，使用 Win32DiskImager 工具对 SD 卡进行写入操作（Write）；将镜像烧录完成的 SD 卡插入 jetson Nano（2GB）卡槽，上电即可。

3.2.2 配置编译环境

安装 Python3.6 版本。

安装输入法。

因 jetson 在运行过程中发热量较大，须安装散热片及散热风扇。采用 PWM 可调速风扇，利用系统时钟 jetson_clock 驱动散热风扇；设置开机自启动，转速 80（最大值 255）。

其中“sudo”命令是以管理员身份运行；PWM 风扇使用 3CM 自攻螺钉固定于散热鳍片处。

安装 Code-Oss Python3.6 编译环境，便于调试。

安装 pip、cmake 等工具。

3.2.3 face_recognition 库环境

开源项目 face_recognition 基于深度学习框架 Dlib^[19]，在 LFW(Labeled Faces in the Wild)准确率为 99.38%。在安装 face_recognition 库环境前，须提前搭建 Dlib 深度学习框架，作为 face_recognition 的依赖库（因 face_recognition 对 Dlib 用 Python 语言进行了二次封装，实为一个人脸识别 API（应用程序接口））。

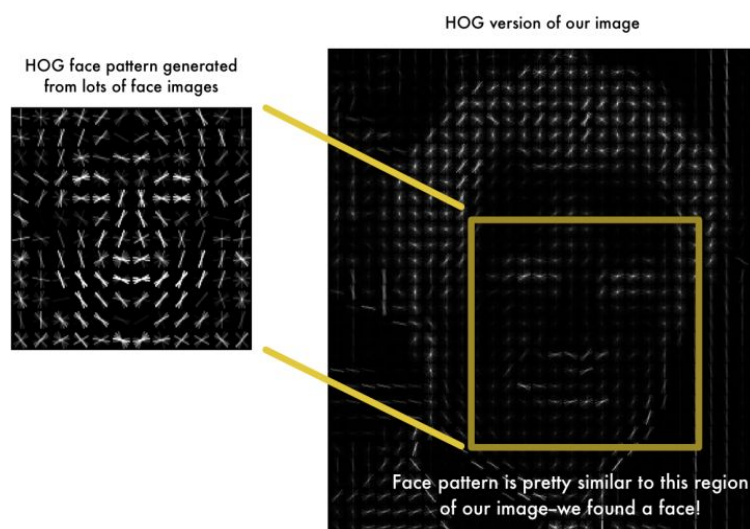


图 3.1 人脸 HOG 图像

人脸检测：即在一幅图片（视频流中抽取一帧）中找出所有面孔。使用函数“face_recognition.face_locations(rgb_img)”即可对图像“rgb_img”中的人脸进行

检测。原理是利用 HOG^[20]（方向梯度直方图）。即将图像灰度化（色彩对定位人脸无贡献）后计算各像素的梯度，得到 HOG 格式图像（图 3.1），从而提取图像特征。

利用 OpenCV 操作 “cv2.rectangle” 将所有人脸框出，实际效果如图 3.2 所示，可准确检测出画面中的有面孔。



图 3.2 人脸检测实际效果

人脸配准：识别面孔中的关键点，实现 68 个特征点定位。这 68 个特征点分布于面部中的鼻梁、左右眼及嘴唇等部位，如图 3.3。

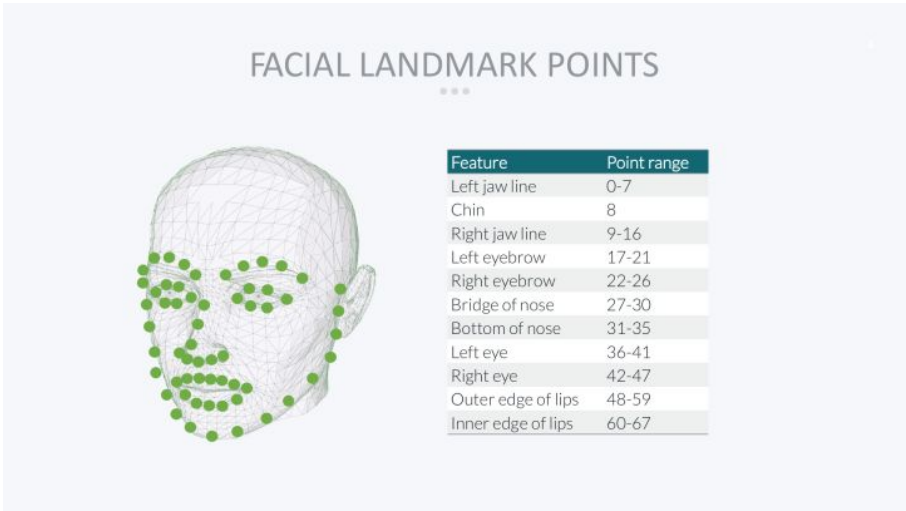


图 3.3 面部 68 个特征点分布

面部编码：根据人脸配准得到的特征点计算该面孔的 128 维特征向量。“face_encodings”函数即执行编码操作，将输入的含有人脸的图片（仅含唯一面孔）输出为 128 维特征值。

对比匹配：人脸库中存入每张面孔的 128 维^[21]编码值。将实时检测到的人脸与之比对。若所得欧氏距离在可接受范围内，则输出距离最小的那一面孔，识别成功；若该距离不在可接受范围内，则识别失败。

人脸对比函数“compare”用于计算两个面孔 128 维特征值的距离（范围在[0,1]区间内，距离越小表示同一张人脸的概率越高；反之越低），距离阈值“tolerance”，即“容忍度”默认为 0.6，计算所得实时欧氏距离若小于该值，则表示是同一副面孔，对比成功；反之对比失败，非同一副面孔。需在系统特定工作条件下反复对 tolerance 的值进行试验、修改，以得最佳识别效果。

Jetson-inference 库是 NVIDIA TensorRT^[22]将神经网络部署于 NVIDIA-jetson 平台，强化了 Jetson 图像处理的速度。使用 Python 语言进行开发工作时，更需借助“jetson_utils”器件库进行 CSI 摄像头的直接调用、访问。

部署 face_recognition 和 jetson-inference 环境，需从 Github 网站进行下载、安装、编译、运行。

3.2.4 邮件报警

基于 SMTP（简单邮件传输协议）^[23]实现自动邮件报警的工作主要过程。



图 3.4 自动邮件报警测试效果

在系统实时画面摄取范围内，若所摄取的人脸处于未注册状态（即“陌生人”），则系统将截取该帧图像，并以邮件附件的形式发送至用户指定的邮箱，图像为 JPG 格式，可下载。邮件主题为“**Intruders**”；邮件正文为该陌生人在一定时间范围内（系统在线时间段）出现在画面中的次序及图像摄取时间，如“No.2-Day07_17-30-11”。

实际效果如图 3.4 所示。

3.2.5 建立人脸库

在往复地执行检测、识别动作前，需要先建立一个人脸识别库，以供本系统将实时检测出的面孔与之比对。

此 Python 程序命名为“facecollection.py”，包含“face(path)”和“save_known_faces()”两个函数。

face 函数读出系统路径 path 中的图片，利用 face_recognition 中的定位人脸函数，若一张图片中的面孔数目大于 1 则向终端打印提示信息“Multiple faces in one picture!”，程序停止，提示用户更换图片进行注册；符合条件后，用 encodings 函数得到人脸的 128 维特征值。

save_known_faces 函数利用 pickle 工具将该 128 维特征值序列化，存入“FaceSet.dat”二进制可读文件中，供后续程序进行读取。



```
xu@xu-desktop: ~/4.21
文件(F) 编辑(E) 标签(T) 帮助(H)
xu@xu-desktop:~$ cd 4.21
xu@xu-desktop:~/4.21$ python3 collection.py
We find 1 face(s)
length of name: 0
length of encodings: 0
XuZhun
['XuZhun']
xu@xu-desktop:~/4.21$
```

图 3.5 人脸库编译信息输出

facecollection.py 运行后将在 Linux 终端中打印所有注册者的姓名（注册者所在图片的命名，英文）及人脸库的大小，如图 3.5 所示。

3.2.6 调用 CSI 摄像头获取实时画面

通过 Sony IMX219 获取实时图像画面（视频流）。NVIDIA-jetson 平台原生支持 IMX219 驱动，即无需额外安装驱动程序。使用异向 15 线 FFC 排线将 IMX219 与 Jetson Nano (2GB) 进行连接（注：必须在断电状态下进行此操作，否则可能导致损坏主板）；上电开机后在 Linux 终端输入“ls /dev/vid*”进行查看 dev 设备是否就绪，若终端打印“/dev/video0”字样则表示 IMX219 已就绪。



图 3.6 CSI 实时视频流

可借助 OpenCV 操作配合 jetson.utils 进行调用摄像头。预先设置 gstreamer 管道参数，即实时捕获图像的长度、宽度，图像显示的长度、宽度，帧率等信息；将管道与视频流绑定；最后使用 cv2.imshow 函数进行图像逐帧显示，形成实时视频流。

实际效果见图 3.6，视频的分辨率为 1280×720。

3.2.7 实时识别人脸

系统实时运行的检测、识别人脸等过程由 Alert.py 完成。Alert.py 由 SendEmail、load_known_faces、lookup_known_face、lookup_unknown_face 及 Record_live 五个函数组成，其功能分别是发送邮件、载入人脸库、查询是否已注册人脸、查询镜头中人脸是否第一次出现，及实时记录。Record_live 为主函数，运行过程中对其余 4 个函数进行调用。

SendEmail 函数用于配置自动发送邮件功能，基于 yagmail 工具实现。yagmail 是 Python 的基于 SMTP 客户端的第三方封装库，从而可以免去繁杂的底层协议操作。发送方选择为网易邮箱，其服务器选择为“smtp.163.com”，发送用户为

x18205010212@163.com，需在网易客户端对该账户进行授权，获取随机授权码；主人邮箱（接收邮件账户）选择 QQ 邮箱：984623208@qq.com。

`load_known_face` 函数用于载入前述 `Fcecollection.py` 中生成的 `FaceSet.dat` 可读二进制文件。建立全局变量 `known_face_encodings` 和 `known_face_name`，用于存放读出的 128 维特征值及相应的用户姓名（图片名）

`lookup_known_face` 函数用于在人脸库中搜寻是否有与镜头中实时摄取到的面孔相似（小于一定阈值）的人脸。若是，则更新 `name` 至 `known_face_name` 全局变量中。

`lookup_unknown_face` 函数用于记录镜头中实时摄取的人脸是否为首次出现，以防止重复发送邮件造成垃圾信息。若经上述 `lookup_known_face` 函数判断为未注册人脸，则此函数记录、保存此实时人脸并与已存储的未注册人脸进行比对。返回 `flag` 变量的值，即 `True` 或 `False`，由主函数进行下一步操作。

`Record_live` 函数用于反复运行。配置摄取、显示参数（长、宽等）后使用 Jetson 原生器件库 `jetson.utils` 调用 `IMX219` 以获取实时画面。待管道 `jetson.utils.gstCamera` 调用就绪后，返回状态值 `True`，进入循环。在实时视频流中抓取一帧（当前帧），从 `CUDA` 操作转为 `Numpy`（作用是计算与对比人脸 128 维特征值的欧式距离，从而判断其是否小于距离阈值），从而进行后续的 `OpenCV` 操作。

由于检测、识别人脸计算量巨大，若对视频流的每一帧均进行识别，则会导致输出视频流过于卡顿、实时性变差，从而导致系统识别效率下降、误识率增加。因此，设置一间隔循环执行操作，每 60 帧执行 2 次识别操作，以提升系统识别的效率和输出视频流的流畅度、方便用户在适宜时间内摆正面孔。仅在“识别帧”进行后续一系列运算操作；在“非识别帧”，仅使用 `jetson.utils` 调用 `IMX219` 摄像头后转为 `OpenCV` 输出实时视频流。

使用 `cv2.resize` 函数对当前帧缩放至原图 1/4 倍尺寸，以便于进行检测。随后对该图像进行通道转换（`rgb_small_frame`），方便进行 `OpenCV` 操作。

由上述 `face_locations` 函数与 `face_encodings` 函数作用于 `rgb_small_frame` 即可得到当前图像中人脸的位置及 128 维特征值。此时调用前述 `lookup_known_face` 对当前实时图像进行人脸对比，判断图像中是否存在已注册人脸（允许实时帧中出现多个人脸）：若存在，则将该面孔所在的姓名更新至 `name` 变量，同时更新至 `face_label`（用

于后续框出人脸)、使用 UART 发送相关信息至 STM32 单片机执行解锁和相关信息提示等指令、在 Linux 终端打印出匹配来访者的姓名;若不存在,则将 face_label 赋“Unknown”,同时调用 lookup_unknown_face 函数,判断该面孔是否为第一次出现。若该陌生来访人脸非首次出现,则发送相关信息至 STM32 单片机进行信息提示,不再进行其他操作;若该陌生人脸首次被捕获,则存储该来访者面孔至本地编码、保存该帧图像,调用 SendEmail 函数进行发送邮件方式向主人用户报警,同时发送相关信息至 STM32 单片机进行信息提示。

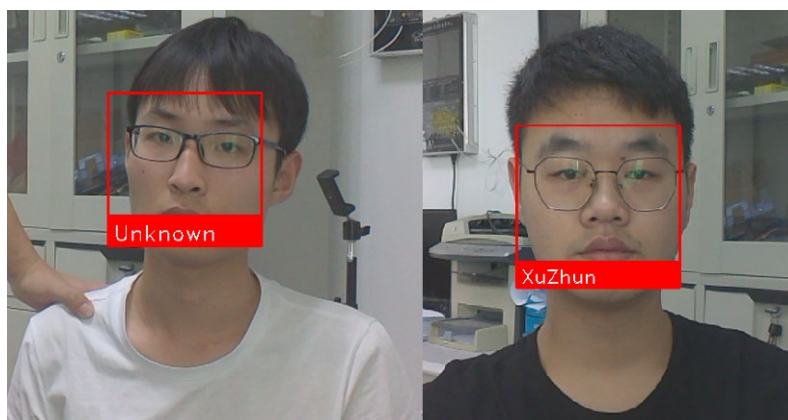


图 3.7 系统输出视频流实际效果

在系统检测到的人脸处生成提示框,用以表示系统检测出人脸及其位置。配置所选图像、标签内容(具体的人名,或“Unknown”)、字符串左上角的坐标,标签字体、字号大小、字体颜色、字体宽度等信息。本系统配置为红色等边长矩形框、标签内容为变量“face_label”、坐标可变,在所识别出的人脸处(left + 6, bottom - 6),字号为 cv2.FONT_HERSHEY_DUPLEX、大小为 0.8、颜色白色(即 R、G、B 值均为最大值 255)、宽度为 1 单位。实际效果如图 3.7,左侧为未注册面孔,标签为“Unknown”;右侧为已注册面孔,标签为已注册用户姓名,如“XuZhun”。

若实时帧图像中包含多个面孔,则按照检测出的人脸顺序依次执行识别、生成提示框等操作。由 OpenCV 工具 cv2.imshow 函数对以上处理过的图像进行逐帧显示,生成输出视频流(图 3.7)。

本文设计了人为关闭 gstreamer 管道流水线命令,调试过程中点击 Q 键即可跳出循环,关闭视频流输出窗口、停止流水线。

图 3.8 为 Record_live 函数的运行逻辑流程图。

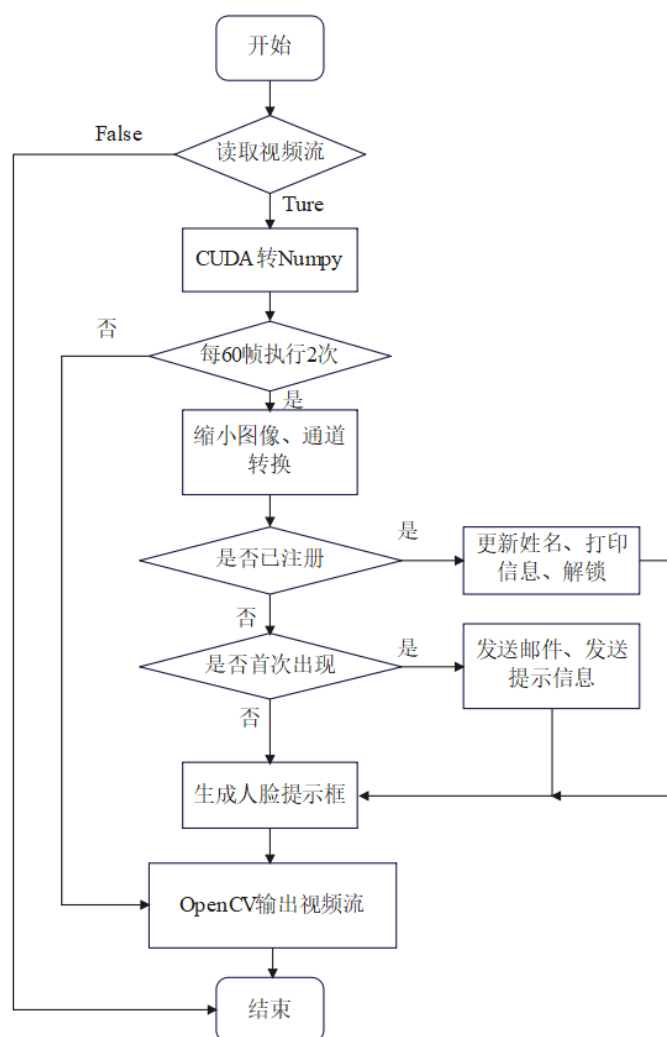


图 3.8 Record_live 函数逻辑流程图

3.3 STM32 单片机部分

3.3.1 AS608 程序设计

AS608 光学指纹识别模块的标准工作主频为 108MHz；内核电压为 $1.8V \pm 5\%$ ，但 I/O 电压在 $3.3V \pm 10\%$ ，即与 STM32 单片机 GPIO 电平一致，可直接进行通信。本系统使用 AS608 SOC 的 UART 工作方式，占用 STM32 的串口 2。AS608 模块 UART 方式缺省波特率为 57.6Kbps，可通过指令更改波特率。

AS608 光学指纹识别模块引脚明细及 PCB 如图 3.9 所示，模块具有手指探测功能：在 Vt 脚有 3.3v 供电的前提下按压手指，WAK 状态输出引脚将输出 3.3V 高电平

（无手指按压时默认为低电平）。因此 WAK 常接 LED 或由单片机输入捕获以指示模块工作状态。

序号	名称	说明
1	Vi	模块电源正输入端。
2	Tx	串行数据输出。TTL 逻辑电平
3	Rx	串行数据输入。TTL 逻辑电平
4	GND	信号地。内部与电源地连接
5	WAK	感应信号输出，默认高电平有效
6	Vt	触摸感应电源输入端，3.3v 供电
7	U+	USB D+
8	U-	USB D-

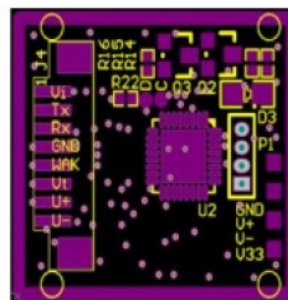


图 3.9 AS608 模块引脚及 PCB 示意图

bsp_as608.c 及其相应头文件 bsp_as608.h 中存放 STM32 与 AS608 模块通信的底层函数，均为地址操作。

PS_GetImage(void)函数用于捕获按下的指纹图像。

PS_Connect()函数用于在系统上电开机时检查 STM32 与 AS608 模块的通信连接是否成功，唯通信无误方可正常运行。该函数的原理是 STM32 单片机使用串口发送固定格式的模块地址至 AS608，若模块返回的数据是开发手册中所规定（*(rx_data->head) == 0XEF）即通信成功，系统继续运行；否则报告错误，提示检查电路连接。

PS_StoreChar()用于下达存储指纹指令至模块。PS_HighSpeedSearch()函数用于进行指纹实时对比时高速搜索指纹库，与识别人脸的对比过程相似。

PS_DeletChar()删除 flash 数据库中指定 ID 的指纹图像。

as608_test.c 及其对应头文件 as608_test.h 则用于存放用户功能类函数，对前述的底层函数进行调用及封装，在运行 STM32 主程序时可直接调用，从而完成相应的操作。

新增指纹函数 Add_FR(void)用于用户新增一个指纹。使用 PS_GetImage()捕获指纹图像，再由 PS_GenChar(CHAR_BUFFER1)生成特征，若此时返回值为 0x00 则首次捕获成功；否则该实时捕获的指纹图像已经存在于指纹库中，跳出函数。同样，再使用 PS_GetImage()再次捕获指纹图像，PS_Match 函数进行匹配若返回 0x00 则指纹 A、B 吻合，录入成功；否则失败。需给录入的新指纹赋予 ID 方可存入指纹库，进

行对比。采用自动赋予的方式，在上一个 ID 的基础上加 1 即可。Add_FR(void)新增指纹功能函数的逻辑流程图如图 3.10 所示。

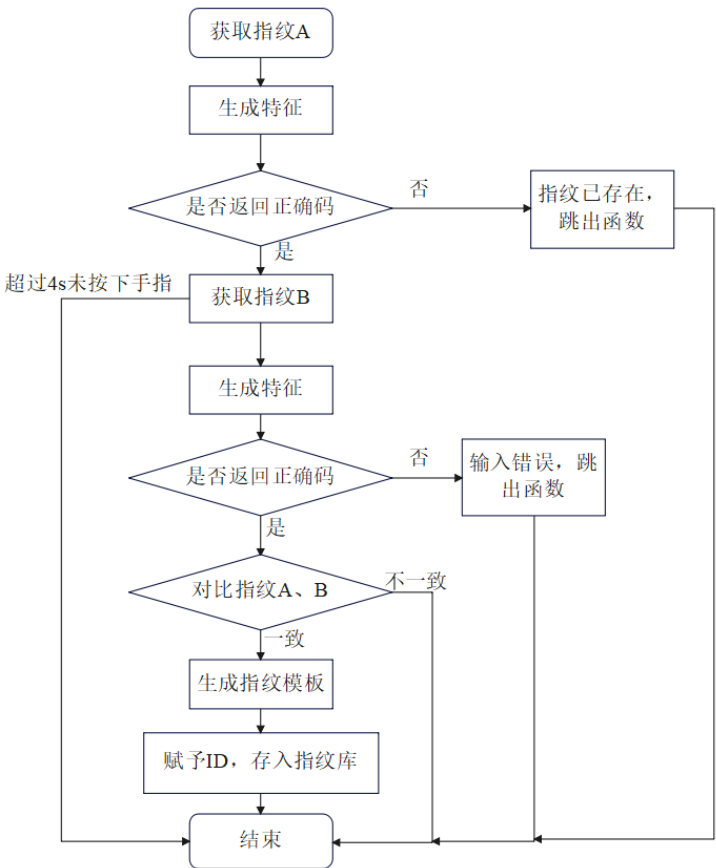


图 3.10 Add_FR 函数逻辑流程图

删除指纹功能函数 Del_FR(void)用于在指纹库中删除特定 ID 的指纹模板，其功能逻辑流程图如图 3.11 所示。

单片机使用 C 语言重定向函数“getchar()”从 TJC 串口液晶屏处获取用户输入的十六进制 ID 码（0X31、0X32、0X33、.....0X3C），存放于“id”变量中；使用 switch 语句对变量“id”进行转换，删除相应的指纹图像模板。

图 3.12 是对比指纹函数 Compare_FR(void)的逻辑流程图，用于将实时录入捕获的指纹图像与指纹库中已注册指纹进行逐一对比，亦即用于指纹识别方式解锁门禁。

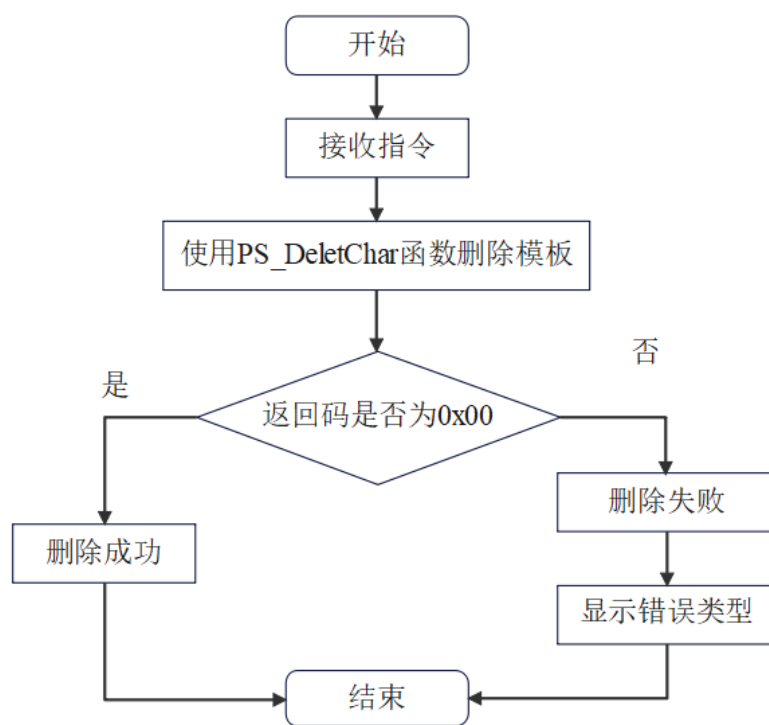


图 3.11 Del_FR 函数逻辑流程图

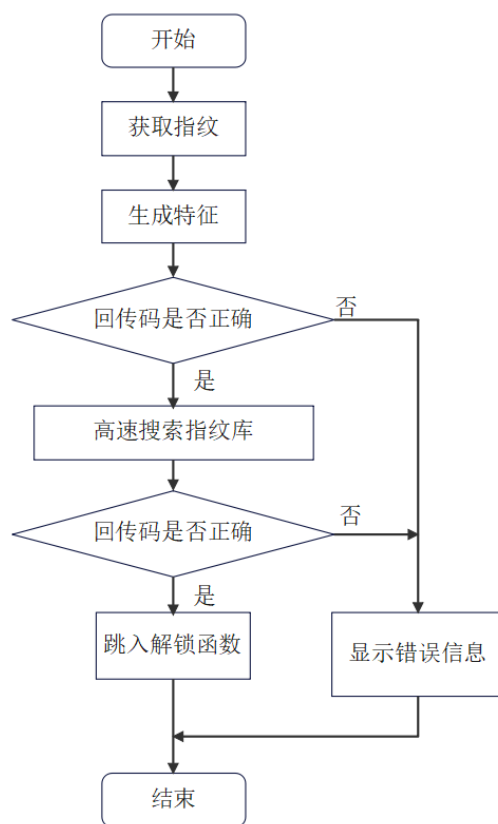


图 3.12 Compare_FR 函数逻辑流程图

3.3.2 MLX90614 程序设计

MLX90614以81101^[24]热电元件作为温度传感器。模块出厂时已经进行过线性化处理，无需在软件上进行二次校正处理；非接触、小体积、高精度，适合本系统应用场景。

SMBus是由intel公司提出的一种同步串行总线:数据线和时钟线。由于STM32单片机没有配备SMBus串行总线接口，使用GPIO软件模拟方式读取MLX90614模块数据。图3.13为模块的原理图。

GPIOC_GPIO_Pin_1用于模拟串行时钟线SMBUS_SCK；GPIOC_GPIO_Pin_2用于模拟数据线SMBUS_SDA。MLX90614的驱动程序存放于板级支持程序bsp_906.c及其相应头文件bsp_906.c中。

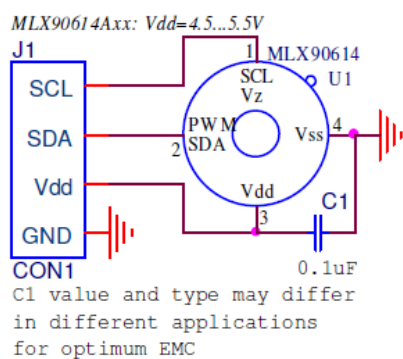


图3.13 MLX90614模块原理图

SMBus_StartBit()、SMBus_StopBit()、SMBus_SendByte()、SMBus_SendBit()、SMBus_ReceiveBit()、SMBus_ReceiveByte()、SMBus_Delay()函数分别用于产生起始位、产生停止位、发送一个字节数据、发送一个比特数据、接收一个比特数据、接收一个字节数据以及SMBus总线专用延时函数，与模拟IIC总线协议相似。

模块初始化函数SMBus_Init()用于系统上电开机时对MLX90614进行初始化，使总线开始工作。打开GPIOC时钟，配置SMBUS_SCK、SMBUS_SDA为集电极开漏输出模式（因I/O用于数据通信，无需较大的灌、拉电流），拉高时钟线与数据线，等待单片机发送起始位信号进行通信。

SMBus_ReadMemory()函数从模块的EEPROM中读取温度。SMBus_ReadTemperature()函数用于转换出温度值（摄氏温标），以浮点数形式返回给主函数。配置一个浮点型

变量Temperature, 使用“Temperature = SMBus_ReadTemp();”语句即可读出当前实时摄氏温度值。因用于测量体温, 使用“%.2f”保留2位小数。

3.3.3 XCFS5152CE 程序设计

XFS5152CE能实现中、英文语音文本的实时快速合成, 合成的人声自然、流畅。该芯片采用LQFP64封装, 64个引脚、芯片规格为10mm×10mm×1.4mm, 兼容3.3V和5V电平。支持UART、IIC、SPI三种通信协议, 本设计采用UART方式通信, 占用STM32单片机的USART3。芯片支持四种最常用波特率, 采用拨码开关方式实现通讯波特率的切换与更改, 本设计采用9600bps波特率与该模块进行通信。

为方便模块化编程、减小冗余, 将XFS5152CE模块、5V音频功率放大器及扬声器集成于一片万用电路板上, 进行二次封装, 如图3.14所示。

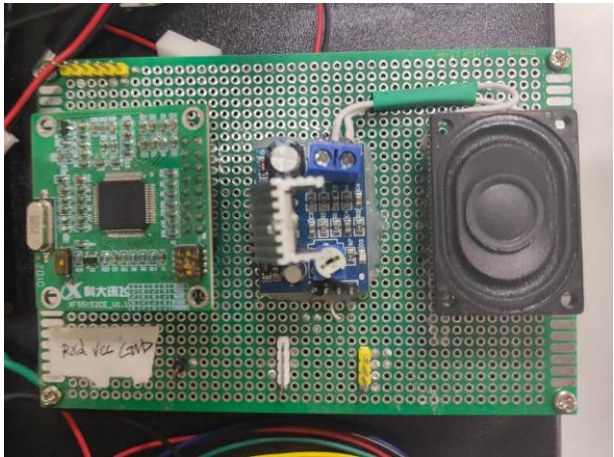


图3.14 语音合成二次封装模块

模块支持多种控制命令, 如语音合成、文本缓存、语音编码解码、语音识别等。该模块在本设计中仅用于语音合成功能, 因此只涉及到语音合成相关命令, 如表3.1所列。

表3.1 XFS5152CE芯片支持的合成相关命令

名称	发送的数据	说明
命令	0X01	语音合成
	0X02	停止合成
	0X03	暂停合成
	0X04	恢复合成

在STM32单片机控制本模块进行语音合成时，STM32发送给XFS5152CE芯片的所有指令均需以固定格式封装后传输。帧头构造为0XFD。单片机使用命令字来向XFS5152CE下达指令，以16进制字符形式进行交互，详见表3.1。

芯片的TXD（数据发送脚）引脚接至GPIOB_Pin_11；芯片的RXD（数据接收脚）引脚接至GPIOB_Pin_10。STM32的GPIOB的第10、11引脚复用为USART3的TXD、RXD功能，无法更改引脚。STM32的GPIOB挂载于内核的APB2总线上，时钟为72MHz；而片上外设USART3则挂载于APB1总线，时钟为36MHz。因此，驱动XFS5152CE需要同时使能APB1与APB2的RCC时钟。

单片机的GPIOB_Pin_10配置为复用、推挽输出模式，时钟频率50MHz；GPIOB_Pin_11配置为复用、浮空输入模式，时钟频率50MHz；串口3波特率初始化为9600bps；8位数据、1位停止位、无校验位、收发一起。

XFS5152CE模块专用串口发送字符串数据函数PrintCom()用于向芯片发送字符串数据，构成一个API，便于下述文本合成函数调用。

依据表3.1，构造相关芯片设置命令如下：

- XFS_StopCom[]={0xFD,0X00,0X01,0X02};//停止合成
- XFS_SuspendCom[]={0xFD,0X00,0X01,0X03};//暂停合成

模块文本实时合成函数XFS_FrameInfo(char *HZdata)用于合成文本、发送至XFS5152CE模块；该函数调用格式如

XFS_FrameInfo("安徽建筑大学");

其中函数内指针变量*HZdata为需要合成的语音文本，无需等待模块回传确认码。函数原理为测量文本字符长度HZ_Length后构造帧：帧头0XFD、数据区长度高字节0X00、低字节HZ_Length+2、命令字0X01（合成播放命令）、编码格式命令字0X01（及GBK格式编码），用前述UART3模块发送函数PrintCom()将此帧命令下达至语音合成模块；模块解码、播放，完成一次语音的合成、播放。

3.3.4 数字舵机模块程序设计

LD-1501MG 数字舵机使用 PWM^[25]脉宽调制驱动。周期为 20ms，其中高电平持续时间 0.5ms~2.5ms 的脉宽电平对应舵机打角 0 至 180 度。占空比与舵机角度呈线性关系。

STM32F103 单片机的定时器资源丰富，共有 8 个定时器，且每个定时器均配备 4 个通道（CH1~CH4）；高级定时器挂载于 APB2 总线，基本定时器和通用定时器挂载于 APB1 总线。因基本定时器无脉冲宽度调制功能，故本设计采用通用定时器的 PWM 输出模式驱动数字舵机。

TIM1、TIM2 的 4 个通道大部分复用于 GPIOA，与本系统的其他功能模块（指纹、语音合成）引脚冲突，因此本设计选择采用 TIM4_CH1 通道，即 GPIOB_Pin6 引脚进行驱动。

定时器 4 的 PWM 模式配置函数 TIM4_PWM_Init 用于初始化配置 TIM4。GPIOB 挂载于 APB2 总线、TIM4 挂载于 APB1 总线，则应同时使能 APB1 和 APB2 时钟，如图 3.15，同时使能 APB2 外设复用功能。

```
RCC_APB1PeriphClockCmd(RCC_APB1Periph_TIM4, ENABLE);  
RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOB | RCC_APB2Periph_AFIO, ENABLE);
```

图 3.15 使能外设操作

GPIOB_Pin_6 配置为复用推挽输出模式（GPIO_Mode_AF_PP），时钟 50MHz。配置 TIM4_CH1 为 PWM 输出模式（TIM_OCMode_PWM1）；使能比较输出功能、配置 TIM4 输出比较极性，即 TIM_OutputState_Enable、TIM_OCPolarity_High；最后使能 TIM4 的总开关（TIM_Cmd(TIM4, ENABLE);）。

在主程序中对舵机初始化为

```
TIM4_PWM_Init(200-1,7200-1);
```

即周期计算为 $(7200*200)/72000000=0.02s=20ms$ 。驱动舵机打角命令为

```
TIM_SetCompare1(TIM4,5);
```

即 TIM4 通道 1 输出 0.5ms 高电平，对应数字舵机 0 度；同理，15 对应于 1.5ms 高电平时间，对应数字舵机 90 度。

3.3.5 STM32 主程序设计

Screen()函数用于查询 TJC 串口屏和 NVIDI-Jetson 发送至 STM32 单片机的指令，对不同的指令做出相应的动作。该函数的逻辑流程图，如图 3.16 所示。

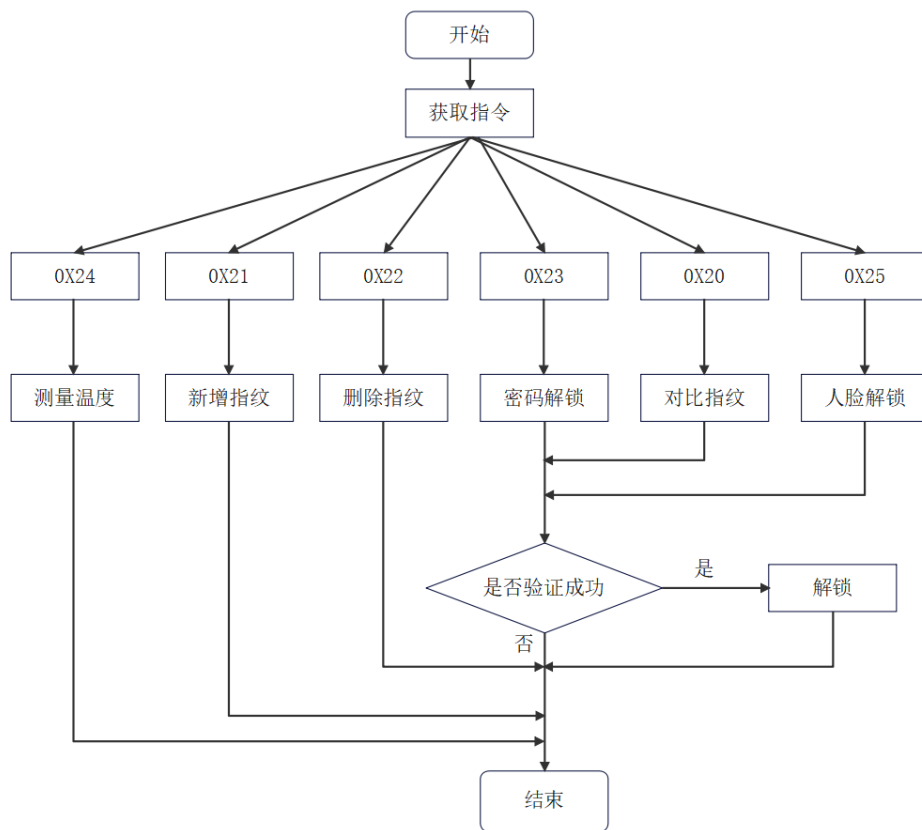


图3.16 Screen()函数流程图

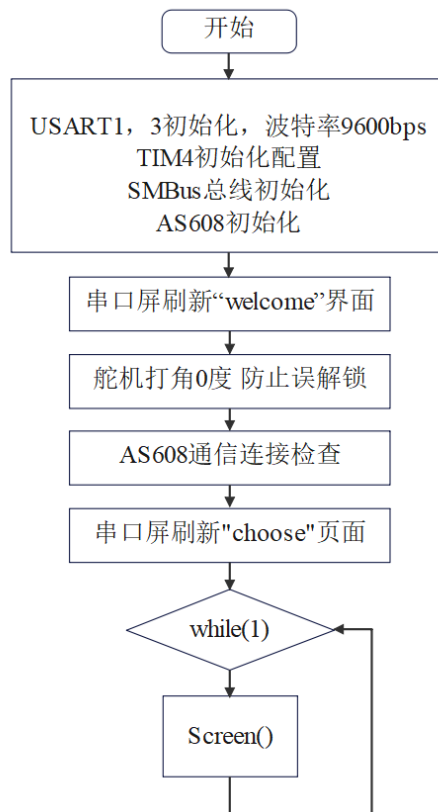


图3.17 main()函数逻辑流程图

在main函数中，首先将各个模块的功能进行初始化；舵机打角0度，以防止误解锁；串口屏上电刷新“welcome”界面，介绍本毕业设计相关的信息及，即

```
printf("page welcome\xff\xff\xff");
```

其中\xff\xff\xff是屏幕的结束符。若干时间后，系统启动完毕，进入选择操作界面“choose”。main函数while(1)大循环中的操作均封装于Screen()函数中，以等待用户进行相关操作。图3.17为Main函数的逻辑流程图。

3.4 TJC4827X343 液晶屏部分

陶晶驰 4.3 寸电容触摸屏幕 TJC4827X343_011C 在本系统中起键盘输入及信息输出接口的作用。在 USART HMI 开发软件中选择相应型号，进行图形化编程；工程可通过串口进行实时模拟器同步仿真，开发完成后使用串口下载方式进行固件烧写（固件刷新）。烧写完成后液晶屏即可独立工作，使用该显示屏配合 STM32 单片机完成相关功能的实现。

本设计中最常用到的控件是文本、按钮。系统中交叉使用 STM32 控制和串口屏内部语句控制两种方式进行相关文本信息的显示、切换页面等。为防止误码干扰，STM32 的 MCU 向 TJC4827X343 以串口通信方式发送相关指令时必须在指令结尾添加三个 0XFF 结束符。

TJC4827X343 上电默认刷新第 0 页，即“welcome”page，如图 3.18 所示。该页面内含有 4 个文本控件、1 个图片控件，用于显示本次毕业设计的选题、学生姓名、指导老师姓名及 STM32 端的实时信息（如 AS608 模块的连接状态等）



图 3.18 “welcome”界面

“choose”页面是系统的“主界面”，默认停留于此页面。默认状态下使用人脸识别方式进行解锁，文本控件 t0 用于提示相关识别结果信息，如“验证人脸请正视摄像头”；按钮控件 b1、b2、b3 分别对应于验证指纹、验证密码、测量体温功能，在按

钮控件的“弹起事件”中是按钮弹起时执行的指令，即发送相应的 16 进制字符至单片机。如“printh 20”表示按钮弹起时发送 16 进制字符 0X20 至 STM32 单片机。

同理，“keyboard”界面用于输入密码、更改密码。按钮控件为数字 0~9，以及“DEL”、“OK”、“BACK”按键；文本控件 t0 用于显示输入的密码。以星号（*）形式显示。更改密码操作可在 TJC4827X343 内部完成，无需借助 STM32。定义一个全局变量 va0，在更改密码时将其写入 TJC4827X343 的内部固定序号的 EEPROM 中，序号为 0。

“right”页面用以提示用户以密码方式解锁成功，t0 文本控件用于显示 STM32 单片机发送的用户具体解锁方式，如“您使用密码验证通过！”。同时在该界面会显示“修改密码”、“新增指纹”、“删除指纹”三个功能按钮，即在密保密码验证成功的前提下方可进行以上操作。“right”界面设置了一个定时器 tm0，定时 3000ms，即 3 秒钟。页面每 3 秒时间查询一次全局变量 va0 的状态，若为 0（即 3 秒内用户无任何操作）自动跳转至“choose”选择主页面，回到锁定状态。该页面刷新时会发送 16 进制字符 0X23 至单片机，表示用户使用密码方式成功解锁门禁，由 STM32 单片机完成后续信息处理。

“false”页面是密码输入错误时跳转，仅含一个文本控件用以提示用户重新输入密码。本页面设置了一个软件延时，1500ms 后自动跳转至“choose”页面，回归锁定状态。

“modify”页面是在“right”界面按下修改密码时所跳转的页面，用于将新的密码确认后写入 TJC4827X343 模块的 EEPROM 中。该页面同样设置了一个倒计时定时器 tm0（专有变量），即使时间为 4500ms。定时倒计时期间若用户无操作，系统将回到锁定状态；亦可直接点击“退出”按钮控件直接退回“choose”界面，锁定。

“fit”界面用以提示指纹相关的信息，如“验证指纹，请按手指”、“指纹对比失败，请重试！”等。t0、t1 文本框的内容由单片机直接确定。

“k2”界面是删除指定指纹 ID 界面，按钮弹起事件中发送相应的 16 进制码至单片机。

“temp”界面是用户按下“测量体温”按钮时所跳转的界面。t0 文本框用以提示“请靠近测温传感器”及显示实时温度值（摄氏温标）。测量结束时由 STM32 发送跳转页面指令至“choose”。

第四章 实验结果与分析

4.1 装配系统

图 4.1 为基于 NVIDIA-jetson 的智能门禁系统实物图。除 Jetson Nano 及其 HDMI 显示器外的所有元器件均由电源模块进行供电。各个模块进行通信及数据交换时应注意共地（GND）以保证统一零标准电平^[26]，否则将导致系统无法正常工作。系统中拨动开关（位于 TJC4827X343 左侧）用于切换默认解锁方式为人脸解锁/手动解锁（密码或指纹方式）。

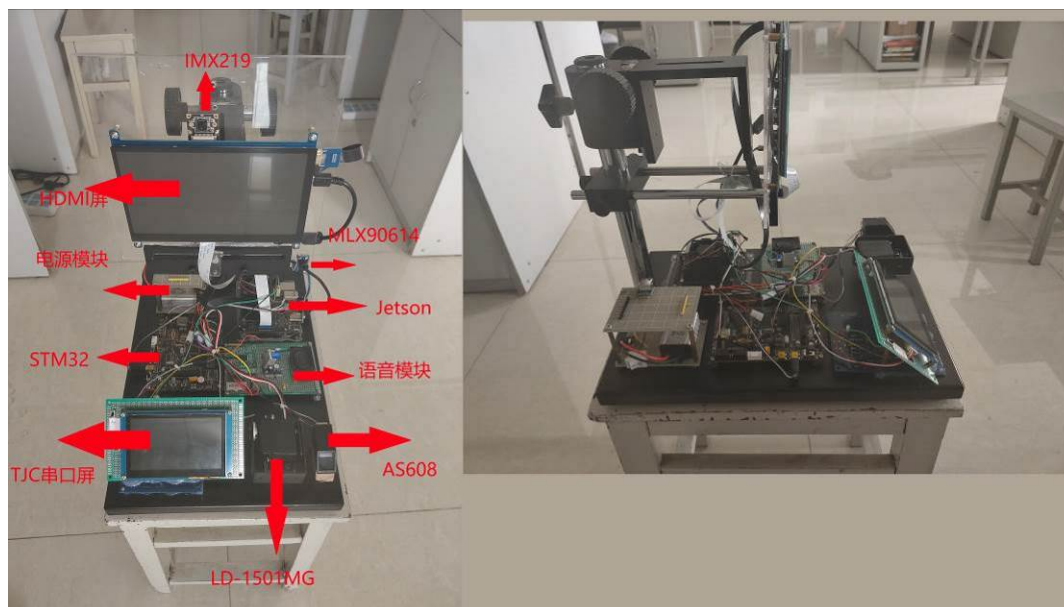


图 4.1 基于 NVIDIA-jetson 的智能门禁系统

4.2 jetson 电源配置测试

NVIDIA-jetson Nano（2GB）的技术规格如表 4.1 所列。则开发调试过程中需配备 USB 接口键盘、鼠标、INMX219 摄像头、无线或有限局域网及 HDMI 接口显示器；在 5W 功率模式中视频流帧率较低，为保证良好的性能释放，配备 Type-C 接口 5V/3A 电源适配器。

表 4.1 NVIDIA-jetson Nano (2GB) 技术规格

组件	描述
GPU	Maxwell 架构, 128 CUDA 核心
CPU	四核 ARM®Cortex®-A57 MPCore 处理器
内存	2 GB 64 位 LPDDR4
存储	Micro SD 卡槽 (32GB 以上)
视频编码	4K @ 30 (H.264/H.265)
视频解码	4K @ 30 (H.264/H.265)
摄像头	12 通道 MIPI
连接	千兆以太网
显示器	HDMI 2.0
I/O	1x SDIO/2x SPI/4x I2C/2x I2S/GPIO

采用 5V/3A Jetson 电源适配器对 NVIDIA-jetson Nano (2GB) 进行测试, 功耗模式选择为最大性能模式。表 4.2 是 NVIDIA-jetson Nano (2GB) 最大性能模式下、5V/3A 电源的相关数据。因低功耗模式 (5W 模式) 下性能释放不充分, 不满足本设计要求, 无需进行相关测试。

表 4.2 Jetson 电源功耗测试

状态	屏幕直连方式登录	SSH 方式登录
空载	0.82A, 5.1V	0.3A, 5.0V
CPU 满载, GPU 空载	1.41A, 5.0V	0.96A, 5.0V
CPU 空载, GPU 满载	1.77A, 4.9V	2.13A, 5.0V
CPU 满载, GPU 满载	2.71A, 5.0V	3.02A, 5.0V

4.3 系统稳定性测试

因本设计面向成熟产品进行开发, 则对系统的稳定性、可靠性提出较高要求。对系统进行不同时间的通电测试, 观测其在该段时间内的相关数据表现。

首先是对电源模块的稳定性进行测试。将电源模块单独上电、静置 8 小时, 每间隔 0.5 小时使用示波器对其输出电压进行抽样测量, 结果如表 4.3 所列。

表 4.3 电源模块稳定性测试结果

时间节点	0.5h	1.0h	1.5h	2.0h	2.5h	3.0h	3.5h	4.0h
电压	5.08V	5.02V	5.01V	4.95V	5.10V	5.10V	5.05V	5.05V
时间节点	4.5h	5.0h	5.5h	6.0h	6.5h	7.0h	7.5h	8.0h
电压	5.12V	5.00V	5.00V	5.10V	4.95V	5.10V	5.16V	5.10V

对系统稳定性进行测试。连续 5 天在同一时间段内对系统进行连续 5 小时的通电测试，在同一时间节点（第 4.5 小时）观测 NVIDIA-jetson 的散热风扇温度大小、GPU 占用率、电路连接是否正常、所有功能是否正常，结果如表 4.4 所列。室内环境温度稳定于 25℃。

表 4.4 系统稳定性测试结果

时间	第 1 天	第 2 天	第 3 天	第 4 天	第 5 天
风扇温度	26.9℃	28.5℃	27.9℃	29.1℃	28.2℃
GPU 占用率	55%	58%	64%	55%	59%
电路连接	正常	正常	正常	正常	正常
系统功能	全部正常	全部正常	全部正常	全部正常	全部正常

实验结果表明：系统稳定性符合要求。

4.4 确定人脸特征欧氏距离阈值

在 `lookup_known_face` 函数中，若实时获取的人脸特征值与已注册人脸库中某一面孔距离小于某一阈值 k ，则认为是同一个人。

这里对 k 的取值进行试验，样本为 40 人，其中 10 人已注册；另外 30 人未注册。分别重复进行了 3 次试验，结果如表 4.5、4.6、4.7 所列。

表 4.5 试验 1 结果

k	已注册者未通过识别人数	未注册者识别通过人数
0.4	1	0
0.5	0	0
0.6	0	2

表 4.6 试验 2 结果

k	已注册者未通过识别人数	未注册者识别通过人数
0.4	0	0
0.5	0	0
0.6	0	1

表 4.7 试验 3 结果

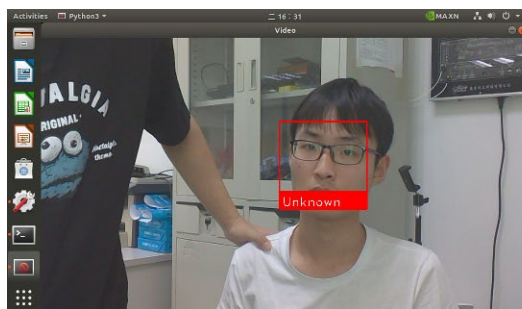
k	已注册者未通过识别人数	未注册者识别通过人数
0.4	1	0
0.5	0	0
0.6	0	0

根据以上试验结果，人脸距离阈值 k 取 0.5 为宜。

4.5 功能实现

在系统上电开机时会有一段初始化时间（约 10s），此时间段内系统信息输出接口（TJC4827X343）将会显示本次毕业设计相关的信息，播报相关的语音提示。提示语为：“安徽建筑大学本科生毕业设计！”、“指导老师：邵慧！”、“作者：许准！”、“系统初始化完成！”。在“系统初始化完成”提示语播报完成后进入选择操作界面，即开机完成，进入正常工作状态。

人脸识别解锁功能。系统在当前人脸识别通过之后将解锁命令发送至 STM32 单片机，由 STM32 控制数字舵机完成解锁门禁动作；同时播报“您使用人脸验证通过”提示语。



(a)实时捕获的画面



(b)实时邮件截图

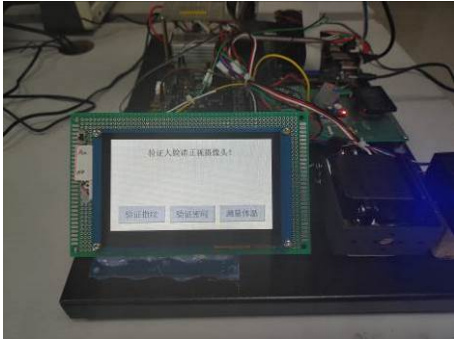
图 4.2 邮件报警

邮件报警功能。在“陌生人”靠近本系统时系统将发送该面孔图像至主人指定邮箱（984623208@qq.com），附具体时间，具体现象见图 4.2 所示，(a)为系统所实时捕获的画面截图，(b)为主人电子邮箱的实时邮件截图。

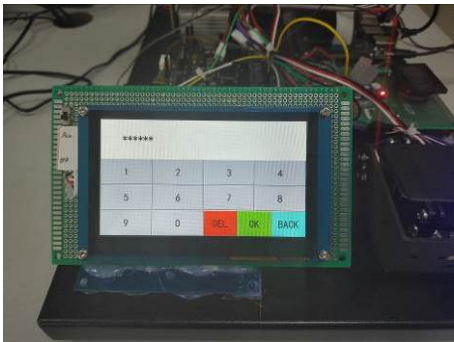
密码方式解锁功能。将系统面板上的拨码开关拨至“ma”端（手动端），点击“验证密码”按钮（图 4.3(b)），系统将展示密码键盘输入界面。其中用户所输入的密码“*”展示，如图 4.3(c)。



(a) 系统开机界面



(b) 主界面



(c) 密码键盘

图 4.3 密码解锁功能实际效果

指纹方式解锁。主界面内选择“验证指纹”按钮即可跳转至指纹界面，同时播报语音“请按指纹”；若 3s 内无指纹按下，则系统屏幕和语音播报提示“系统未检测到指纹！”，返回锁定主界面。若指纹验证通过，则播报“指纹验证通过！”，同时显示当前用户的指纹 ID，舵机响应解锁门禁动作，如图 4.5 所示。

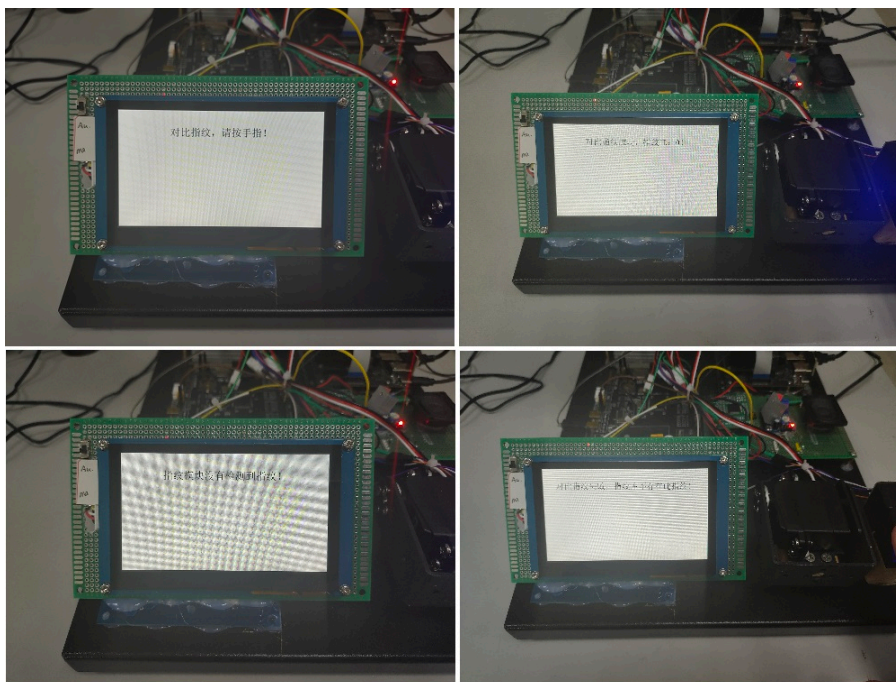


图 4.4 指纹方式解锁界面

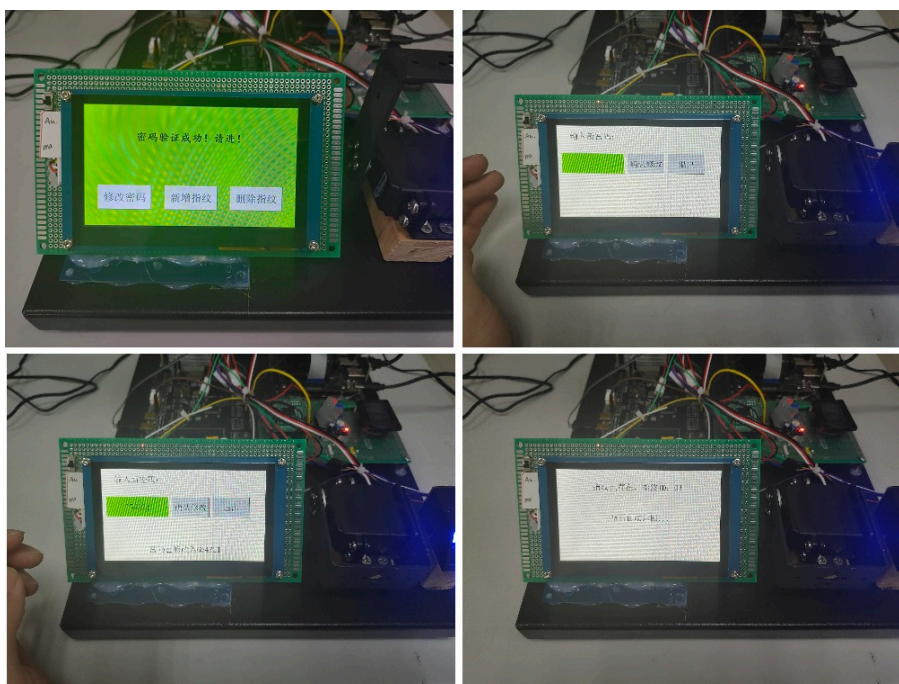


图 4.5 更改密码及指纹界面

更改密码、新增及删除指纹功能，如图 4.5。若新增的指纹已存在，则会提示“指纹已存在，指纹 ID: 0”字样，“0”为当前指纹在指纹库中的 ID；并自动返回至锁定状态。

无接触测量温度功能。在主界面选择“测量温度”按钮，系统即播报“请靠近测温传感器”，并显示的相关温度信息(图 4.6)，约 5s 后自动返回至主界面。

对于本系统的测量温度功能，MLX90614 已出厂校验、线性化处理，且测量范围宽（-70℃~380℃），也可进行异常高温检测。

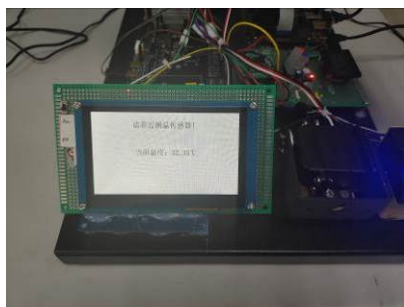


图 4.6 测量温度界面

第五章 总结与展望

5.1 总结

本文完成基于 NVIDIA-jetson 的智能门禁系统的设计和实现，实现基本的人脸识别解锁功能，并拓展了较多功能，如陌生人脸的邮件报警、实时语音提示、指纹及密码解锁等。

针对陌生来访者预警设计了自动邮件报警功能，提高了安全性；解锁方式多样化且均无需携带“实体钥匙”，便捷高效；采用 17kg 扭力舵机，可直接安装于现实门锁。

5.2 后续工作展望

综合考虑算力、市场价格等因素，主控选择为 NVIDIA-jetson Nano(2GB)，而 NVIDIA 公司的 Jetson AGX 系列和 Jetson Orin 系列开发平台算力是 Nano 的几何倍数，如果选择 AGX 或 Orin 作为主控模块，将大幅提升系统的输出视频流畅度。

因 NVIDIA-jetson 对电源适配器要求较高：功率需大于 10W（5V/2A）以保障平台可以满载满功率运行；电源纹波要小，否则将导致平台频繁死机甚至烧毁主板。因此，本系统在设计电源模块时将 Jetson 电源单独适配，后续工作中将设计出符合 Jetson 工作要求的电源模块，将整个系统的电源进行封装处理。

致 谢

鲜衣怒马少年时，不负韶华行且知。行文至此，我的本科生涯也将画上句号，一路走来，诸多不易。

首先向恩师邵慧教授表示由衷的感谢，有师如此，庆幸之至。邵老师在这四年里给了我太多的帮助与提携，学生的感激之情溢于言表。邵老师治学严谨、一丝不苟但和蔼可亲，她曾亲力亲为地带我参加过大大小小的各类竞赛。大三考研时，老师更是破例允许我在她的研究生实验室中学习，给我提供了优渥的学习环境。我仍记得在去年研考报名时自己备考状态不佳，萌生退意，便询问邵老师自己能否继续备考电子科技大学。老师当即给我非常肯定的答复：“可以考！”，给了我莫大的信心与鼓励。复试前夕，老师更是尽全力地把我推荐给自己在成电工作的朋友、为我亲笔写了专家推荐信，并承诺我“万一真的发挥失常，我这里还有一个位置留给你！”……也许我不是千里马，但邵老师一定是我的伯乐！愿吾师身体健康，桃李满园！

感谢永远在背后默默支持我的父母亲和一直支持我考研的姐姐。你们是我求学路上的不竭动力和坚实后盾，谢谢你们给了我追求美好未来的机会。养育之恩，无以为报。唯踏实向前、加倍努力，继续成为你们的骄傲。愿家人长乐常安。

人生遇知己，足以慰风尘！感谢我的挚友杨海燕和王重，相知十载，忧喜同享。时间无言，却证明了所有真心！愿18级杨同学东大研途顺利，愿19级王同学一战成硕，愿我们前程似锦！

感谢我的考研老师张宇，您是我漆黑考研路上的一道柔光。感谢21届硕士何、郭、夏三位师兄对我的照顾，感谢身边的室友和常联系的朋友们在考研路上的相互陪伴和激励……愿大家学业有成、事业顺利！

结束亦是新的开始，我也将奔赴下一段征程。再见了紫云路292号，再见了我的本科时代！

参考文献

- [1] 何燕琴,吴恋,郭清粉,曾桂南.人脸识别技术发展现状与未来趋势分析[J].无线互联科技,2021,18(13):80-82.
- [2] 王佳颖,黄章红,马万钧,罗杨,彭依.基于百度 AI 人脸识别的考勤系统设计与实现[J].电脑编程技巧与维护,2021(04):118-119.
- [3] Marwa Mohammed. Face Recognition based Texture Analysis Methods[J]. International Journal of Image, ,2019(07):11-19.
- [4] 景亮,赵程,刘平.人脸识别技术在城市轨道交通自动售检票系统中的应用及探讨[J].上海建设科技,2021(02):24-26.
- [5] 王瑶,李科,孙家炜.闸机式三维人脸识别系统[J].长江信息通信,2021,34(01):153-155.
- [6] 汪小艺,王珺,魏志强,郑英.基于人脸识别的图书馆智能门禁系统[J].电子产品世界,2021,28(04):65-67+80.
- [7] 范世朝,郑国强,孙国庆,韩旭.基于 FPGA 和卷积神经网络的人脸识别系统[J].电子元器件与信息技术,2021,5(05):98-99.
- [8] 王浩.基于 K210 芯片的 KPU 硬件加速人脸检测应用系统设计[J].办公自动化,2021,26(11):58-59+39.
- [9] 王伟彬.基于 Jetson Nano 平台对传统门锁智能化改造的探索[J].电脑编程技巧与维护,2022(01):114-116.
- [10] 周宇,杨国平,刘淼.基于 Jetson Nano 和 OpenCV 的智能门锁系统实现[J].智能计算机与应用,2022,12(02):120-122+125.
- [11] 胡佳玲,施一萍,谢思雅,陈藩,刘瑾.基于 Jetson nano 的改进 MobileNet 人脸识别系统[J].传感器与微系统,2021,40(03):102-105.
- [12] 施金汝,许凌峰,陈金立,陈璐,王辰.基于树莓派和 YOLOv5 算法的路面裂缝检测系统[J].信息技术,2022(04):22-28+34.
- [13] 明鑫.基于 AD623 及串口屏的智能电子称方案[J].自动化应用,2021(05):25-27.
- [14] 甄晓杰,周海涛,周曙东,陈世强.海康威视网络数字相机在植被物候监测的应用初探[J].中国农业文摘-农业工程,2021,33(03):65-73.
- [15] 杨向荣.Exynos4412 MIPI 摄像头接口研究[J].电子技术与软件工程,2015(12):134.
- [16] 林佳智,常凤筠,朱文斌,朱瑞昕,张天赐,王志鹏.基于 stm32 系统与物联网的 AI 多功能门禁系统研究[J].科学技术创新,2021(13):88-89.
- [17] 伍森.基于 XFS5152CE 的 CAN 总线声光报警系统设计[J].电子世界,2017(11):198-199.

- [18] 孙靖凡,李威,刘宇婕,杨文珺.基于树莓派 Linux 开发的两种编辑器研究及其应用实例[J].电脑知识与技术,2018,14(09):91-94.
- [19] 刘兆丰.Dlib 在人脸识别技术中的运用[J].电子制作,2020(21):39-41+7.
- [20] 宋建华,袁磊,李嘉辰,李祐来,侯珂,党莉彬.一种基于 HOG 和 SVM 的人眼开闭状态实时检测方法[J].牡丹江师范学院学报(自然科学版),2022(02):36-40.
- [21] 张岩,孙世宇,李建增,李德良,杜玉龙.鲁棒性交叠的标准特征描述子[J].北京理工大学学报,2017,37(08):842-847.
- [22] 耿硕,李云栋.基于深度学习的轻量级目标检测算法的研究[J].工业控制计算机,2022,35(04):97-99.
- [23] 张显.基于 SMTP 协议的嵌入式邮件通知软件[J].计算机系统应用,2015,24(11):209-212.
- [24] 陈诚,何锦清,孙华.基于 MLX90640 探头的热成像图划分区域的研究[J].电子制作,2021(05):88-90+76.
- [25] Sa Zhu, Jianbo Lu, Feng Zeng. Fast Calculation of Electromagnetic Forces in IPMSMs Under PWM VSI Supply Based on Small-Signal Time-Harmonic Finite Element Method[J]. CIES Transactions on Electrical Machines and Systems,2022,6(01):67-76.
- [26] 吴美之.电子仪器共地常出现的测量错误研究[J].通信电源技术,2018,35(07):203-204.