**基于Retinaface和Jetson Nano的智能门锁系统**

**Intelligent Door Lock System based on Retinaface and Jetson Nano**

**摘 要**

随着社会的不断发展和科技的迅猛进步，门锁系统作为人们生活中不可或缺的一部分，其安全性和便利性需求也日益提升。而人脸识别技术因其高效、准确的特性，在智能门锁系统中的应用备受关注。本文旨在设计并实现一种基于Nvidia Jetson Nano平台的智能门锁系统，结合Retinaface和FaceNet算法，实现高效、准确的人脸识别功能。通过对现有门锁系统及人脸识别技术的研究，探讨其发展现状，并提出改进方案，以提升门锁系统的安全性和便利性。

本文在第一二章介绍了门锁系统和人脸识别技术的发展背景及意义，分析了国内外相关研究现状。然后，设计了智能门锁系统的架构，并详细阐述了所采用的Retinaface和FaceNet算法的原理及优势。并结合PyQt界面和Esp8266控制舵机等技术，实现了系统的用户界面和硬件控制。最后，通过实验验证了算法的准确性和系统的稳定性，通过云平台的回传数据验证了门锁系统的可用性。结果表明，所设计的智能门锁系统在人脸识别方面表现出较高的准确率和快速响应能力，能够有效识别注册用户并进行门锁控制。智能门锁系统界面友好，操作简便，具有良好的用户体验。因此，本研究为智能门锁系统的发展提供了一种有效的解决方案，并具有广泛的应用前景。

本论文综合了对门锁系统和人脸识别技术的研究，通过实验验证了该系统的可行性和有效性，为智能门锁系统的进一步发展和实际应用提供了有益的参考和指导。在研究中，本文详细探讨了不同模型的实现方法，并对实验结果进行了分析，证明了其在提升安全性和便捷性方面的能力。这一研究成果不仅丰富了智能家居领域的技术储备，也为未来相关产品的设计和优化提供了重要依据。

关键词：人脸识别技术；Jetson Nano；Retinaface；Facenet；PyQt；人脸检测技术

**ABSTRACT**

With the continuous development of society and the rapid progress of technology, the locking system, as an indispensable part of people's lives, has increasingly higher demands for security and convenience. Facial recognition technology, with its high efficiency and accuracy, has attracted much attention in the application of smart locking systems. This paper aims to design and implement a smart locking system based on the Nvidia Jetson Nano platform, combining the Retinaface and FaceNet algorithms to achieve efficient and accurate facial recognition. By studying the existing locking systems and facial recognition technologies, this paper discusses the current status of development and proposes improvement solutions to enhance the security and convenience of the locking system.

In the first and second chapters, the background and significance of locking systems and facial recognition technologies are introduced, and the current research status at home and abroad is analyzed. Then, the architecture of the smart locking system is designed, and the principles and advantages of the Retinaface and FaceNet algorithms adopted are elaborately described. And by combining PyQt interface and Esp8266 control servo technology, the system's user interface and hardware control are realized. Finally, the accuracy and stability of the algorithm are verified through experiments, and the usability of the locking system is verified through the data returned by the cloud platform. The results show that the intelligent door lock system designed in this study has a high accuracy and fast response rate in facial recognition, which can effectively identify registered users and control the door lock. The intelligent door lock system has a user-friendly interface and simple operation, with a good user experience. Therefore, this study provides an effective solution for the development of intelligent door lock systems and has broad application prospects.

This thesis integrates research on door lock systems and facial recognition technology, verifies the feasibility and effectiveness of the system through experiments, and provides useful reference and guidance for the further development and practical application of intelligent door lock systems. In the study, the author has discussed in detail the implementation methods of different models and analyzed the experimental results to prove the system's ability to enhance security and convenience. This research not only enriches the technical reserves in the field of smart home, but also provides important basis for the design and optimization of related products in the future.

**Key words**:Face recognition; Jetson Nano；Retinaface；Facenet；PyQt；Face detection

目 录

[第一章 绪论 1](#_Toc167788124)

[1.1 背景及意义 1](#_Toc167788125)

[1.1.1 背景 1](#_Toc167788126)

[1.1.2 意义 1](#_Toc167788127)

[1.2 门锁系统的发展及人脸识别技术的国内国外研究现状 1](#_Toc167788128)

[1.2.1 门锁系统的国内国外研究现状 1](#_Toc167788129)

[1.2.2 人脸识别技术的国内国外研究现状 2](#_Toc167788130)

[1.3 主要研究内容 3](#_Toc167788131)

[1.4 本章小结 3](#_Toc167788132)

[第二章 智能门锁系统架构设计及算法 5](#_Toc167788133)

[2.1 Nvidia Jetson Nano平台 5](#_Toc167788134)

[2.2 智能门锁系统的工作逻辑 5](#_Toc167788135)

[2.3 Retinaface+Facenet人脸检测算法 7](#_Toc167788136)

[2.3.1 Retinaface算法的基本介绍 7](#_Toc167788137)

[2.3.2 FaceNet算法的基本介绍 11](#_Toc167788138)

[2.3.3算法的优势 13](#_Toc167788139)

[2.4 PyQt界面和舵机 14](#_Toc167788140)

[2.4.1PyQT介绍 14](#_Toc167788141)

[2.4.2基本页面介绍 14](#_Toc167788142)

[2.4.3MG996R舵机 16](#_Toc167788143)

[2.5 本章小结 17](#_Toc167788144)

[第三章 Retinaface+FaceNet算法的实现流程 18](#_Toc167788145)

[3.1 Retinaface+FaceNet算法的实现 18](#_Toc167788146)

[3.1.1 Retinaface算法的实现 18](#_Toc167788147)

[3.1.2 FaceNet算法的实现 25](#_Toc167788148)

[3.2 Retinaface+FaceNet的图片检测模式与视频检测模式的实现 26](#_Toc167788149)

[3.2.1 数据库的初始化 26](#_Toc167788150)

[3.2.2 人脸的截取与对齐 27](#_Toc167788151)

[3.2.3 利用FaceNet对人脸进行识别 29](#_Toc167788152)

[3.2.4 利用Retinaface+FaceNet实现视频检测 30](#_Toc167788153)

[3.3 本章小结 31](#_Toc167788154)

[第四章 PyQt界面的制作 33](#_Toc167788155)

[4.1 登录页面的实现 33](#_Toc167788156)

[4.2 主界面的实现 34](#_Toc167788157)

[4.3 人脸录入界面的实现 35](#_Toc167788158)

[4.4 Esp8266控制MG996R舵机 37](#_Toc167788159)

[4.5 登录页面的实现 39](#_Toc167788160)

[第五章 总结与展望 39](#_Toc167788161)

[5.1 本文总结 39](#_Toc167788162)

[5.2 未来展望 40](#_Toc167788163)

[参考文献 41](#_Toc167788164)

[致 谢 **错误!未定义书签。**](#_Toc167788165)

**第一章 绪论**

1.1 **背景及意义**

1.1.1 **背景**

在科技发展的推动下，门锁系统的演进日新月异。从原始、简单的机械构造，门锁技术已迈向了一个全新的智能化阶段，这一转变涵盖了生物识别、通讯技术、机械工程以及计算机科学的深度融合与应用。门锁系统的初期阶段一般使用机械门锁，但由于其仅依赖于钥匙这一验证方式，容易被盗用，安全性不足。所以，改进后的门锁就诞生了，它有电子锁，也有电子卡，可以用密码和磁卡进行识别。随着网络、物联网、计算机和生物特征识别等技术的不断进步，门禁系统越来越具有安全性、可靠性、便利性和便于管理的特点。

1.1.2 **意义**

在生物辨识领域，指纹识别采用了接触式收集方法，但是其在卫生、方便等方面仍有缺陷，影响了用户的使用体验。虽然仪器价格低廉，但是客观环境容易对测量结果产生影响。虹膜识别技术是一种鉴别能力强、识别速度快的技术，但由于采集设备昂贵，容易引起图像畸变，从而影响到原图像的提取。相比之下，指静脉识别由于其部位在人体内，受外部环境影响较少，因此具有较大优势。然而，其严格的设计制作要求和高成本限制了其在大范围内进行普及。

人脸识别技术凭借普通摄像头即可轻松采集信息，为用户提供了便捷的基础信息获取途径，同时非接触的识别过程确保了良好的用户体验。对于高度安全需求的应用场景，该技术凭借其出色的活性分辨能力，有效防范了非动态图片、木偶或蜡像等欺诈手段，并且，人脸识别技术的实施并无需大量特殊装置，只需对现有计算机系统进行功能升级即可实现，不仅可以保证使用者的资金投入，而且可以扩充装置的功能，更能满足使用者的安全性要求。相对于其它的身份识别方法，人脸识别是一种性能价格比高的技术。

因此，在门锁系统中引入人脸识别技术，是一项很有实用价值的研究课题。实现该系统所面临的主要挑战在于如何在缩短识别所需时间的同时把准确率进一步提升，并确识别的实时性和整个门锁系统的安全性。

1.2 **门锁系统的发展及人脸识别技术的国内国外研究现状**

1.2.1 **门锁系统的国内国外研究现状**

在国内，文献[1]设计出的门锁系统，利用手背静脉识别的方式，进行了门锁系统的设计，并且基于这种方法，文献提出了引入欧氏距离的方法，提高了准确性，但是在日常的生活中，用户需要把手背放在专门的设备上进行识别，相较于其他无接触的识别方式，使用略不方便。

文献[2]所设计的门锁系统，是一种基于ZigBee技术的智能门锁系统。其优点在于，用户能够通过远程控制端下发控制指令，实现对门锁的远程控制，同时门锁端可以监控门锁的启动与关闭状态并将信息上传到远程控制端，使得整个系统的电量消耗较低。该系统存在的缺点在于，无线信号容易受到其他电子设备或物理障碍的干扰，可能导致通信不稳定或中断，影响门锁的正常使用。

文献[3]提出了一种虹膜识别智能门锁，其优点在于，虹膜具有固有的个体特征，并且难以伪造或模仿，虹膜识别门锁具有很高的安全性，能够有效防止非法进入。因为每个人的虹膜图案是独一无二的，几乎可以达到百万分之一的识别误差率。虹膜识别智能门锁所存在的问题在于，他的设备成本相对较高，包括识别设备的采购、安装和维护成本。

在国外，文献[4]中提到的门锁设计了一种门锁系统，当门锁遭遇物理冲击而受损时，系统会立即向已注册用户发送损坏通知，以此作为一道预警机制，防止潜在的恶意破坏行为，确保用户的财产安全。然而，尽管系统能迅速传递信息，但它本身并不具备直接阻止门锁被非法开启的能力。

文献[5]设计了两层确认，第一层利用摄像头验证，第二层使用指纹进行确认，在确认的最后阶段，如果闯入者试图在进行面部识别或者，没有进行指纹验证的情况下，打破入口进入房间，加速度传感器会发出警告。该框架还利用哈尔过程计算来对抗发现，利用特征面计算来进行对抗确认。这一系统适用于对安全性需求度较高的场景，但对于居民日常的使用来说，使用这种复杂度高、安全度高的门锁，反而是造成了麻烦。

文献[6]也采用了RFID的方式进行识别，虽然此类门锁也是无接触式结构，有着成本低、性能先进、界面简单等优点，但是由RFID和Arduino组成的门锁系统过于简单，会有被破解的风险，所以，此类门锁并不适合用于做普通家庭的门锁。

1.2.2 **人脸识别技术的国内国外研究现状**

在人脸识别门锁系统中，由于小区居民人数不多，所以小区门口的门锁系统仅对小区住户进行门锁管理，使得人脸图像的测试样本数量很小。另外，受硬件条件的制约，采样率也会受到很大的影响。

在国内，文献[7]首先学习多分类人脸识别任务，再进行人脸的验证工作，这一模式可以推广到其他任务或者未知身份的新训练集，其主要思想是通过深度学习得到了高度紧凑和有判别度的特征DeepID。首先，将人脸图像输入到DeepID网络中。这些图像可以是彩色图像或灰度图像，通常会经过预处理，如归一化和尺寸调整，以满足网络的输入要求。DeepID网络通过多层卷积神经网络(CNN)来学习图像的特征表示。在这个阶段，每个卷积层都会提取出图像的不同层次和抽象程度的特征，例如边缘、纹理和面部结构等。在特征提取的过程中，深层网络会逐渐融合和整合图像中的信息。通常在卷积层之后会连接全连接层，将卷积层提取的特征进行进一步处理和融合，以生成更加抽象和高维度的特征表示。最后，经过特征融合的过程后，DeepID网络将输出的特征向量再输入到分类器softmax中，其意义在于后续执行人脸识别的分类任务。softmax分类器会计算每个人脸的识别分数，并将其映射到对应的类别标签，以确定输入图像属于哪个人。在整个流程中，DeepID网络的参数会通过训练数据进行学习和调整，以最大化识别准确率。训练数据通常包括大量的人脸图像和对应的标签，用于指导网络参数的优化。经过训练后的DeepID网络可以用于识别未见过的人脸图像。当输入一张新的人脸图像时，网络会根据学习到的特征表示和分类器进行识别，输出图像对应的类别标签，即可识别出人脸所属的身份信息。

在国外，文献[8]中所展示的VGG-Face模型由3个全连接层和11个层、8个卷积层组成，每一个卷积层后再执行一个ReLU函数，在每一个卷积模块的末尾执行一个max pool操作。VGG网络相较于LeNet网络的一个显著改进，在于它选择使用多个较小尺寸的卷积核来替代原先的大尺寸卷积核。这种策略在保持感受野大小不变的同时，通过叠加更多的小卷积层，有效地增加了网络的深度。这样做不仅优化了计算效率，还显著提升了网络在特征提取方面的能力。VGG-Face模型首先利用大规模的人脸图像数据集(如CASIA-WebFace或LFW)对网络进行预训练，去通用的人脸特征表示进行学习。接下来，为了让人脸识别任务更加精准，文献[8]使用针对该任务的数据集对网络进行精细调整，从而使其更好地适应并满足特定的人脸识别需求。其优点在于VGG-Face通过深度卷积神经网络学习到了高度抽象的人脸特征表示，具有较高的识别准确率。并且基于CASIA-WebFace与LFW数据集，可以使模型具有较好的泛化能力。解决了在人脸识别领域中，存在的识别精度和泛化能力的问题。

1.3 **主要研究内容**

本文分为五个主要章节，主要研究基于Retinaface和Jetson Nano的智能门锁，通过Retinaface算法来实现人脸识别，并把模型部署在Jetson Nano平台，完成智能门锁的整体设计。

第一章是绪论。本章对Retinaface算法和Jetson Nano进行了探讨和分析，深入研究了算法的实现逻辑和部署模型的方法，对本课题的总体进程作进行了细致地规划。

在第二章中，本章探讨了国内以及国外的智能门锁的发展现状，这是一项非常重要的任务。并且还研究了国内国外的人脸识别算法的现状，这一章为后续的展开奠定基础。

在第三章中，本章对Retinaface和Facenet的算法进行了解析与代码部分的实现，这项工作是系统的核心算法部分。

在第四章中，本章对PyQT和MG996R舵机进行了设计，由PyQt创建了本项目的主界面等界面，后续对Esp8266的控制代码进行了设计，并配置了阿里云，舵机启动的消息会通过Mqtt协议回传到阿里云。

在第五章中，本章对完成的智能门锁系统进行了总结与分析，总结了在设计、制作智能门锁过程中所遇到的问题，并后续对其进行了分析与讨论。最后，对智能门锁的发展前景进行了探讨。

1.4 **本章小结**

第一章介绍了门锁系统的发展背景及其智能化的意义。门锁技术从机械锁进化到现代智能锁，融合了生物识别、通讯技术、机械工程和计算机科学，提升了安全性和便利性。特别是人脸识别技术，由于其非接触性和高效性，成为智能门锁系统研究的重点。通过分析国内外智能门锁及人脸识别技术的现状，本文确定了研究方向，主要研究基于Retinaface算法和Jetson Nano平台的人脸识别智能门锁系统。本章为后续章节的研究奠定了基础，明确了研究的背景、意义和主要内容。

**第二章 智能门锁系统架构设计及算法**

本章对智能门锁系统的每一个组成部分进行了详细描述，尤其是人脸检测和人脸识别算法。也对课题整体的完成情况和工作逻辑进行了详细地介绍。

2.1 **Nvidia Jetson Nano平台**

Nvidia Jetson Nano是一款由Nvidia推出的小型嵌入式计算机，如图2.1所示，其旨在提供高性能的人工智能计算能力。Jetson Nano采用了Nvidia的Tegra系列系统级芯片，这款设备配备了四核 ARM Cortex-A57 处理器和 Nvidia Maxwell 架构的 GPU，带有 128 个 CUDA 核心，同时还搭载了 4GB LPDDR4 内存。提供了强大的计算能力，支持各种人工智能算法和深度学习模型的部署和运行。并且有丰富的连接接口，包括4个USB 3.0端口、1个Gigabit Ethernet端口、1个HDMI输出端口、1个DisplayPort输出端口、1个MIPI CSI摄像头接口和1个MIPI DSI显示器接口等，与此同时，Jetson Nano采用了低功耗设计，功耗仅为5-10瓦，适合于嵌入式应用和便携式设备。

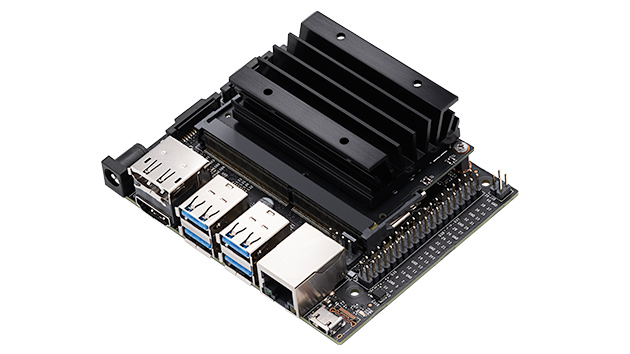


图2.1 Jetson Nano B01

2.2 **智能门锁系统的工作逻辑**

智能门锁的工作逻辑如图2.3所示，系统是由Jetson Nano作为主控芯片，PyQt5界面作为用户端窗口，系统启动后，为用户展示登录界面，用户在登录后，可以进行检测视频、检测照片、人脸编码、录入人脸的功能，启动程序后，选择检测视频模式，程序调用Retinaface和FaceNet开始实时对人脸进行检测，当检测到的人脸信息与在数据库中的人脸信息匹配后，把检测结果到Esp8266，假如Esp8266收到了身份确认的消息，则开门，反之，假如Esp8266没有收到确认身份的消息，则不开门，开门的信息会以日志形式回传到阿里云平台，用户可以远程监测门锁开启的时间。

用户在选择“检测图片”模式后，会弹出文件选择窗口，供用户选取测试图片，在系统目录下/img文件夹内有若干个图片可供测试，此功能用于测试模型能否工作。

用户在首次启动本系统时，要先选择“录入人脸信息”模式，点击此按钮会弹出新窗口，可以进行拍照、命名、保存，在人脸信息成功保存后，退出当前窗口，再点击“编码人脸信息”，系统则会自动开始编码，当编码完成之后，用户便可选择“检测视频”模式，查看系统能否正常工作。

项目的整体情况如图2.2所示，屏幕上显示的用户的登陆与注册界面，在用户登陆进入系统之后，即可开始执行人脸检测模式或人脸录入模式。Jetson Nano连接了Esp8266，假如人脸数据在数据中并且识别通过的话，Jetson Nano会向Esp8266发送指令，Esp8266收到指令后，控制舵机的开启。

图形用户界面

描述已自动生成

图2.2 智能门锁系统工作流程图

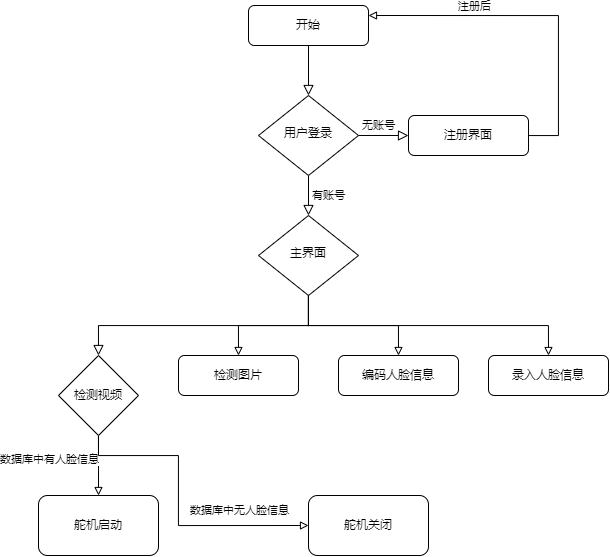


图2.3 智能门锁系统工作流程图

2.3 **Retinaface+Facenet人脸检测算法**

2.3.1 **Retinaface算法的基本介绍**

Retinaface算法在WiderFace数据集上取得了显著的AP提升，这一结果显示了其在人脸识别领域的出色性能。Retinaface的人脸检测方式如图2.4所示,该算法的源代码已在insightface平台上公开，原模型基于mxnet框架开发，但目前在社区中，基于其他框架的复现版本也颇受欢迎。Retinaface的核心创新在于在原有的检测网络RetinaNet基础上，巧妙地融合了三个层级的SSH检测模组，这一设计显著提升了人脸侦测的准确度。研究人员给出了两个不同的网络模型，前者是为了提高检测的速度，后者是为了提高检测的准确性。

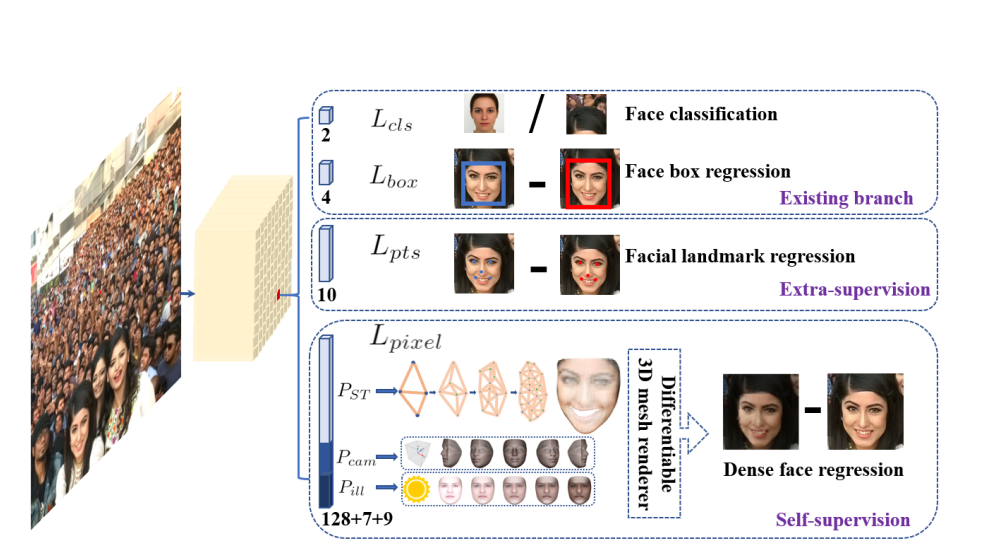


图2.4 Reinaface的人脸检测方式[11]

Retinaface网络对待检测的图片：首先，本文采用 MobileNet0.25 或者 ResNet50 进行主干特征网络提取。接着，结合使用了 FPN(特征金字塔网络)和 SSH来进一步增强特征提取的效果。随后，模型使用 ClassHead、BoxHead 和 LandmarkHead 模型从这一部分特征中得到预测后的结果。最后，本文对预测结果进行解码(decode)并利用 NMS(非极大抑制)算法来消除重复检测，从而得出最终的结果。

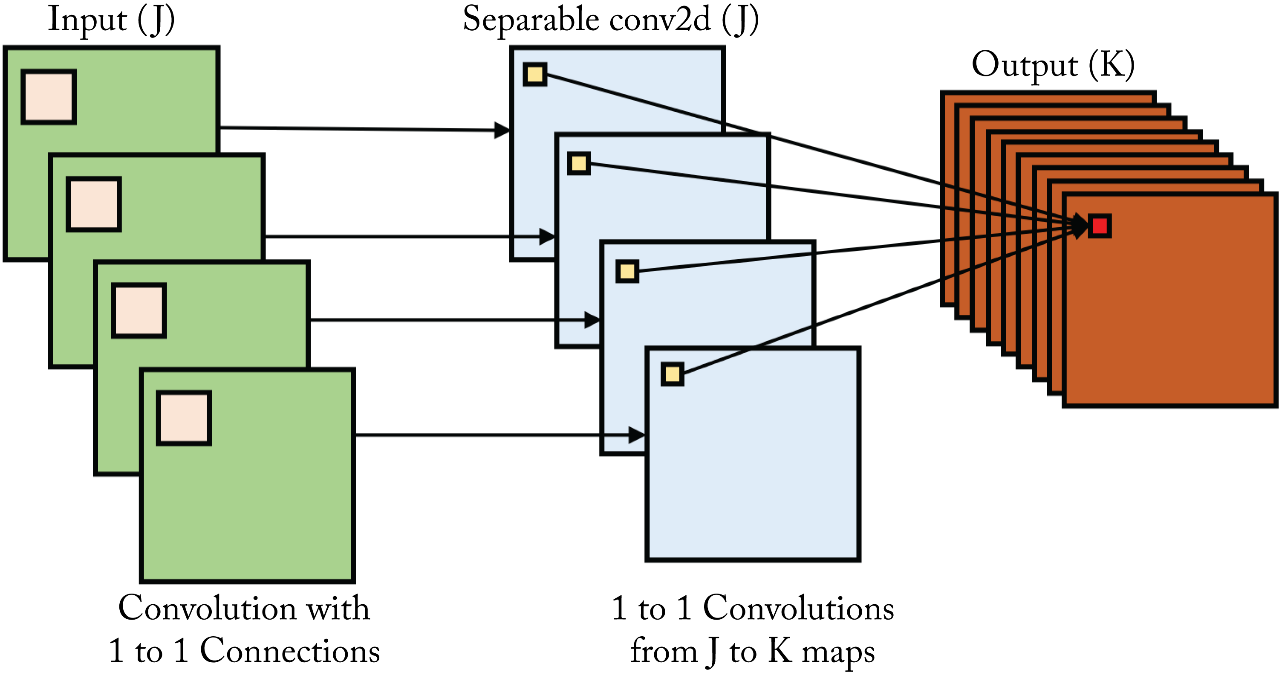
本项目的运行平台式Jetson Nano，因为其更适用于嵌入式设备，所以相比PC来说，算力较小，所以在本文中选择使用MobileNetV1作为主干的特征提取网络，这是基于MobileNet模型的优秀性能与轻量级特性。MobileNet是Google特别为手机等嵌入式设备设计的一种深度神经网络架构，其核心理念在于深度可分离卷积(depthwise separable convolution)如图2.5。这种卷积方式不仅大幅减少了计算量和参数数量，还保持了较高的模型性能，使得MobileNet成为在资源受限环境中进行高效特征提取的理想选择。

图2.5 depthwise separable convolution实现步骤[12]

该模型的参数被减少了，其运行的速度便会提高，这对于Jetson Nano这样的轻量化嵌入式平台来说较为合适，因此，本文在Retinaface网络中选择mobilenet0.25作为主干特征提取网络。

如图2.7所示就是MobileNet0.25的网络结构，相当于mobilenetV1-1通道数压缩为原来1/4的网络，通过对通道数的压缩，使MobileNet0.25能够在资源受限的设备上运行，比如移动手机、嵌入式设备等，且在保持较小模型规模的同时，能够保持良好的性能。

同时， Retinaface也利用 FPN的架构，如图2.6所示，在 Mobilenet的三条带状图的有效特征层上构造出 FPN结构。

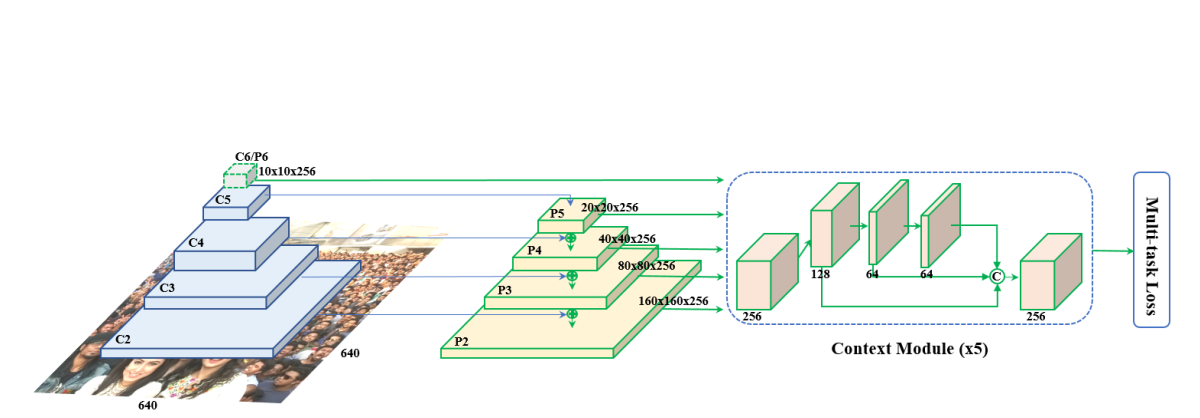


图2.6 Retinaface中的FPN金字塔结构[11]

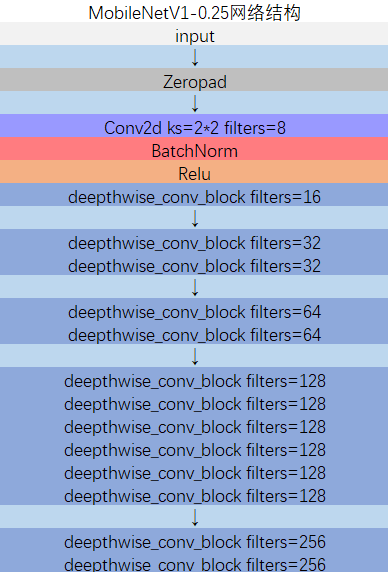


图2.7 MobileNetV1-0.25网络结构

该网络模型的构造方法是：先通过1x1卷积来调节三个有效特征层的信道数目；在此基础上，采用 Upsample与 Add对图像进行特征融合。FPN金字塔在模型中的构建方式如图2.8所示。

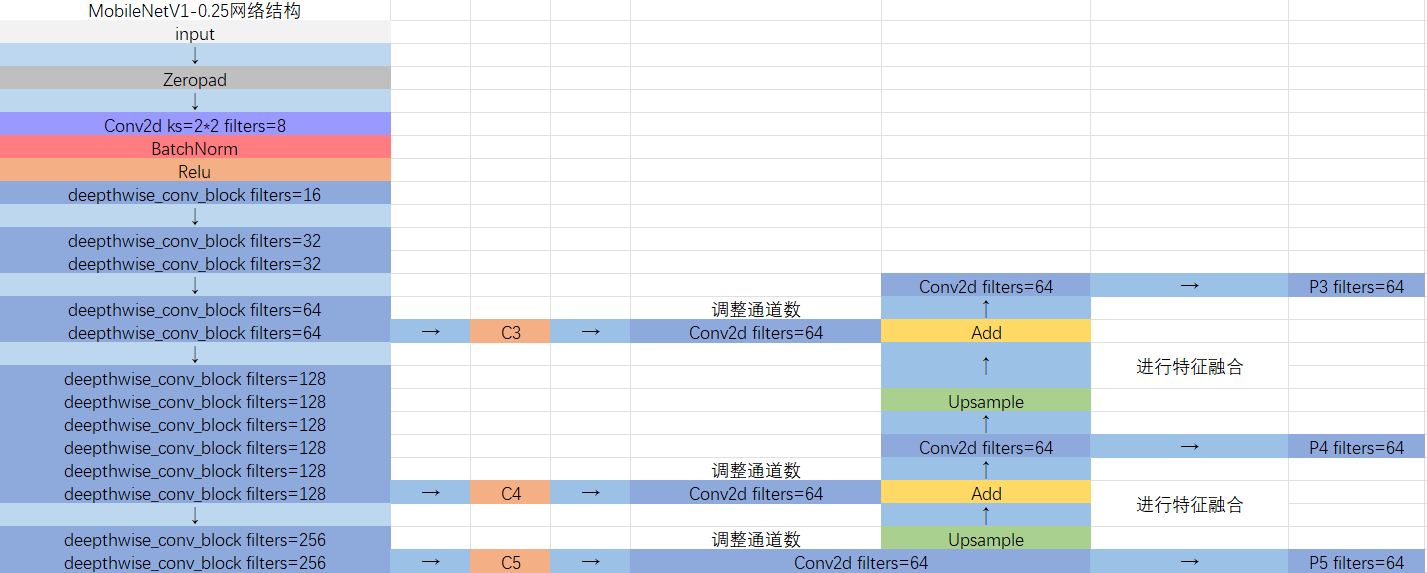


图2.8 FPN金字塔在模型中的构建方式

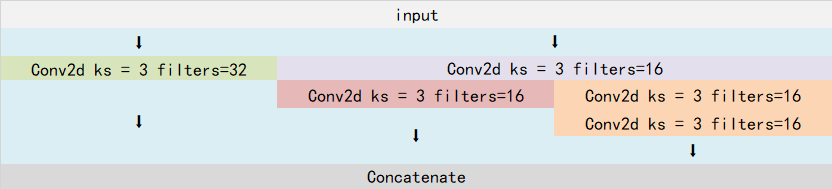
在本文的运算流程中，获取了P3、P4、P5三个关键特征层。为了进一步优化和强化这些特征层的提取能力，Retinaface算法巧妙地引入了SSH(Scale-Invariant Feature Transform，SSH)模块，SSH的网络结构如图2.9所示，以拓宽网络的视野范围。SSH模块的设计精髓在于其并行结构，通过巧妙地堆叠3x3卷积来模拟更大尺寸卷积的效果：具体来说，左侧分支仅使用单一的3x3卷积，中间分支则通过连续两次3x3卷积来近似5x5卷积的效果，而右侧分支则通过三次3x3卷积的叠加来模拟7x7卷积的功能。这种设计在提升特征丰富度的同时，也保证了计算效率，从而有效增强了模型的特征提取能力。

图2.9 SSH的网络结构

所以使用SSH模块可以利用现代硬件的并行计算能力加速模型的推理过程。这一点对于实时性要求较高的人脸检测任务尤为重要。

通过以上的介绍，Retinaface结合了MobileNetV1-0.25、FPN特征金字塔和SSH模块，所以本项目采用了Retinaface作为人脸检测的网络。

2.3.2 **FaceNet算法的基本介绍**

FaceNet的核心策略在于利用卷积神经网络来学习人脸图像在欧式空间中的特征表示。这种方法的设计初衷在于通过训练，使得在欧式空间中，两张属于同一人的人脸图像所对应的特征向量之间的距离尽可能小。换言之，如果两幅人脸图像的特征向量在欧式空间中的距离越近，那么这两幅图像属于同一人的可能性就越大。这种策略为人脸识别任务提供了一种高效且准确的方法，使得在海量的人脸数据中快速、准确地识别出目标个体成为可能。此算法的实现过程：首先计算人脸特征的特征向量，然后通过比较距离并设置阈值，即可确定两张人脸照片是否来自同一人。在成功构建人脸图像特征提取模型后，人脸验证过程实质上转化为了一个比较过程：比较两幅图像特征向量的相似度与预设阈值的大小。对于人脸识别任务，问题进一步转化为在已提取的特征向量集中进行K近邻分类的问题，通过寻找与待识别图像特征最相近的K个已知类别特征，来确定其所属身份。此外，对于人脸聚类任务，本文可以采用k-means聚类算法对提取到的人脸特征集进行分组，从而实现将相似的人脸图像聚类到同一个类别中的目的。这样，不同的人脸图像处理任务就可以通过这些高效且准确的方法得到妥善解决。

传统算法通常通过使用已知身份的人脸图像集来训练分类模型，然后提取中间某个层的输出作为人脸的特征表示。这种方法存在两方面的弊端：一是希望学习的指定层的特征能够很好地泛化到未知人脸上；二是特征维度通常很高(大于1000维)，而有些方法对提取的特征进行PCA降维处理，但这只是一个线性变换，可以通过一个网络层很简单地实现。

FaceNet模型通过triplet loss直接训练以生成128维的特征向量。FaceNet的网络结构如图2.10所示。在训练过程中，triplets是由三个图像构成的，包括两张来自同一人的图像和一张来自不同人的图像。训练的核心目标是确保在欧式空间中，属于同一人的两张人脸图像的特征向量之间的距离显著小于来自不同人的图像对之间的距离。输入到FaceNet中的人脸图像是初步检测的结果，并未进行额外的二维或三维对齐预处理，这进一步展示了FaceNet在处理人脸数据时的灵活性和鲁棒性。FaceNet用到了两个网络，分别是ZFNet和MobilenetV1，在选定神经网络架构后，训练过程采用端到端的方式，将网络视为一个“黑盒子”进行处理。在训练过程中，本文重点关注的是网络输出的triplet loss，该损失函数不仅指导着网络的权重优化，同时也是衡量人脸特征提取、识别和聚类性能的关键指标。通过这种方式，本文能够有效地提升网络在人脸相关任务上的性能。

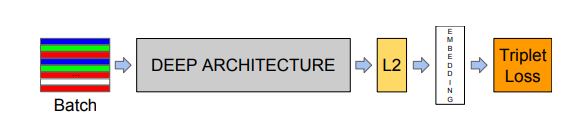


图2.10 FaceNet的模型结构

FaceNet的主干网络起到提取特征的作用，为了保证模型能够在Jetson Nano平台正常运行，在本文中选用了MobileNetV1作为主要的特征提取网络。该网络能够从输入的图片中提取出丰富的特征，形成一个形状为(batch\_size, h,w,channels)的特征层。为了简化后续的处理流程，并减少计算复杂度，本文采用全局平均池化操作将特征层的空间维度(h,w)压缩为(1,1)，从而得到一个形状为(batch\_size,channels)的特征向量。

接下来，为了将特征向量映射到一个更适合后续任务(如人脸识别或聚类)的维度，本文应用了一个全连接层，该层的神经元数量为128。通过这一步骤，本文得到了一个长度为128的特征向量，这个向量可以被视为输入图片的特征浓缩，它有效地捕捉了原始图片中的关键信息，为后续的任务处理提供了便利。为了使得不同人脸的特征向量可以属于同一数量级，方便进行比较，所以本文在获得一个长度为128的特征向量后，还需要进行l2标准化的处理。在进行L2归一化处理之前，一个重要的前置步骤是计算特征或数据的2-范数，通常也被称为欧几里得范数。



当完成l2标准化之后，输入模型的图片已经被l2标准化为向量了，长度为128。l2标准化可以减少向量之间的差异，使得模型更加稳定并且对输入数据的变化更具有鲁棒性，增强了模型对人脸特征的区分能力。所以本文采用FaceNet作为人脸识别的网络模型。

2.3.3**算法的优势**

Retinaface相比文献[9]中的SSD算法来说，有以下几点优势：

* RetinaFace：RetinaFace采用了深度学习模型，通过多任务学习来同时进行人脸检测、关键点定位和人脸角度估计。这种综合考虑多个任务的方法通常能够提高准确率，使得RetinaFace在复杂场景下的人脸检测效果较好。
* SSD：SSD是一种流行的目标检测算法，其设计简单高效，采用单个神经网络进行目标检测，因此可以在保证较高准确率的同时提高检测速度。然而，相比于RetinaFace，SSD可能在人脸检测的准确率上略有差异。
* RetinaFace：RetinaFace采用了深度学习模型，通过多任务学习来同时进行人脸检测、关键点定位和人脸角度估计。这种综合考虑多个任务的方法通常能够提高准确率，使得RetinaFace在复杂场景下的人脸检测效果较好。
* SSD：SSD是一种流行的目标检测算法，其设计简单高效，采用单个神经网络进行目标检测，因此可以在保证较高准确率的同时提高检测速度。然而，相比于RetinaFace，SSD可能在人脸检测的准确率上略有差异，尤其在边界框精度和小目标检测方面。

FaceNet相比文献[10]所提出的ArcFace相比，有以下几个优势：

* FaceNet：FaceNet通过深度学习模型学习到人脸图像的特征表示， triplet loss通过优化特征向量来确保来自相同个体的人脸图像在特征空间中的欧式距离尽可能接近，而来自不同个体的人脸图像之间的距离则尽可能拉远。FaceNet在运用triplet loss进行训练后，通常能够在人脸识别任务中展现出卓越的准确率。
* 它具有很好的跨平台性ArcFace：ArcFace是另一种基于深度学习的人脸识别算法，它引入了一种特殊的角度余弦距离和一种特殊的角度余弦距离损失函数，可以更好地优化特征向量的表示，提高人脸识别的准确率。
* FaceNet：FaceNet虽然需要训练一个深度学习模型，但是一旦模型训练好，特征提取过程通常可以在较短的时间内完成。并且FaceNet在训练的过程中用到了大量的数据，模型的泛化能力较强。
* ArcFace：ArcFace采用了一种特殊的损失函数来优化特征向量的表示，通常也需要较长的训练时间。但在进行人脸识别时，ArcFace通常能够快速地生成特征向量，并且由于其优化的特征表示，通常可以在较短的时间内进行准确的人脸匹配。

通过以上的对比，本文选择了Retinaface+FaceNet作为人脸检测和识别的模型，这样做既能满足人脸检测的精度要求，又能满足模型的轻量化和便捷性，使模型能够在Jetson Nano平台上运行。

2.4 **PyQt界面和舵机**

2.4.1**PyQT介绍**

PyQt5是一个用Python语言编写的Qt应用程序开发框架。它允许开发者使用Python语言创建跨平台的桌面应用程序，具有强大的图形用户界面(GUI)功能和丰富的Qt库支持。PyQt5具备出色的跨平台兼容性，支持在Windows、macOS以及Linux等主流操作系统上顺畅运行。这一特性极大地便利了开发者，他们仅需进行一次开发，即可轻松将应用程序部署到多个平台，几乎无需进行额外的修改或调整。这种跨平台的灵活性显著提高了开发效率，并确保了应用程序在不同操作系统下的一致性和稳定性。这一特点方便了开发者的开发，他们只需要在一个平台开发好基本框架，就可以很方便的进行跨平台的移植。

2.4.2**基本页面介绍**

通过PyQt5，本文创建了三个基本页面，分别是登录界面、主窗口、录入界面。程序启动后，会展示出登录界面，登录界面有两个基本功能，一个是登录，一个是注册，用户注册后的信息会存储到csv的表格中。登陆界面如图2.11所示。



图2.11 登录界面

当用户输入帐号密码后，会进入到人脸检测系统的主页面。主界面如图2.12所示。主界面有四个功能，分别是检测视频、检测图片、编码人脸信息、录入人脸。检测视频的功能是开启摄像头，利用摄像头进行实时检测，检测图片的主要功能是测试算法能否正常运行。点击“检测图片”按钮后，会弹出选择人脸图片的窗口，选择好人脸图片后，执行算法显示检测信息。编码人脸信息的主要功能是对录入的人脸信息进行编码，把人脸信息存储到数据库中。录入人脸会打开新的窗口，启动Jetson Nano的摄像头进行拍摄功能。



图2.12 主界面

点击第四个按钮会跳转到录入人脸界面，如图2.13所示。首先点击打开摄像头，将人脸完全展示在摄像头中之后，点击拍照，并在输入框内填写人脸信息，最后点击保存，此人脸的信息就存储到数据库中了，返回主页面后，点击“编码人脸信息”，会对保存的所有人脸信息进行编码，以便模型可以准确地检测到人脸。



图2.13 录入人脸界面

2.4.3**MG996R舵机**

MG996R舵0机是一个集成度高的装置，其结构涵盖了外壳、电路板、驱动马达、减速器以及位置检测元件等多个核心组件。MG996R舵机如图2.14所示，其运作机制在于，当接收到来自接收机的指令信号后，电路板上的集成电路会激活驱动马达进行旋转。减速器的作用在于将电机的旋转动力平稳地传递到摆臂上。同时，内置的位置检测元件会实时监测舵机的位置状态，并反馈信号以确保舵机能够精准地到达预设位置。具体来说，这个位置检测元件实质上是一个可变电阻，它会随着舵机的旋转而相应地改变其电阻值。通过检测这一电阻值的变化，本文可以精确地掌握舵机的旋转角度，从而实现精准的控制和定位。

在这一课题中，舵机由Esp8266控制，在检测视频的模式下，当确认身份信息后，Jetson Nano会将确认信息发送给Esp8266，Esp8266收到确认信息后，控制舵机的开启与关闭。舵机的启动结果会通过mqtt协议上传到阿里云，用户可以远程监控门锁的开启情况。



图2.14 MG1996R舵机

2.5 **本章小结**

第二章详细介绍了智能门锁系统的架构设计及算法。介绍了Nvidia Jetson Nano平台，这是一款高性能的人工智能计算设备，适用于部署和运行深度学习模型。上文还讨论了智能门锁系统的工作逻辑，并重点介绍了人脸检测和识别算法，特别是Retinaface和FaceNet算法。这些算法通过多个小卷积核叠加，增强了特征提取能力，提升了识别准确率和泛化能力。本章还阐述了PyQt界面和MG996R舵机的设计，展示了通过PyQt创建项目界面和使用Esp8266控制舵机的过程。

**第三章 Retinaface+FaceNet算法的实现流程**

3.1 **Retinaface+FaceNet算法的实现**

本项目的核心部分就是模型，首先要做的就是对模型进行训练，本文将利用大量的带标签的数据集来训练模型，以使其能够准确地学习人脸特征。在训练过程中，本文会采用一系列的优化算法和技术，例如随机梯度下降、批量归一化和学习率调度等，以最大程度地提升模型的性能和泛化能力。同时，为了有效地训练模型并提高其对人脸的识别能力，本文还将采取一些数据增强技术，如随机裁剪、旋转、缩放等，以增加训练样本的多样性，从而帮助模型更好地泛化到不同的人脸样本上。为了描述整个模型的实现流程，本文进行了对模型模块的介绍和如何用代码实现整个过程。

3.1.1 **Retinaface算法的实现**

RetinaFace的训练过程主要依赖于WIDER FACE数据集，该数据集总共包含了32203张图像，每张图像中标注了393703个人脸区域。这些图像涵盖了人脸在尺度、姿态、表情、遮挡以及光照条件下的多样化变化，确保了模型训练的全面性和泛化能力。为了更系统地评估模型性能，WIDER FACE数据集被精心划分为训练集(占比40%)、验证集(占比10%)和测试集(占比50%)。

基于EdgeBox的检测算法，研究者们对WIDER FACE数据集中的样本进行了难度评估，并将整个数据集分为三个难度级别：容易、中性和困难。这一划分有助于模型在训练过程中逐步适应并克服更具挑战性的样本，从而提升模型在实际应用中的表现。此外，RetinaFace团队进行了额外的标注工作，根据人脸关键点的标注难度，定义了5个人脸质量级别，并标注了5个关键点(包括眼睛中心2个、鼻尖1个、嘴角2个)。他们共标注了84.6k个训练集人脸和18.5k个验证集人脸。标注的样式如图3.1所示。

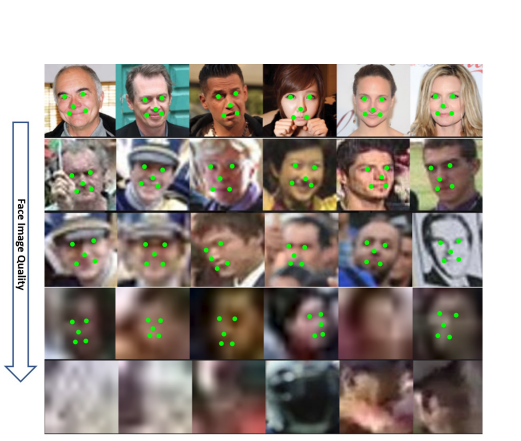


图3.1 WiderFace数据集标注情况

本文在实际训练过程中，主干特征提取网络采用MobileNetV1-0.25，实现的基本代码如图3.2所示，self.stage1第一个阶段，包含一系列卷积层和批量归一化层。这些层被串联在一起以形成第一个阶段的特征提取部分。

几个深度可分离卷积层，其输入通道数和输出通道数在逐步变化，每个卷积层后都跟有批量归一化层。

self.stage2: 第二个阶段，同样包含一系列深度可分离卷积层和批量归一化层，用于进一步提取特征。其中，每个深度可分离卷积层的输入通道数为64，输出通道数为128，步长为2或1。

self.stage3: 第三个阶段，也是一系列深度可分离卷积层和批量归一化层，用于继续提取特征。最后的卷积层将输出特征图的通道数从128增加到256，步长为2或1。

self.avg: 平均池化层，将输入特征图的尺寸调整为(1, 1)，保持通道数不变。

self.fc: 全连接层，将特征映射到1000个类别的输出。

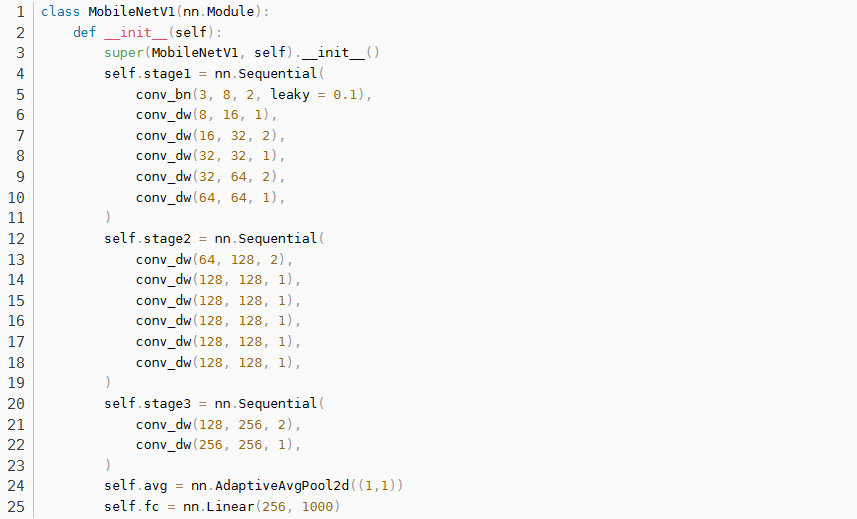


图3.2 MobileNetV1-0.25代码实现

第二步就是FPN特征金字塔的实现，本文对MobileNet的最后三个有效特征层进行FPN的构建，下面是对代码的功能描述：

1. 首先，根据输入的图像数据，将其分别经过三个卷积层(output1、output2、output3)，以提取不同尺度的特征。这三个卷积层分别专注于处理输入图像的不同层次的特征信息。
2. 然后，通过上采样的方式将较低分辨率的特征图(output3)上采样到与中等分辨率的特征图(output2)相同的尺寸，并将它们相加，以获得更丰富的特征表示。这一步是为了让不同尺度的特征图能够在空间上对齐。
3. 接着，再次进行上采样，将得到的融合特征图(output2)上采样到与最高分辨率的特征图(output1)相同的尺寸，并将它们相加，得到更加综合的特征表示。
4. 最后，将得到的三个尺度的特征图以列表的形式返回，这就是模型的输出。

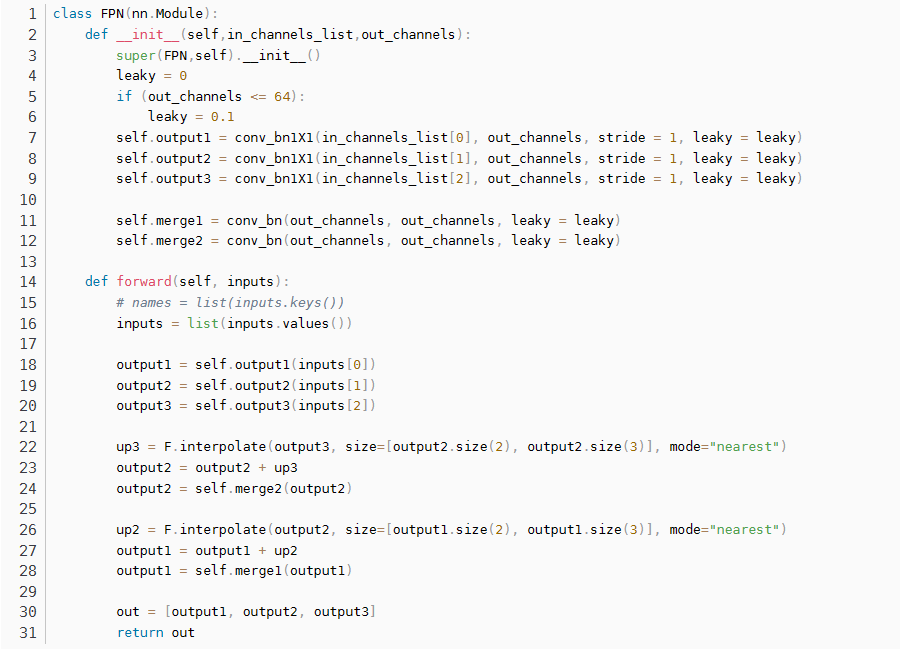


图3.3 FPN特征金字塔代码实现

本文利用FPN(Feature Pyramid Network，FPN)金字塔结构的计算机制，成功提取了三个具有显著表征能力的特征层。为了进一步加强特征提取，本文需要采用SSH模块，来进行下一步的运算。首先，SSH模块根据输入的通道数和输出的通道数来初始化模型。然后，它使用不同尺寸的卷积核(3x3、5x5、7x7)对输入的特征图进行卷积操作。这些卷积核分别用于捕获不同大小的特征信息。对于 5x5 和 7x7 的卷积核，采用了分步骤的方式进行卷积，先进行一次卷积，然后再进行一次卷积。这一步骤可以提升模型的非线性能力，提高特征提取的效果。最后，将三个不同尺寸的卷积操作得到的特征图拼接在一起，形成一个更加丰富和多样化的特征表示。最终，对拼接得到的特征图进行 ReLU 激活函数处理，并将处理后的特征图返回作为模型的输出。

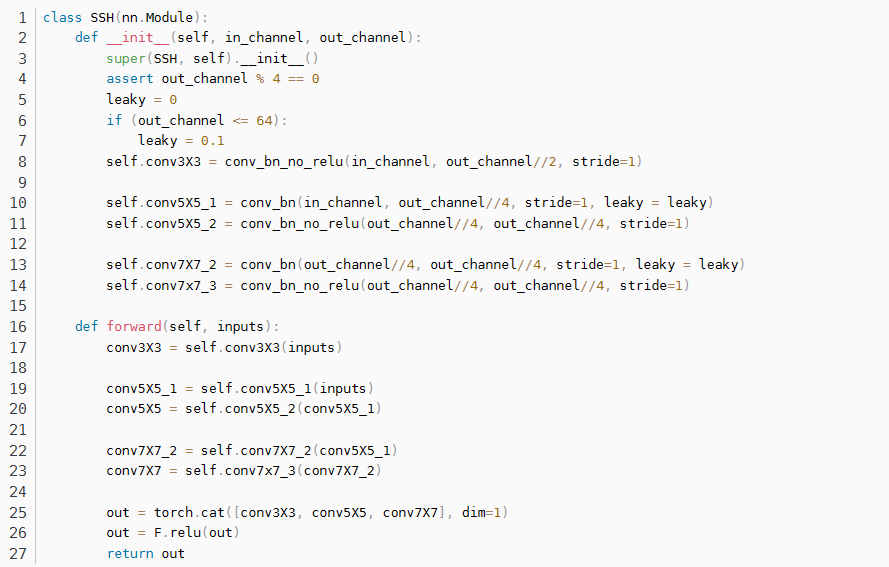


图3.4 SSH模块代码实现

本文借助SSH模块的计算流程，如图3.4所示，成功提取了SSH1、SSH2、SSH3这三个具备显著特征表达能力的层次。下面通过这三个精心提取的特征层来进一步推导和获得最终的预测结果。

Retinaface将预测的结果分成三个部份：类别预测、检测框回归与脸部关键点回归。

判断先验框内物体存在性：首先，本文依据分类预测的结果来确定先验框内是否包含目标物体。为了量化每个先验框内物体的潜在存在性，Retinaface原版采用了softmax分类器。为了实现这一点，本文利用1x1卷积操作将SSH特征层的通道数调整为num\_anchors x 2，其中2代表每个先验框内存在物体或不存在的两种可能性。

先验框的调整与预测框生成：接着，为了从先验框得到更为精确的预测框，本文利用框的回归预测结果来调整先验框的位置和大小。这通常通过修改先验框的四个参数来实现。为了提取这些回归参数，本文同样使用1x1卷积将SSH特征层的通道数转换为num\_anchors x 4，每个先验框对应4个调整参数。

人脸关键点的精确定位：为了精确定位人脸的关键点，如眼睛、鼻子和嘴巴等，本文采用人脸关键点的回归预测结果来微调先验框。考虑到每个关键点需要两个调整参数(如x和y坐标的偏移)，而通常人脸有五个关键点，本文再次利用1x1卷积将SSH特征层的通道数设置为num\_anchors x 10，这样每个先验框就可以对应到其所有关键点的微调参数。

接下来对Retinaface进行搭建，其实现代码如图3.5所示，首先，根据传入的配置参数和阶段初始化模型。根据配置参数选择合适的主干网络。如果配置中指定了使用 MobileNetV1，则选择 MobileNetV1 作为主干网络；如果指定了使用 ResNet50，则选择 ResNet50，在这里本文选择MobileNetV1作为主干网络。设置模型的主体部分，这部分是主干网络的中间层，用于提取图像特征。根据配置参数中的通道数设置不同尺度的特征金字塔网络(FPN)。FPN 会从主体部分提取的特征中生成多尺度的特征图，以便在不同尺度下检测人脸。设置三个分支(ssh1、ssh2、ssh3)，每个分支都是一个 SSH 模块，用于对特征金字塔网络中的特征图进行进一步处理，以提取更加丰富和具体的特征信息。创建用于分类、框回归和人脸关键点回归的头部网络(ClassHead、BboxHead、LandmarkHead)。这些头部网络会从特征金字塔网络中提取的特征中进一步预测人脸的位置、类别和关键点位置。



图3.5 Retinaface代码实现

通过上文所述过程，本文可以得到三个关键的特征层，称为SSH1、SSH2和SSH3。这些特征层可以看作是对原始图像进行了不同程度的细分，将图像划分为不同大小的网格。举例来说，当本文有一个输入图像大小为(640, 640, 3)时：

SSH1的形状是(80, 80, 64)，它将原始图像划分为80x80个网格。

SSH2的形状是(40, 40, 64)，它将原始图像划分为40x40个网格。

SSH3的形状是(20, 20, 64)，它将原始图像划分为20x20个网格。

在Retinaface中，每个划分的网格区域都预设有两个先验框，这些框代表图像中可能的目标区域。Retinaface的预测机制旨在判断每个先验框是否包含人脸，并据此调整包含人脸的先验框，以得到更为准确的预测框和人脸关键点的精确位置。

详细来说，本文利用分类预测的结果来判定每个先验框是否为人脸区域。为实现这一点，本文采用了一个1x1的卷积层来转化SSH特征层的通道数，将其调整为num\_anchors x 2，从而可以表达每个先验框内存在人脸的可能性。通过这一操作，本文能够有效地对图像中的人脸区域进行识别与定位。

框的回归预测结果被用来微调先验框，以确定预测框的位置。为了优化每个先验框的位置和尺寸，本文采用了四个参数。其中，前两个参数负责微调先验框的中心坐标，而后两个参数则用于调整先验框的宽度和高度。为了提取这些微调参数，本文利用了一个1x1的卷积层，该层将SSH特征层的通道数调整为num\_anchors x 4。

在人脸检测任务中，为了更精确地定位人脸的关键点，本文依赖于关键点的回归预测结果对先验框进行微调。具体来说，每个人脸关键点都需要两个参数来精细调整其位置，而通常一个人脸会有五个这样的关键点。为了实现这一过程，本文采用了一个1x1的卷积层，它将SSH特征层的通道数转变为num\_anchors x 10，这样的设计允许本文捕获每个先验框内每个关键点的具体微调信息。通过这种方式，本文能够更准确地确定人脸关键点的精确位置。下图所示就是经过Retinaface的人脸检测结果，并在图像上将检测结果绘制出来，根据检测结果中的关键点坐标，分别绘制圆圈在图像上，以标识人脸的五个关键点。绘制关键点代码如图3.6所示。

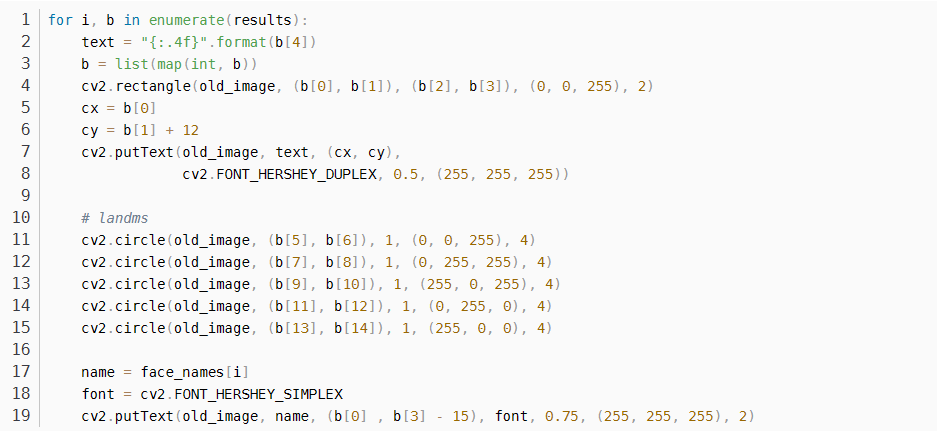


图3.6 绘制5个关键点代码

在原图像上将人脸的关键点绘制完毕后，即可显示图像，如图3.7所示。



图3.7 Retinaface人脸检测结果

3.1.2 **FaceNet算法的实现**

FaceNet也采用MobileNet作为主干特征提取网络，但与上文中Retinaface所采用的有所不同，FaceNet的MobileNet在第一阶段，将输入图像的通道数从 3 增加到 256，并对特征图的尺寸进行降采样，在第二阶段，继续进行深度可分离卷积操作，将输入特征图的通道数增加到 512，对特征图的尺寸进行降采样，第三阶段将输入特征图的通道数增加到 1024，并进行一次降采样，最后全连接层输出维度1024。FaceNet所使用的MobileNetV1如图3.8所示。

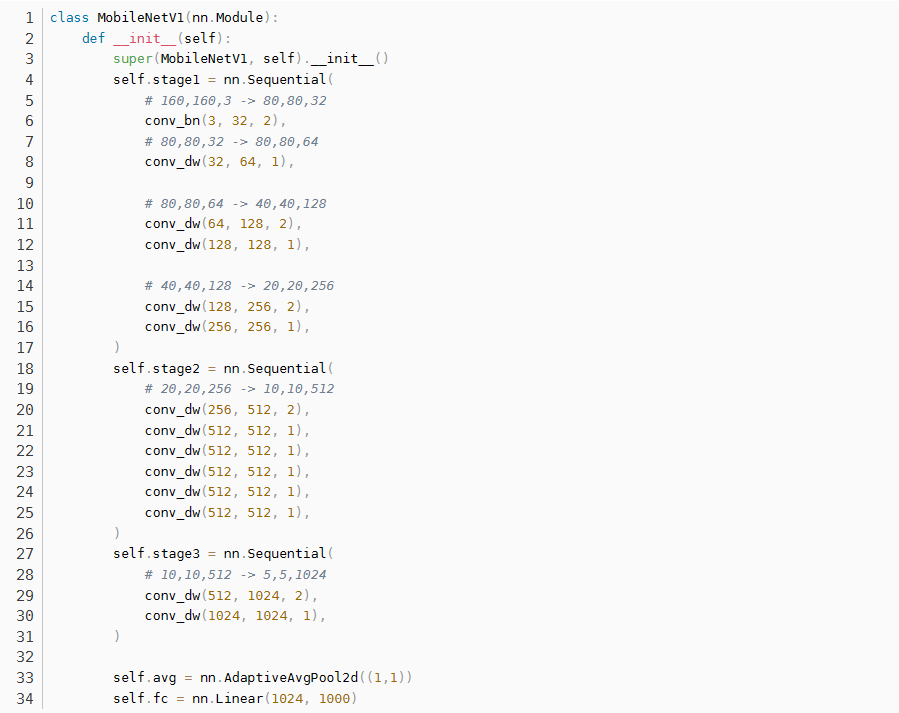


图3.8 MobileNetV1代码实现图

接下来开始对FaceNet的核心部分进行构建，实现的代码如图3.9所示。首先MobileNetV1作为主干网络，随后对主干网络的特征图进行自适应平均池化，将特征图的大小调整为固定大小(1x1)，以便后续处理，为了防止过拟合，本文在全连接层之前加入了 Dropout 层，用于随机地将一部分神经元输出设置为0，而后通过全连接层，将经过 Dropout 层处理的特征向量映射到一个低维度的特征空间并进行批归一化操作，最后通过 L2 范数归一化操作，返回最终的特征表示。

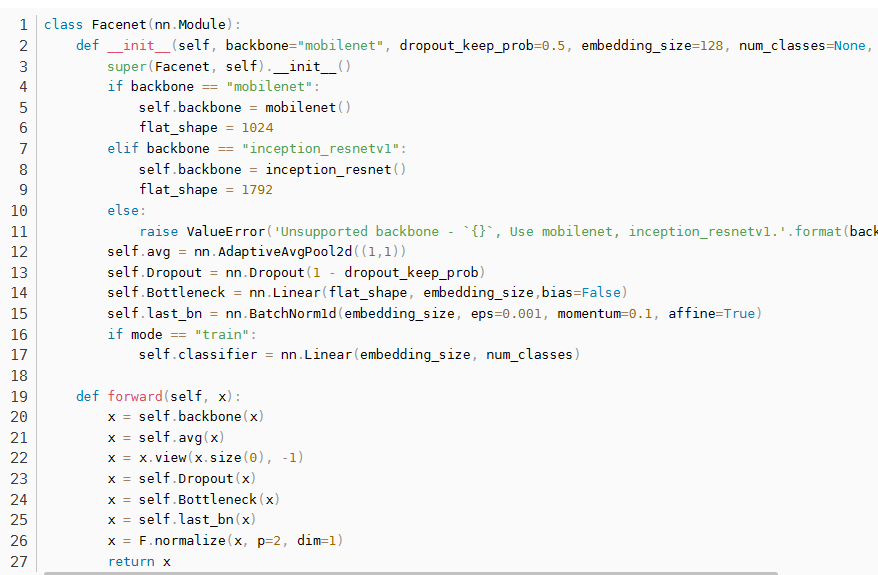


图3.9 FaceNet整体代码实现图

之所以采用FaceNet，是为了将人脸信息进行编码，在智能门锁的程序中，本文通过Retinaface来确定人脸，再通过FaceNet对人脸编码得到长度为128的特征向量，并通过此特征向量与人脸数据库中的特征向量进行比对，确定该人脸信息是否为数据库中所存储的人脸。

3.2 **Retinaface+FaceNet的图片检测模式与视频检测模式的实现**

3.2.1 **数据库的初始化**

项目的目录部分如图3.10所示：

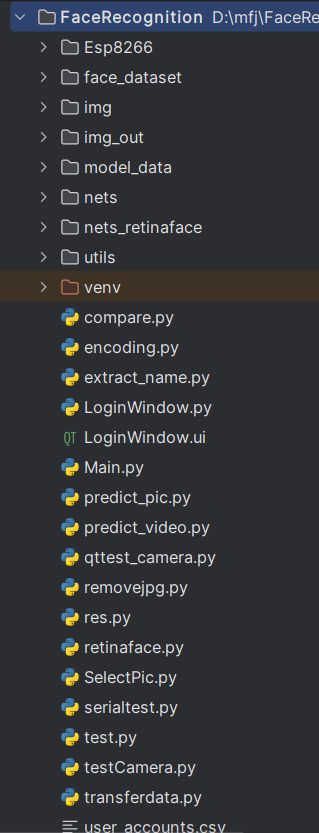


图3.10 项目整体文件目录

face\_dataset文件夹里面是想要识别的人脸，数据库初始化指的是人脸数据库的初始化。数据库初始化的具体执行过程就是：首先，本文遍历数据库中的所有图片。然后，使用 Retinaface 检测每张图片中的人脸位置，并将其截取下来。接着，对获取到的人脸进行对齐处理。随后，利用 Facenet 对人脸进行编码。将所有人脸的编码结果汇总到一个列表中。最后，以 npy 的格式保存。在第6步得到的列表就是已知的人脸特征列表，在之后进行识别的过程中，都需要与其进行人脸比对，只有这样才能确认人脸信息。

3.2.2 **人脸的截取与对齐**

在检测图片时，在图片中人脸的位置不一定是正向的,如图3.11所示：



图3.11 非正向人脸图

如果把人脸正过来，对于人脸的特征提取，可以提高一定的效率。所以本文在检测图片前，先对人脸进行校正，将人脸校正需要选择两个参数，分别是两眼之间的连线相对于水平线的夹角与图片的中心点，夹角图片如图3.12所示。

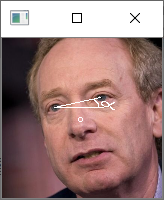


图3.12 旋转角度和旋转中心标注图

通过计算眼睛与水平线之间的倾角。这个角度可以帮助本文确定人脸是否需要旋转，以使得眼睛连线变得水平。计算图像的中心点坐标，以便后续的旋转操作。使用 OpenCV 提供的旋转矩阵函数，根据计算出的角度和中心点坐标，得到一个旋转矩阵。使用 OpenCV 提供的仿射变换函数，根据旋转矩阵对输入的图像进行旋转操作，得到一个新的图像，如图3.13所示。

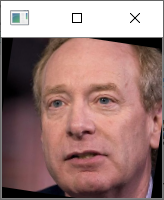


图3.13 经过旋转后的人脸图

3.2.3 **利用FaceNet对人脸进行识别**

将人脸校正后，就可以将图片传入FaceNet模型进行编码，FaceNet进行编码的代码如图3.14所示。首先，遍历了一组包含人脸框、置信度和关键点坐标的信息。其次，对每张人脸图像进行了裁剪，根据人脸关键点坐标对人脸进行了矫正，使得人脸在图像中更加规整和居中。最后，对矫正后的人脸图像进行编码，将其转换成一个长度为128的特征向量，将这个特征向量添加到一个列表中，作为人脸的编码结果。



图3.14 FaceNet编码代码实现

当特征向量被成功提取后，人脸比对流程随即启动，比对代码如图3.15所示。这一比对过程通过循环机制实现，旨在从实时捕获的图像中逐一提取人脸的特征向量。接下来，这些实时提取的特征向量会与数据库中存储的已知人脸特征向量逐一进行对比，并利用算法计算它们之间的相似度距离。在对比过程中，一旦发现两个特征向量之间的距离低于预设的相似度阈值，系统将认定这两张人脸具有一定的匹配度。对于每一个待识别的人脸，系统会进一步确定数据库中与之最为匹配的人脸记录，并再次校验该匹配度是否达到设定的阈值标准。若匹配度满足要求，系统则会将该实时人脸识别为数据库中对应序号的人脸。



图3.15 FaceNet人脸比对代码

在完成比对过程后，最后要在人脸上绘制检测结果，对于每个检测到的人脸，首先从其检测结果中获取置信度，然后将其格式化为字符串并绘制在图像上。接着，从检测结果中提取人脸框的坐标，绘制矩形框在图像上，以突出显示检测到的人脸。根据检测结果中的关键点坐标，分别绘制圆圈在图像上，以标识人脸的五个关键点。如果存在人脸名称(存储在列表 face\_names 中)，则将其绘制在人脸框的上方，以标识检测到的人脸的名称。绘制的代码同上，检测结果示例如图3.16所示。

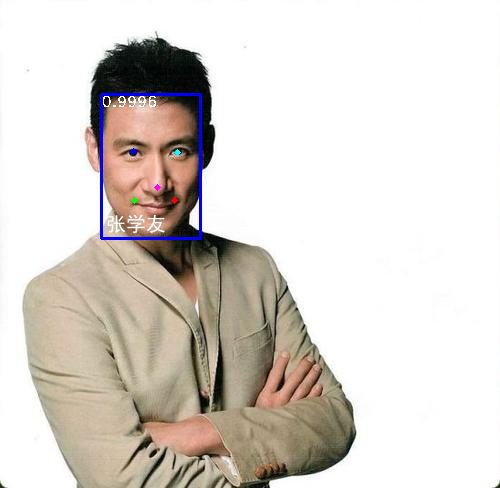


图3.16 Retinaface+FaceNet人脸检测结果

3.2.4 **利用Retinaface+FaceNet实现视频检测**

本文已经实现了Retinaface+FaceNet的图片检测功能，现在对视频检测模式的实现进行描述，首先，连接摄像头设备，程序启动后会打开摄像头，在一个循环中，逐帧读取视频流，如果无法读取，则跳出循环，使用已经训练好的模型进行检测，随后将人脸检测结果与已知的人脸数据库进行比对，判断是否存在已知的人脸，如果是已经人脸则显示人脸信息，如果非已知人脸，就显示“Unknown”，整体实现代码如图3.17所示。



图3.17 Retinaface+FaceNet视频检测代码

3.3 **本章小结**

本章深入研究了Retinaface和FaceNet算法的实现细节，涵盖了从基本原理到具体实现的每一个步骤。首先，在第一节中，本文仔细探讨了如何分别实现Retinaface和FaceNet。本文将逐层地分析这两个模型的结构，并介绍如何在实际代码中实现它们。在第二节中，本文着重讨论了在开始对图片和视频进行检测之前必须进行的预处理步骤。这包括对图像进行尺寸调整、灰度化处理、直方图均衡化等操作，以提高算法的准确性和稳定性。接着，详细介绍了如何校正人脸信息以及将其编码到数据库中的过程。第二节的后半部分聚焦于FaceNet的编码功能和与Retinaface的结合。描述了如何利用FaceNet模型将人脸图像转换为具有唯一特征的向量表示，并介绍如何将这些向量用于人脸识别和验证。此外，还展示了如何将Retinaface和FaceNet结合起来，实现对视频中实时人脸信息的检测和识别。通过调用OpenCV库的摄像头功能，实时地捕获视频流，并对其中的人脸进行检测和识别。

**第四章 PyQt界面的制作**

在先前的章节中，详细地介绍了算法实现的基本流程，而在本章中，将会把这些算法集成到PyQT中，实现用户的界面。

页面一共分为三部分，首先是第一部分：登录、注册，用户在首次启动系统时，需要进行注册，在注册完成后，才能进行登录，进入下一个界面，其次是第二部分：主界面有四个功能：检测图片、检测视频、人脸信息编码、录入人脸，他们的功能分别是检测输入的图片，开启摄像头检测，编码已录入的人脸信息，启动录入界面，最后是第三部分：录入人脸界面，界面的主要功能就是开启摄像头，在输入框内输入人脸信息后，点击拍照，程序就将人脸信息保存到了数据库。

4.1 **登录页面的实现**

首先，使用Qt Designer设计登录和注册的基本页面。在设计完成后，得到一个.ui文件，它包含了界面的布局和控件配置。接下来，利用PyUIC工具可以将.ui文件转换成.py文件，这样就可以在Python中使用这些界面。PyUIC是Qt中的一个工具，它能够将.ui文件转换成相应的Python代码。最后，编写一个Main.py文件，这个文件是整个应用的入口。在Main.py文件中，编写代码来处理界面上的各种交互和功能。例如，验证用户输入的登录信息、注册新用户等等。登录界面和注册界面如图4.1和图4.2所示。



图4.1 PyQT实现的登录界面



图4.2 PyQT实现的注册界面

用户在填写注册信息后，用户的账号和密码会储存到user\_accounts.csv中，方便了管理员对用户信息的管理，实现登录和注册功能的核心代码如图4.5所示。

4.2 **主界面的实现**

在用户登录进入系统后，弹出主界面，主界面是控制整个系统实现的核心界面，如图4.4所示，其包括了四个功能：检测视频、检测图片、编码人脸信息、录入人脸。



图4.3 主界面

首先是检测视频模式的实现流程，点击“检测视频”按钮，按钮绑定了openCam函数，此函数会向模型传递“video”参数，并执行模型，随后Retinaface+FaceNet的视频检测模式会被启动，摄像头开启，开始进行检测，与此同时，检测的结果会以布尔值的形式回传，在后面本文需要用到该布尔值来控制舵机的启动与关闭。

然后是检测图片模式，此功能旨在测试模型能否正常工作，当点击“检测图片”按钮后，会弹出选择图片文件的串口，用户自定义选择face\_data目录下的图片，如果模型正常工作，则可以看到返回的图片有清晰的5个关键点和被标注的姓名，如果模型没有正常工作，则需要检查传入的模型文件是否正确。

接下来就是录入人脸功能，点击“录入人脸”按钮，会弹出一个新界面，用户要现在输入框内输入人脸信息，然后点击“开启摄像头”，拍照完成后，会弹出“The picture have saved !”字样，提示用户拍照完毕。

将人脸信息录入完毕后，即可进行编码，点击“编码人脸信息”会执行上文所展示的编码程序，得到长度为128的特征向量，把每个人脸的特征向量都存储到一个列表。

4.3 **人脸录入界面的实现**

人脸录入界面有三个按钮和一个输入框，输入框内输入人脸的信息，三个按钮控制摄像头的功能，拍照按钮同时执行拍照和保存功能，人脸录入界面如图4.4所示。



图4.4 人脸录入界面



图4.5 实现登录和注册的代码

其主要是通过OpenCV库调用摄像头拍照，并且给拍照按钮绑定文件的保存函数，实现了在拍照的同时录入人脸信息，其核心代码如图4.6所示。



图4.6 人脸录入界面核心代码

4.4 **Esp8266控制MG996R舵机**

通过以上算法的判断，系统首先使用人脸识别算法分析输入的图像或视频流，检测图像中是否存在人脸，并尝试确认这些人脸是否与已知的人脸信息相匹配。如果人脸信息被确认，系统通过串口通信向外部设备发送指令“1”，表示人脸信息已被确认；如果人脸信息没有被确认，则发送指令“0”，表示未确认。串口输出的结果会被ESP8266接收，ESP8266是一种Wi-Fi模块，能够通过Wi-Fi连接到网络，并处理串口通信。当ESP8266接收到指令“1”时，它会发送PWM信号，启动舵机执行特定动作；如果没有收到“1”，舵机不会启动。这个系统通过图像处理和串口通信，实现对人脸信息的检测和控制外部设备的功能。

舵机启动后，会将启动信息返回给阿里云，用户可以远程查看舵机的启动时间，整体的实现代码如图4.7所示，这一功能的实现加强了智能门锁的安全性。



图4.7 Esp8266控制舵机核心代码

根据代码所描述，舵机在启动之后会通过Mqtt通信协议将启动信息“Start！”发送到阿里云，用户可以在物联网云平台实时监测舵机的启动情况，如图4.8所示，在舵机启动的同时，阿里云的设备监控运维模块会收到Esp8266发送的启动消息。

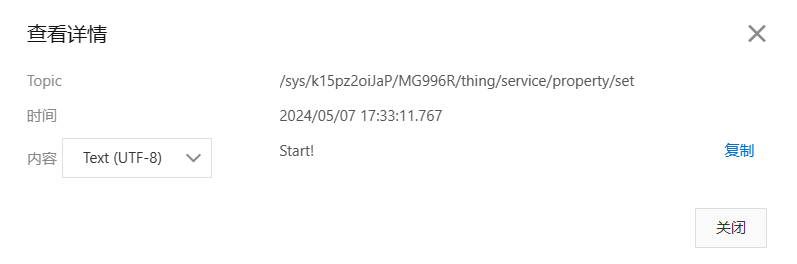


图4.8 舵机启动后发送到阿里云的消息日志

4.5 **登录页面的实现**

第四章集中讲述了PyQt界面的制作和硬件控制部分。具体内容包括登录页面、主界面和人脸录入界面的实现。本章还介绍了如何通过Esp8266控制MG996R舵机，并配置阿里云来接收门锁启动消息。通过这些设计和实现，使智能门锁系统不仅在算法层面上表现优异，还在用户交互和硬件控制上达到了较高的集成和协调。

**第五章 总结与展望**

5.1 **本文总结**

智能门锁技术的发展一直是智能家居领域备受关注的热点之一。在本文中，我们对当前智能门锁技术的发展趋势进行了全面的探讨，并深入分析其在不同应用场景下的应用与前景。同时，我们也对人脸识别技术的国内外研究现状进行了深入剖析，包括其在安全性、准确性以及实用性方面的最新进展，为后续研究提供重要参考。

智能门锁技术的发展一直在不断地推进，从最初的密码锁到指纹识别，再到如今的人脸识别技术，每一次的进步都给人们的生活带来了更多的便利和安全保障。在不同的应用场景下，智能门锁技术也呈现出多样化的发展趋势。比如，在家庭场景下，智能门锁不仅可以提供便捷的开锁方式，还能通过与其他智能设备的联动，实现更智能化的家居管理；而在商业场景下，智能门锁技术则更加注重安全性和稳定性，以满足高强度使用的需求。

人脸识别技术作为智能门锁技术的重要组成部分，其研究和应用也备受关注。国内外在人脸识别技术方面的研究已经取得了长足的进步，不论是在算法的改进上，还是在硬件设备的优化上，都取得了显著的成果。尤其是近年来，深度学习等人工智能技术的发展，为人脸识别技术的提升提供了强大的支持，使得其在安全性、准确性和实用性方面都有了长足的进步。本文采用了Retinaface和FaceNet相结合的形式，实现了人脸的检测和识别，本文首先对Retinaface的原理进行叙述，深入剖析了其如何实现的人脸检测，下一部分对FaceNet展开了研究，通过FaceNet对人脸数据库包含的人脸进行编码，后续再通过摄像头对拍摄到的人脸进行比对。

在系统设计与实现方面，本文详细阐述了人脸识别和检测技术的原理与算法，深入探讨了PyQt界面设计的关键技术与方法，利用PyQt设计登陆和注册界面、主界面、人脸录入界面。最后对舵机的控制程序进行了设计，通过Esp8266接收系统发送的数据，来对舵机进行控制，最后舵机的开启信号会回传到阿里云。通过将理论知识与实际操作相结合，本文提出了一套系统化的解决方案，为智能门锁系统的搭建奠定了坚实的基础。在系统设计中，不仅要考虑到识别的准确性和速度，还要考虑到系统的稳定性和用户体验，这是智能门锁系统设计中需要重点关注的方面。

最终，通过反复的调试和优化，本文成功实现了智能门锁系统，并通过多轮测试与比较，选择出最优模型，确保系统运行速度最快、识别效果最佳。值得注意的是，本课题所研究的智能门锁还需要在活体识别再进入深入的研究，因为平台算力的限制等原因，本课题并没有设计活体识别部分，但对于安全性而言，活体识别在智能门锁系统中具有重要的意义，所以在未来，我们会继续进行研究，在本课题所制作的智能门锁的基础上实现活体识别。整个研究过程不仅提升了智能门锁系统的技术水平，也为相关领域的进一步研究与应用奠定了重要基础。智能门锁技术的发展离不开对技术的不断探索与创新，相信在不久的将来，智能门锁技术将会得到更广泛的应用，为人们的生活带来更多的便利和安全保障。

5.2 **未来展望**

在未来，随着智能科技的不断进步与普及，智能门锁技术将迎来更广阔的发展空间和更深远的影响。首先，可以预见的是智能门锁系统将更加智能化和个性化。未来的智能门锁将不仅仅是通过人脸识别来实现开锁功能，还将整合更多生物特征识别技术，如虹膜识别、指纹识别等，从而提高门锁系统的安全性和准确性。此外，随着人工智能和大数据技术的发展，智能门锁系统还将具备更强大的智能分析和学习能力，能够根据用户的习惯和行为进行个性化的服务和安全管理。

其次，未来的智能门锁系统将更加融入智能家居生态系统，成为智能家居的重要组成部分。智能门锁将与家庭中的其他智能设备实现无缝连接，如智能摄像头、智能灯光、智能家电等，通过智能化的联动和控制，实现更智能、更便捷的家居生活体验。例如，当智能门锁识别到用户回家时，可以自动开启家中的灯光和空调，提高用户的舒适度和便利性。

最后，未来智能门锁技术将在智能化、个性化、智能家居和智能城市建设等方面发挥越来越重要的作用，为人们的生活和城市的发展带来更多便利和可能。

**参考文献**

1. 李浩轩.手背静脉身份识别方法门锁系统研究[J].冶金设备,2023(S1):14-17+117.
2. 贺云飞,甘雨,肖国锐.基于ZigBee的智能门锁系统设计[J].电子设计工程,2023,31(16):6-10.DOI:10.14022/j.issn1674-6236.2023.16.002.
3. 谷月. 虹膜识别“解锁”智能门锁[N]. 中国电子报,2023-04-21(006).DOI:10.28065/n.cnki.ncdzb.2023.000474.
4. J. B. Jin, H. K. Hee, S. S. Soo. Door-Lock System to Detect and Transmit in Real Time according to External Shock Sensitivity[J]. Journal of the Korea Convergence Society, 2018, 9(7): 9-16.
5. P.Jayasri ,C.Pradeepthi,B.Madhavi, et al.SMART DOOR LOCK SYSTEM INTEGRATED WITH RASPBERRY PI USING IoT[J].Journal of Critical Reviews,2020,7(16):2244-2250.
6. Jalalu G ,Srinivasulu S R ,T. N , et al.IoT-Enhanced Smart Door Locking System with Security[J].SN Computer Science,2023,4(2):
7. Y. Sun, X. Wang and X. Tang, "Deep Learning Face Representation from Predicting 10,000 Classes," 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 2014, pp. 1891-1898, doi: 10.1109/CVPR.2014.244.
8. X. Deng and P. L. Dragotti, "Deep Convolutional Neural Network for Multi-Modal Image Restoration and Fusion," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 10, pp. 3333-3348, 1 Oct. 2021, doi: 10.1109/TPAMI.2020.2984244.
9. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S.E., Fu, C., & Berg, A.C. (2015). SSD: Single Shot MultiBox Detector. European Conference on Computer Vision.
10. J. Deng, J. Guo, N. Xue and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019, pp. 4685-4694, doi: 10.1109/CVPR.2019.00482.
11. J. Deng, J. Guo, E. Ververas, I. Kotsia and S. Zafeiriou, "RetinaFace: Single-Shot Multi-Level Face Localisation in the Wild," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 2020, pp. 5202-5211, doi: 10.1109/CVPR42600.2020.00525.
12. Smith, John. Deep Learning Systems: Algorithms, Compilers, and Processors for Large-Scale Production. New York: Morgan & Claypool, 2020.
13. 姚冲,邓在辉.基于Jetson,Nano和改进YOLOv7算法的安全帽佩戴目标检测技术分析[J].电子技术,2024,53(01):304-307.
14. 叶光泽,童宣科,侯保冀.基于Jetson Nano的人脸口罩识别系统设计[J].电子产品世界,2024,31(01):44-47.
15. 李云鹏,席志红.基于RetinaFace与FaceNet的动态人脸识别系统设计[J/OL].电子科技,1-8[2024-05-12].https://doi.org/10.16180/j.cnki.issn1007-7820.2024.12.012.
16. 随玉腾,阎志远,戴琳琳,等.基于RetinaFace的人脸多属性检测算法研究[J].铁路计算机应用,2021,30(03):1-4.
17. 陶文玲,侯冬青.PyQt5与Qt设计师在GUI开发中的应用[J].湖南邮电职业技术学院学报,2020,19(01):19-21.
18. 肖文鹏.用PyQt进行Python下的GUI开发[J].中文信息,2002,(07):73-75.
19. 黄率.基于PYQT5的AI图像识别工具[J].现代工业经济和信息化,2023,13(01):90-91+94.DOI:10.16525/j.cnki.14-1362/n.2023.01.033.
20. 范兴隆.ESP8266在智能家居监控系统中的应用[J].单片机与嵌入式系统应用,2016,16(09):52-56.
21. 张萍.基于ESP8266和OneNET云平台的远程报警系统[J].单片机与嵌入式系统应用,2017,17(12):64-67.
22. 于彬.基于人脸识别技术的楼宇门禁系统的研制[D].黑龙江大学,2021.DOI:10.27123/d.cnki.ghlju.2021.000263.
23. 王传传,高婕.基于改进FaceNet算法的人脸智能识别方法[J].工业控制计算机,2024,37(02):126-128.
24. 董涛,周旭辉,张浩.基于FaceNet的智能自习室的设计与实现[J].电子产品世界,2023,30(06):19-21+28.
25. 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.
26. 张萌.智能门锁产品功能发展的一些新看点[J].家用电器,2023,(11):70-71.
27. 李鑫,王树强,李猛,等.基于Jetson Nano的视觉识别搬运智能车[J].机电工程技术,2023,52(11):186-190.
28. 黄率.基于PYQT5的AI图像识别工具[J].现代工业经济和信息化,2023,13(01):90-91+94.DOI:10.16525/j.cnki.14-1362/n.2023.01.033.
29. 江健,张琪,王财勇.基于OpenCV和PyQt的数字图像处理综合实验平台开发[J].电脑知识与技术,2023,19(24):6-8+13.DOI:10.14004/j.cnki.ckt.2023.1253.
30. 张焱鑫.基于PyQt5和百度AI开放平台的物体图像识别界面系统的设计与实现[J].软件,2021,42(09):58-60+134.