

## 摘 要

中文摘要内容。

**关键词：**关键词 1； 关键词 2

## **Abstract**

English abstract contents.

**Keywords:** keyword1; keyword2

# 目 录

1 绪论 .....	1
1.1 研究背景 .....	1
1.2 国内外研究现状 .....	1
1.3 研究目的与意义 .....	3
1.4 研究内容 .....	3
2 系统相关技术介绍 .....	5
2.1 Django 介绍 .....	5
2.2 Bootstrap 介绍 .....	5
2.3 PostgreSQL 介绍 .....	5
2.4 MTCNN 介绍 .....	6
2.5 Facenet 介绍 .....	6
3 系统分析 .....	7
3.1 可行性分析 .....	7
3.2 业务需求 .....	7
3.2.1 账号注册与登录 .....	7
3.2.2 工厂信息管理 .....	8
3.2.3 人脸识别签到 .....	8
4 系统设计 .....	10
4.1 系统总体功能设计 .....	10
4.2 数据库设计 .....	11
4.3 员工考勤功能设计 .....	12
4.3.1 构造数据集 .....	12
4.3.2 模型设计 .....	13
4.3.3 TripletLoss 损失函数 .....	14
4.3.4 模型评估性能指标 .....	14
参考文献 .....	15

# 1 绪论

## 1.1 研究背景

随着信息技术的高速发展，越来越多的行业领域开始与这一学科开始交叉挂钩，并且结合出了非常多样的理论知识与应用场景。关于信息管理系统，不管各行各业，无论是高校学籍档案、医疗设备信息还是农业大棚温度测控的场景，都离不开简单高效信息管理系统应用。而且在一些特定的应用场景，不同的信息管理系统所面向的功能也不尽相同，例如有些系统需要面对超大批量用户访问的高并发场景，而有些系统则需要在信息的安全性方面做进一步加强。然而在一些中小型企业工厂的信息管理方式中，例如对工厂员工的信息、客户信息、供应商信息、产品信息以及原材料信息的管理，工厂管理人员大部分往往都是通过手工的方式对信息进行管理，即使人们会用到一些办公软件，例如 Word、Excel 等；工厂与客户以及供应商的合作方式还都是以电话、微信等联系方式进行沟通交流，但这种方式毫无疑问是离散的、不统一的，并且信息的传递往往都并不是实时的，所以针对中小型工厂企业，设计实现一个通用的、简单易管理的信息管理系统往往有着很大的需求。

人脸识别 (Face Recognition) 作为一种主流生物识别技术，已经充分的渗透各个领域之中，例如军事、金融、公共安全和日常生活等。并且人脸识别在 CVPR (Computer Vision and Pattern Recognition) 社区一直是一个火热并且存在长久的研究话题。在上个世纪末，基于特征的人脸识别技术的里程碑出现了，通过特定的分布假设，派生出了一些综合的数学方法对数据的低维表示。直到二十一世纪初，一个众所周知的问题便是这些理论上似乎可信的综合数学方法无法处理的，那就是不可控的面部改变问题。这些问题在<sup>[1]</sup>中提到的人脸识别的四个关键时期中，前三个时期尽管解决人脸识别中的某些面部变化问题，例如姿势变化、光线亮度变化、面部表情和伪装打扮等，但是解决这些问题的方法往往都是比较单一，而且对于识别准确性的提升都是非常有限的。然而这一切都在 2012 年 AlexNet 使用了深度学习技术赢下了 ImageNet 图像分类竞赛的时候发生了巨大的改变。深度学习使用诸如卷积神经网络模型，将多层处理单元进行级联，对输入数据的特征进行提取和变换，从而得到输入数据的多层抽象表示。通过这些多层次的抽象表示，对于人脸姿势、面部表情和光照强度都有着非常强大的不变特性。并且人脸识别技术随着深度学习技术一同飞速发展，直到今天，各种各样的用于人脸识别系统中各个模块的网络模型，都有着超越人类表现的性能。

## 1.2 国内外研究现状

在 2012 年，随着 AlexNet<sup>[2]</sup>在 ImageNet 图像分类的竞赛上以非常优越的性能超越了当时的所有模型的结果，人们开始考虑将用于图像分类的网络模型作为骨干结构来构造人脸识别系统，例如 Deepface<sup>[3]</sup>。到了 2015 年，由 Google 提出的一个

22 层的 GoogleNet<sup>[4]</sup>网络模型，人们开始将网络模型变得越来越“深”，深度学习这一技术开始受到越来越多人们研究。同年，将 GoogNet 作为主干结构，Goole 再次提出了一个网络模型 Facenet<sup>[5]</sup>用于构造人脸识别系统，该网络模型提出了一种名为 TripletLoss 的损失函数，此系统可以直接学习将输入的人脸图像映射到欧氏距离空间中，相同身份的图像有着更近的距离，不同身份的图像则有着更远的距离。Facenet 将输入图像映射到一个 128 维的 embedding 空间中，用于比较不同人脸图像的相似度。该系统在 Labeled Faces in the Wild (LFW) 数据集上实现了 99.63% 的准确率，在 On YouTube Faces DB 数据集上实现了 95.12% 的准确率，在当时有着最先进的性能。

随着网络模型变得越来越深，很多大型的网络模型有着上百万的参数用来训练优化，人们发现以传统的损失函数作为对用于人脸识别网络模型的训练变得越来越难以收敛。所以，人们开始将研究重点放在各种训练技巧上，例如设计各式各样的损失函数用于改变训练方式并且加快收敛速度。上文提到的 TripletLoss 损失函数是作为基于欧氏距离的一种，另一种损失函数的类型则是基于角度和余弦边距的损失函数，不同于将人脸图像映射到欧式空间中，第二种损失函数用于将不同身份的人脸图像，即学到的特征，变得更加具有分辨性，也就是增加它们的角度或是余弦距离，作为第二种损失函数类型的代表是 Arcface<sup>[6]</sup>提出的一种带有额外角度边距的损失函数，基于此损失函数构造的人脸识别系统，也实现了最先进的识别性能。

随着越来越多的网络模型被提出，人脸识别的准确率也一步步上升。但是，通常实现最先进性能的模型，往往都是由互联网巨头提出的，例如 Google、Facebook 等，他们用于训练网络模型的数据集都是私人的脸数据集，所以这使得很多研究人员无法准确的复现他们所报道的性能，这给学术界的研究人员带来了很大的挑战。为了解决这一问题，CASIA-Webface<sup>[7]</sup>首次提出了广泛使用的公开人脸训练数据集，此数据集有 10,575 位收集自互联网的人物，共 494,414 张人脸图像。然而，CASIA-Webface 数据集相对较小的数据量和人物数量并不能反映出深度学习的很多高级特性。所以，VGGFace2<sup>[8]</sup>数据集便被提出了。此数据集有大约三百三十一万人脸图像分为 9131 个类别，每一个类别代表了一位身份人物。可以毫不夸张的说，人脸数据集的发展过程在很大程度上引领了人脸识别研究的方向。一个足够大的数据集，包含不同的人脸姿势、光线强度、面部表情、不同种族、不同年龄以及性别等，可以使得网络模型有着非常强大的泛化能力。

在各种训练数据集训练之后，需要可以用于公平比较模型以及更好地反应模型的泛化能力的评估数据集，学术界提出了很多评估测试数据集，例如 Labeled Faces in the Wild (LFW<sup>[9]</sup>) 数据集。LFW 数据集包含了标注的人脸图像，涵盖了日常生活中各种各样可能遇到的典型环境，并且还展示了在各种自然可变性的因素，例如姿势、光照、种族、配饰、遮挡和背景等。之后，由于新冠疫情在 2019 年底在全国范围内全面爆发人们出行必须佩戴口罩。武汉大学国家多媒体软件工程技术研究中心在

2020 年 3 月便公布了一个用于训练佩戴口罩的人脸训练数据集 Real-World Masked Face Dataset (RMFD<sup>[10]</sup>)。同年 8 月, Aqeel Anwar 和 Arijit Raychowdhury 等人开源了一个用于模拟佩戴口罩的工具 MaskTheFace<sup>[11]</sup>用来对现有的人脸数据集进行模拟佩戴口罩用来训练佩戴口罩之后的人脸识别系统以及提出了用于评估佩戴口罩的人脸识别模型的测试集 MFR2。

### 1.3 研究目的与意义

由于新冠疫情在全国范围内的影响,这使得人们的生活方式发生了全面的改变。上班人员开始居家办公,学生也开始在家上网课;这对线下实体的一些销售实体行业产生了巨大的冲击,由于新冠疫情具有强烈传染性,关闭了一些公共场所的商业店铺,社会生产方面全面停止生产。在我国政府以及中国健委会的管理控制下,新冠疫情在我国及时得到了有效控制。尽管如此,人们的日常出行方式须佩戴医用外科口罩。在此期间,以往的生物识别的认证方式并无法有效的防止病毒的入侵,例如,一些场所的通行方式是通过触摸电子屏幕等物体进行生物识别认证,如果没有及时做好清洁消毒工作,若之后触摸到了眼睛、鼻子或嘴巴等身体部位,也使得非常大的几率感染新冠病毒。所以,很多的出行场所将认证方式改为人脸识别,可以实现零触摸的生物认证,然而,在疫情还没有得到完全消除的背景下,通常这些场所都是人口比较密集的公共场所,所以人们在通往关口的时候还要佩戴口罩来有效抵挡病毒的入侵,但是由于佩戴了口罩将人脸进行了遮挡,这使得一些人脸识别的系统出现了失效问题,例如识别准确率下降或无法识别等问题,这给人脸识别领域带来了很大的挑战。所以研究人员要开始探索新的网络架构、损失函数的设计或对人脸进行遮挡的训练测试数据集等,来处理佩戴口罩带来的人脸识别失效的问题。

并且尽管在应对新冠疫情的时候,已经有了一套完整且系统的控制处理方式,这也引起了很多的线下实体生产产业的反思,人们开始考虑将一些业务逻辑结合信息技术在互联网平台上进行业务交流以及信息管理等。特别是对于一些中小型的生产工厂,工厂管理人员对员工信息的管理、客户信息、订单、生产原材料和供应商信息等管理方式往往都还是以半手工半信息技术的方式对工厂进行管理。并且对于一些生产工厂,工厂的安防问题也有着很大的隐患,所以结合深度学习技术,整合一个全面系统的工厂管理平台,既可以实现佩戴口罩的人脸识别以及处理安防问题,例如生产员工是否正确穿戴安全帽、防护服等,可以有效的提升工厂企业的生产效率。

### 1.4 研究内容

使用 Django 框架开发一个智慧工厂管理站点,前端页面使用 Bootstrap 框架设计,将数据存储到 PostgreSQL 数据库,对交易记录、员工信息、客户信息、供应商、产品和原材料等信息进行管理。并且结合深度学习技术,在 Facenet 网络模型的预训练模型之上,基于 CASIA-Webface 人脸数据集使用 MaskTheFace 工具对数据集中的

人脸进行检测，并且将模板中的口罩对人脸进行遮罩，将口罩“佩戴”到人脸图像中，构造一个由 10,575 位人物、494,414 张模拟佩戴口罩的人脸图像数据集进行微调训练，将训练好的模型整合到工厂管理系统中，可以用来作为车间出入、员工考勤打卡功能。

## 2 系统相关技术介绍

### 2.1 Django 介绍

Django 是由 Python 语言编写的开源免费的 web 应用程序的框架，可以在很短的时间内完成从概念到启动的 web 应用的开发。使用 Django 可以只关注于应用的开发无须重造多余的轮子。Django 具有以下特点：

1. 快速性：Django 从设计之初的目的就是为了使得开发人员可以快速地从应用概念到部署启动的过程。
2. 完整性：Django 包含了一系列的额外的工具来处理常见的 web 开发任务。例如用户认证、内容管理、站点地图和 RSS 订阅等，这些工具都是开箱即用。
3. 安全性：Django 的安全性可以帮助开发人员避免常见的安全漏洞，例如 SQL 注入、跨站脚本、跨站请求伪造和点击劫持等。它的用户认证系统提供了一个安全的方式来管理用户账户和密码。
4. 可扩展性：一些复杂的站点使用了 Django 的功能来迅速并且灵活的处理繁重的需求。
5. 灵活性：很多公司、组织和政府使用 Django 构造了各式各样的应用，例如内容管理系统、社交网络和科学计算平台等。

### 2.2 Bootstrap 介绍

Bootstrap 是一个免费开源的 CSS 框架，针对响应式、移动优先的前端 web 开发。它包含 HTML、CSS 和基于 JavaScript 的设计模板用来排版、表单、按钮、导航和其它界面部分等。Bootstrap 只关注于简化 web 页面的开发，它最初用于 web 项目的目的是使用它的颜色选择、尺寸、字体和布局等。因此，开发人员的主要任务就是找到自己满意的色彩样式，并将其添加到自己的 web 项目中，提供了 HTML 元素的基本样式定义；除此之外，还可以利用 CSS 的类来进一步定义不同组件的样式。

### 2.3 PostgreSQL 介绍

PostgreSQL 是一个免费开源的关系型数据库，强调以高扩展性和 SQL 合规性为特点。Postgres 以 ACID 为特性的事务管理、自动更新视图、物化视图、触发器、外键和存储过程为特性。它的设计主要用来处理一系列从单台机器到数据仓库或用很多并发用户访问的 web 站点等的工作量。Postgres 具有丰富的数据类型，除了最基本的数据类型以外，还包括结构化的数据类型、文档数据类型和几何类型，还可以自定义复杂的数据类型；并且还能处理并发场景并且有着优越的性能。



## 2.4 MTCNN 介绍

人脸检测和人脸对齐在一些不受控制的条件下有着各种各样的挑战，例如人脸姿势、照明情况和遮挡情况，然而，随着深度学习技术的飞速发展，人们可以利用端到端 (end-to-end) 框架，使用深度卷积网络模型，将这些问题一次性解决。Multi-Task Convolutional Neural Network (MTCNN<sup>[12]</sup>) 是一种深度层叠的多任务框架，可以利用它们之间的相关性提升模型的表现性能。MTCNN 引入了一种层叠结构，由三个阶段的深度卷积神经网络构成，以由粗到细的方式对人脸进行检测和人脸关键点定位。在训练学习过程，提出了一种新型的在线硬抽样挖掘策略 (online hard sample mining strategy)，可以在无手工样本选择的条件下自动地提升表现性能。

## 2.5 Facenet 介绍

Arcface<sup>[5]</sup>是 Google 团队在 2015 年 CVPR 会议上提出的人脸识别网络模型，该网络模型可以直接学习人脸图像到欧氏空间之间的映射，其中具有相似的人脸有着相近的距离，不同的人脸有着较远的距离。当这个空间被训练得到之后，人脸识别的任务就可以轻松的在 Facenet 系统上进行实现。该模型摒弃了之前深度学习的主流方法——对网络模型中间的瓶颈层 (bottleneck layer) 进行训练，而是通过深度卷积网络对输入图像处理得到的 embedding 向量进行训练优化。改变了训练目标，也需要对损失函数进行调整。Facenet 中使用的是一种基于粗略对齐的匹配、不匹配三元组的损失函数，称之为 triplet loss，在训练集产生三元组的方法叫做在线的三元组挖掘法 (online triplet mining method)。使用这种方法可以非常高效的对人脸图像进行表示，可以仅仅使用 128 维的人脸 embedding 向量在 LFW 测试数据集上实现了 99.63% 的准确率。

## 3 系统分析

### 3.1 可行性分析

本系统使用 Django 后端开发框架进行开发，开发环境在 Ubuntu20.04 操作系统下进行开发，基于 Facenet 的网络模型在 Kaggle 社区中提供的云计算平台进行训练，Kaggle 社区为每个用户每周免费提供 36 个小时的加速计算平台，GPU 硬件型号为 Tesla P100，训练神经网络模型的框架使用 PyTorch，以上所涉及到的编程框架以及开发环境都是开源的，所以在开发阶段并不需要投入过多的经济费用开支，而在开发完成之后的部署阶段，该系统即可选择云服务商提供的云主机进行部署，也可选择使用部署在其它服务器之上的系统。即使使用云服务商提供的云主机的方式进行部署，维护与租用费用都是完全可以接受的。从市场需求的角度来讲，现阶段大部分的中小型工厂企业所使用的的信息管理方式大多都为半手工半计算存储的方式进行管理，使用本系统既可以解决手工管理信息的任务量大，也可以高效地对信息进行统一组织，并且将数据存储到云主机上，也可以有效防止信息丢失的问题。

### 3.2 业务需求

#### 3.2.1 账号注册与登录

在用户访问首页时，首先在访问请求中查询用户是否已经登录账号，之后根据登录与否，在页面顶部导航栏中显示欢迎用语或者显示登录按钮供用户登录。在注册页面，在用户填写注册信息之后，首先在用户终端使用 JavaScript 检测当前输入的信息格式是否正确，通过验证后发送请求到服务后端，再根据填写的信息进行进一步验证，例如验证电子邮件是否已经存在或组织名称是否冲突等，之后在数据库的用户表插入记录，存储用户信息，当注册成功之后，自动跳转到登录页面，之后登录流程如图3-1所示。

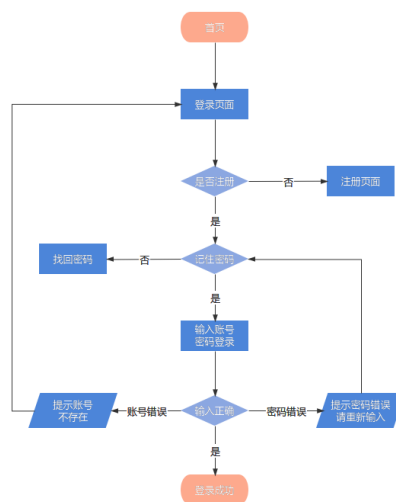


图 3-1 用户登录流程图

### 3.2.2 工厂信息管理

管理员对工厂信息进行管理时，包括交易记录、员工信息、客户信息、供应商信息、产品信息和原材料信息等。其中对员工信息进行管理时，除了员工的个人信息以外，还需要对指定员工进行下达工作任务，例如指定制作的产品、使用的原材料和生产重量等；还包括对员工的薪资进行支付等操作。在涉及到客户信息管理，除了客户个人信息之外，还需要包括对指定客户根据产品以及重量信息的下单功能进行设计。还需根据客户的订单状态，分别查看已送达的订单和未送达的订单，对订单进行管理。在对原材料信息进行管理时，还需要添加根据供应商来购入指定的原材料，由管理人员录入需要购入的价格以及重量等信息。在对员工进行考勤时，使用人脸识别对员工识别并且进行签到，将信息存储到签到表。系统的用例图如图3-2所示。

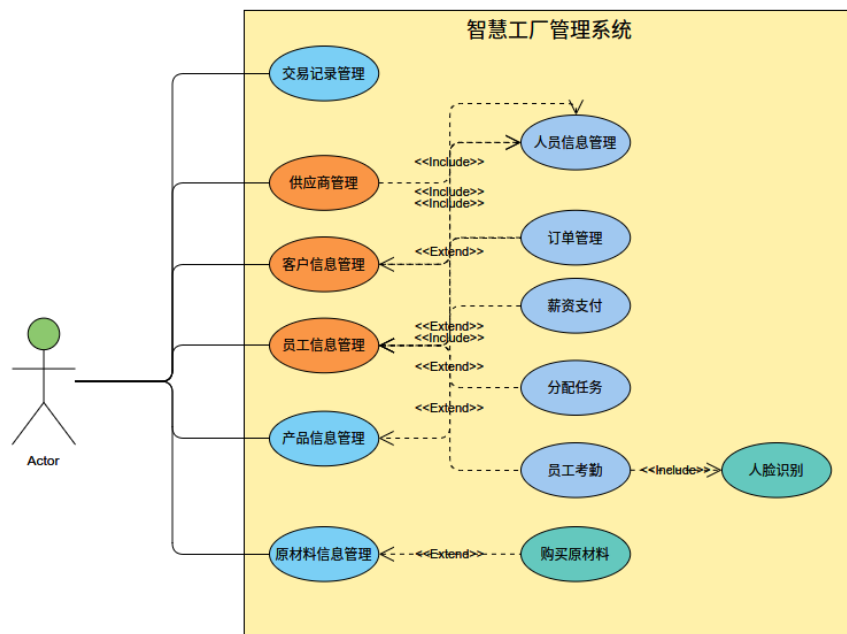


图 3-2 系统用例图

### 3.2.3 人脸识别签到

管理员对员工进行考核时，使用人脸识别对输入图像进行扫描，将根据人脸识别模块的输出对应的员工信息进行签到记录，将记录插入到员工签到表。对于人脸识别系统<sup>[1]</sup>，有 3 个不可或缺的组成部分如图3-3所示。

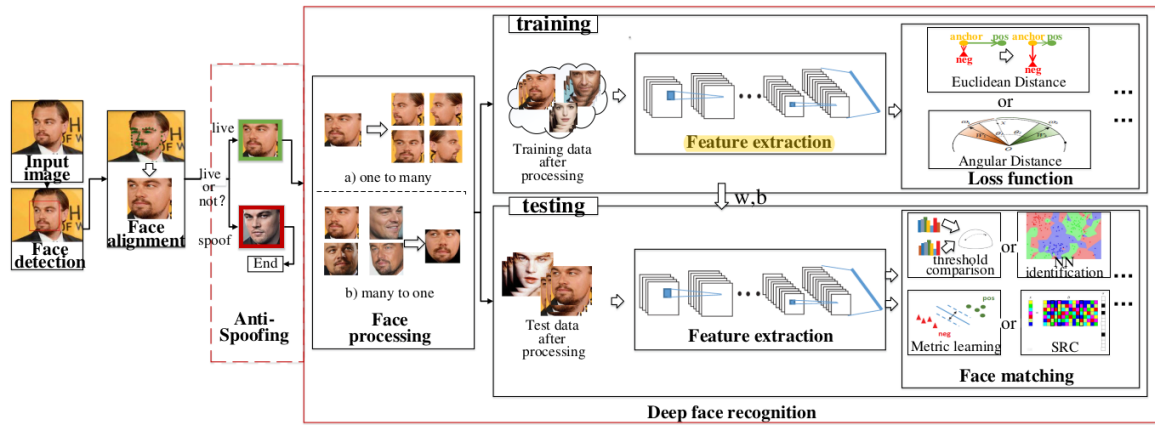


图 3-3 人脸识别系统组成部分

首先，使用人脸检测模块对输入的图像或视频进行人脸检测，返回当前输入的图像或视频是否存在人脸。之后使用人脸对齐模块，对输入图像中的人脸进行关键点检测，将图像中的人脸部分对齐规范到标准坐标中。将人脸对齐模块的输出图像输入到特征提取模块，通常为深度卷积神经网络进行特征提取。将提取出的特征输入到最后的人脸匹配模块，用于将当前输入的图像与人脸数据库中的图像特征进行匹配比较，通常有阈值比较、K 近邻识别、指标学习和 SRC 等。

## 4 系统设计

### 4.1 系统总体功能设计

本系统的整体架构采用 Django 的 MVT 三层设计模式，其中 M 表示 Model 层、V 表示 View 层以及 T 表示 Template 层。从工厂信息管理着手，如图4-4所示，首先对系统总体功能进行设计，选择开发环境以及数据库，之后对数据层进行功能设计，例如事务管理以及读写数据库等，然后设计业务层的业务逻辑，例如交易信息管理、员工信息管理和人脸识别签到等，之后使用模板渲染对前端页面进行后端数据的渲染，最后向用户展示，系统的总体功能模块如图4-5所示。

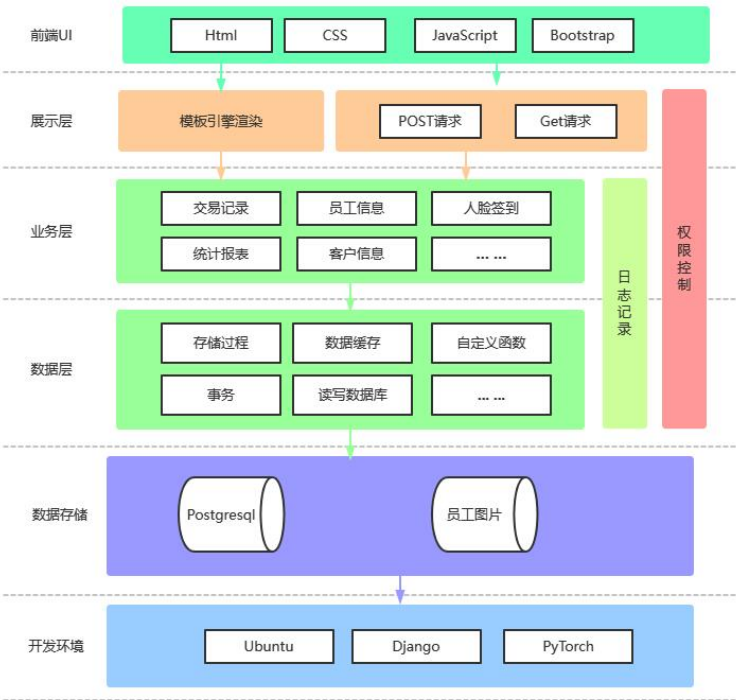


图 4-4 系统架构图

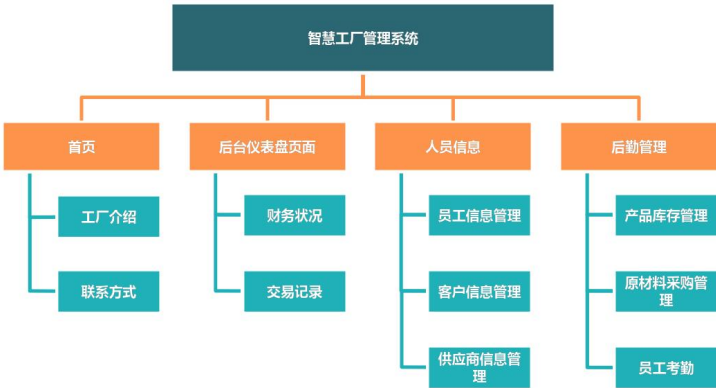


图 4-5 总体功能模块图

## 4.2 数据库设计

在工厂管理中,首先要对三种不同的实体人物进行抽象,有员工信息、客户信息以及供应商信息。分别对三种实体信息进行抽象抽取,设计数据库关系,如表4-1所示,给出了员工信息的关系表。

表 4-1 员工信息设计表

字段名	数据类型	默认值	描述信息
name	CharField	NULL	员工姓名
basicSalary	DecimalField	0	基本工资
bonus	DecimalField	0	奖金
total	DecimalField	0	工资共计
isPaid	BooleanField	False	是否支付薪资
lastSalary	DateField	NULL	最后一次工资更新日期
designation	CharField	其他	职称
address	CharField	NULL	员工地址
phone	CharField	NULL	电话号码
dob	DateField	NULL	出生日期
doj	DateField	NULL	入职日期
gender	IntegerField	2	性别

对后勤信息管理部分进行抽象设计,有产品信息、原材料信息以及员工考勤的签到表。分别对产品和原材料的实体信息进行抽象并设计数据库关系表,如表4-2所示,给出了产品信息的关系表。

表 4-2 产品信息设计表

字段名	数据类型	默认值	描述信息
name	CharField	NULL	产品名称
cost	DecimalField	NULL	产品成本
wages	DecimalField	NULL	产品利润
weight	DecimalField	NULL	产品重量

之后将所有独立的实体表进行定义基数关系,其中包含在客户对某个产品进行下单时,定义客户表和产品表之间的多对多关系的订单表;在员工生产某一个产品时就会有多对多关系的工作表;供应商对某一种原材料进行供应时的多对多的供应表等,如图4-6所示,展示本系统关键实体的 E-R 关系图。

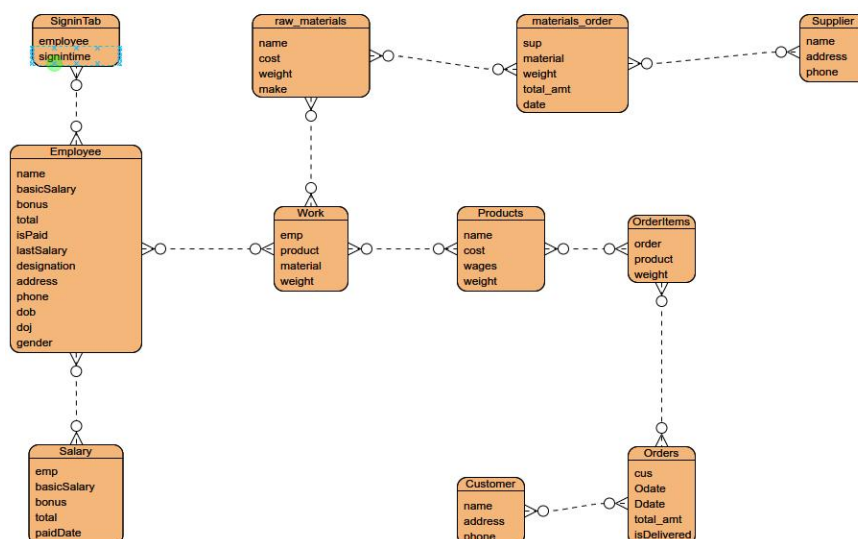


图 4-6 系统关键实体 E-R 图

### 4.3 员工考勤功能设计

本系统结合深度学习人脸识别技术，使用 Facenet 网络模型进行训练，来实现对员工考勤签到功能。进入到员工考勤页面，首先向管理员展示最近的员工签到表，包含员工姓名以及签到时间。进行人脸签到时，计算人脸图像的 embedding 编码，与本地员工人脸图像数据库进行比对，得到两者最小距离的员工，并且检查是否小于阈值，通过后便向签到表添加记录。

#### 4.3.1 构造数据集

由于新冠疫情的影响，人们在日常出行的时候佩戴口罩是不可避免的。由于佩戴口罩对人脸进行了遮挡，所以在一些基于人脸识别的传统系统受到了很多挑战。所以本系统选择基于用于训练人脸识别的 CASIA-Webface 的人脸数据集基础之上，使用 MaskTheFace 工具对数据集进行构造，构造一个模拟佩戴口罩人脸数据集用于训练。CASIA-Webface 由 10,575 位人物共 494,414 张人脸图像构成，部分人脸图像如图4-7所示。



图 4-7 CASIA-Webface 人脸数据集部分图像



训练完神经网络模型之后，使用 LFW 数据集和 MFR2 数据集进行测试评估模型的性能，参考模型对人脸识别的准确率。其中 LFW 是现实世界的涵盖各种自然可变因素的人脸数据集，由 5,749 位人物共 13,233 张人脸图像，其中 1,680 位人物有 2 张以上的人脸图像，部分人脸图像如图4-8所示。MFR2 是现实世界的佩戴口罩的人脸数据集，由 53 位名人和政治家共 269 张佩戴口罩的人脸图像构成，部分人脸图像如图4-9所示。



图 4-8 LFW 人脸数据集部分图像



图 4-9 MFR2 人脸数据集部分图像

#### 4.3.2 模型设计

本系统选用 Facenet 网络模型用于人脸识别，模型整体结构如图4-10所示，其中最左边表示一个批量的图像输入到深度卷积神经网络中，对卷积神经网络的输出进行  $L_2$  normalization 得到人脸图像的 embedding 编码，之后使用 TripletLoss 为损失函数进行训练。



图 4-10 模型整体架构图

TripletLoss 可以直接反映出人脸验证、识别和聚类所能达到的效果。换句话说，对于一张输入人脸图像  $x$  映射到特征空间  $\mathbb{R}^d$  中，通过  $f(x)$  得到 embedding，来使得相同人物的人脸图像之间的平方距离更小（与成像条件相独立），而对于来自于不同人物的图像对之间的平方距离尽可能地大。



#### 4.3.3 TripletLoss 损失函数

人脸图像的 embedding 编码由  $f(x) \in \mathbb{R}^d$  表示，它将一张图像  $x$  嵌入到一个  $d$  维的欧式空间。并且将 embedding 编码限制到一个  $d$  维的超球面，也就是使得  $\|f(x)\|_2 = 1$ 。在本模型中，要确保一个特定人物的图像  $x_i^a$  (anchor) 尽可能接近与其相同人物的其它图像  $x_i^p$  (positive)，而与剩余任何一个其他人物的图像  $x_i^n$  (negative) 尽可能远，以上过程如图4-11所示。



图 4-11 TripletLoss 训练图

以上过程可以由公式4-1表示，其中， $\alpha$  表示正类图像对与负类图像对之间的边距， $\mathcal{T}$  表示在训练集中所有可能的三元组集合。

$$\|x_i^a - x_i^p\|_2^2 + \alpha < \|x_i^a - x_i^n\|_2^2, \forall (x_i^a, x_i^p, x_i^n) \in \mathcal{T} \quad (4-1)$$

由此便可得到要被最小化的损失函数  $L$ ，由公式4-2表示。

$$L = \sum_i^N \left[ \|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha \right]_+ \quad (4-2)$$

#### 4.3.4 模型评估性能指标

所有的相同人物的人脸图像对  $(i, j)$  被表示为  $\mathcal{P}_{\text{same}}$ ，不同人物的人脸图像对表示为  $\mathcal{P}_{\text{diff}}$ 。之后，定义所有 true accept 的集合，由公式4-3表示有人脸对  $(i, j)$  被正确地分为同一位人物，其中  $D(x_i, x_j)$  表示为一对图像之间的距离， $d$  表示为距离阈值。

$$\text{TA}(d) = \{(i, j) \in \mathcal{P}_{\text{same}}, \text{ with } D(x_i, x_j) \leq d\} \quad (4-3)$$

相似地，可以定义 false accept，公式为4-4。

$$\text{FA}(d) = \{(i, j) \in \mathcal{P}_{\text{diff}}, \text{ with } D(x_i, x_j) \leq d\} \quad (4-4)$$

对于一个给定的人脸距离阈值  $d$ ，验证率 (validation rate)  $\text{VAL}(d)$  和错误接受率 (false accept rate)  $\text{FAR}(d)$  被定义为公式4-5所示。

$$\text{VAL}(d) = \frac{|\text{TA}(d)|}{|\mathcal{P}_{\text{same}}|}, \quad \text{FAR}(d) = \frac{|\text{FA}(d)|}{|\mathcal{P}_{\text{diff}}|} \quad (4-5)$$

## 参考文献

- [1] MASI I, WU Y, HASSNER T, et al. Deep face recognition: A survey[C]//2018 31st SIBGRAPI conference on graphics, patterns and images (SIBGRAPI). 2018: 471-478.
- [2] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks[J]. Advances in neural information processing systems, 2012, 25.
- [3] TAIGMAN Y, YANG M, RANZATO M, et al. Deepface: Closing the gap to human-level performance in face verification[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 1701-1708.
- [4] SZEGEDY C, LIU W, JIA Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.
- [5] SCHROFF F, KALENICHENKO D, PHILBIN J. Facenet: A unified embedding for face recognition and clustering[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 815-823.
- [6] DENG J, GUO J, XUE N, et al. Arcface: Additive angular margin loss for deep face recognition[C] //Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 4690-4699.
- [7] YID, LEI Z, LIAO S, et al. Learning face representation from scratch[J]. ArXiv preprint arXiv:1411.7923, 2014.
- [8] CAO Q, SHEN L, XIE W, et al. Vggface2: A dataset for recognising faces across pose and age[C] //2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018). 2018: 67-74.
- [9] HUANG G B, MATTAR M, BERG T, et al. Labeled faces in the wild: A database for studying face recognition in unconstrained environments[C]//Workshop on faces in 'Real-Life' Images: detection, alignment, and recognition. 2008.
- [10] WANG Z, WANG G, HUANG B, et al. Masked face recognition dataset and application[J]. ArXiv preprint arXiv:2003.09093, 2020.
- [11] ANWAR A, RAYCHOWDHURY A. Masked face recognition for secure authentication[J]. ArXiv preprint arXiv:2008.11104, 2020.
- [12] ZHANG K, ZHANG Z, LI Z, et al. Joint face detection and alignment using multitask cascaded convolutional networks[J]. IEEE signal processing letters, 2016, 23(10): 1499-1503.