

# 智能快递机器人——人脸识别算法设计

## 摘 要

智能快递机器人是一种以自动导航轮式小车为载体,搭载人脸识别检测模块以及储存箱体的智能运输机器人。其可以实现 GPS 自动导航,箱门自动开合,识别取件人面部等功能,主要运用于写字楼,小区,高校宿舍楼等物流运输的终端场所,研究目的在于为传统快递业注入一个新的生命力,以及作为智能生活应用方面的一个代表。

而人脸识别技术基于生物特征安全识别技术,作为机器视觉方面的应用,是智能识别的一个重要体现,已经广泛应用于安防,门禁,交通图像处理等领域,具有很高的安全性和可靠性,将其运用在智能快递机器人上无疑能提高产品的智能化程度。而两者结合在一起的智能快递机器人不仅能具备快捷的配送功能,也具备了安全防盗的功能。

本次研究中,人脸识别模块核心采用 k210 芯片,编程语言使用 python,基于人脸识别特征库来实现。主要使用到 python 中的 opencv, numpy, pillow 等 pip。在经过人脸检测,人脸特征采集,模型训练,人脸识别四个步骤后,达到所需的检测效果。主要以实体模块来展示人脸识别功能,并且详细对关键环节:人脸检测,人脸特征,人脸识别这三大环节来进行讲解。这款快递机器人结合先进的人脸识别技术,以一种智能生活的理念给消费者带来全新的体验,也大大减轻了快递员的负担,使派件变得更加便捷。

智能快递机器人的研究想法来源于生活,对于我们平常的取快递时种种的不便产生的疑惑和烦恼,这款机器人都能很大程度上解决这些问题,智能机器人最重要的一点就是能够为人类提供最大的帮助,这也是我们想法的初衷。

关键词: 智能快递机器人, 人脸识别, 生物特征识别技术, python

# **Intelligent delivery robot - face recognition algorithm design**

## **Abstract**

Intelligent express robot is a kind of intelligent transport robot with automatic navigation wheeled vehicle as the carrier, equipped with face recognition detection module and storage box. It can realize the functions of GPS automatic navigation, automatic opening and closing of the box door, recognizing the face of the person taking the package, etc. It is mainly used in the terminal places of logistics transportation such as office buildings, residential areas, university dormitory buildings, etc. The purpose of the research is to inject a new vitality into the traditional express industry, and as a representative of intelligent life application.

And facial recognition technology based on biometric security identification technology, as a machine vision applications, is an important embodiment of intelligent recognition, has been widely used in security, access control, traffic areas such as image processing, has the very high safety and reliability, and applied it in the intelligent express robot can improve the intelligent degree of the product. The combination of the two intelligent delivery robot can not only have a quick delivery function, but also has the function of security.

In this study, k210 chip is used as the core of the face recognition module, and python is used as the programming language, which is realized based on the face recognition feature library. Mainly used to python in opencv,numpy,pillow and other PIP. After four steps of face detection, face feature acquisition, model training and face recognition, the desired detection effect is achieved. The function of face recognition is mainly demonstrated by entity module, and the key links, such as face detection, face features and face recognition, are explained in detail. The delivery robot combines with advanced face recognition technology to bring a new experience to consumers with the concept of intelligent life, which also greatly reduces the burden of couriers and makes delivery more convenient.

The research idea of intelligent express robot comes from life. As for the doubts and troubles caused by various inconveniences when we pick up express, this robot can solve these problems to a great extent. The most important point of intelligent robot is that it can provide the greatest help to human beings, which is also the original intention of our idea.

Key words: intelligent express robot, Face recognition,Biometrics, python

# 目 录

<a href="#">1 前言</a> .....	1
<a href="#">1.1 智能快递机器人及人脸识别技术</a> 的目的、意义.....	1
<a href="#">1.2 智能快递机器人</a> 应达到的技术要求.....	4
<a href="#">1.3 智能快递机器人</a> 在国内外的的发展概况及存在的问题.....	5
<a href="#">1.4 智能快递机器人</a> 应解决的主要问题.....	7
<a href="#">2 人脸识别算法设计</a> .....	9
<a href="#">2.1 设计原理</a> .....	10
<a href="#">2.1.1 各部分技术原理</a> .....	10
<a href="#">2.1.2 人脸识别模块参数</a> .....	12
<a href="#">2.2 方案选择</a> .....	13
<a href="#">2.2.1 存在的影响因素</a> .....	15
<a href="#">2.2 设计过程</a> .....	15
<a href="#">2.2.1 人脸检测</a> .....	16
<a href="#">2.2.1 人脸特征数据采集</a> .....	17
<a href="#">2.2.1 人脸训练</a> .....	19
<a href="#">2.2.1 人脸识别</a> .....	20
<a href="#">2.2.1 模块的准备工作及调试</a> .....	22
<a href="#">3 结论</a> .....	23
<a href="#">参考文献</a> .....	25
<a href="#">谢辞</a> .....	26
<a href="#">附录</a> .....	27

# 1 前言

众所周知，现在中国的快递业正在面临巨大的挑战，尤其是在最后的环节——终端配送的难题逐渐成为各快递公司的关注点和急需解决的地方。而且这样的难题不仅仅存在于快递行业，也紧紧关系着城市和农村的信息化建设和网络化服务的铺设。而在分析国内外的智能机器人的发展情况以及物流行业的运作方式之间的差异之后，我们自主进行了智能快递机器人课题的调查研究以及机器人实物的研发工作，对于传统快递业提出了几点新的理念和建议。

首先智能快递机器人是一种基本可以完全实现无人控制的一种物流配送终端的智能型机器人。它以强大的智能芯片为控制中枢，搭载了四个直流驱动电机为动力部件，采用先进的人脸识别模块，人脸识别速率可达到一秒以内，然后在人脸识别成功后给箱门继电器输出一个高电平信号，使连接箱门的机械手臂电机运作，完成打开箱门动作，在收件人取出快递后可自动闭合箱门。智能快递机器人的主要特点在于轻巧的身躯，强大的 GPS 自动导航技术，以及安全可靠的人脸识别技术。

而人脸识别技术是基于计算机视觉应用的一个重要研究领域，在图像检索，身份识别等都有广泛应用。而光线补偿技术，高斯平滑技术以及二值化技术均是提高人脸识别准确率的重要技术之一。在许多国家，人脸识别在军事领域，智能家居，移动式电子产品上都已经相当普及。

这两者的结合产物——智能快递机器人的主要作用是能解决物流的终端配送问题，可以在任意指定的时间内前往收件人指定的相对固定场所，由 GPS 自动导航到达目的地后，经过人脸识别读取到收件人的信息，确认无误后打开箱门送出快递，完成配送过程。这是在物流领域中一次创新性的尝试，突破了传统人力配送的固定方式，可给传统快递业注入新的生命力。

## 1.1 智能快递机器人及人脸识别技术的目的、意义

智能快递机器人的设计主要有三个目的：

### 1. 结合人工智能，加速快递行业的发展，提高中国智能化生活程度

快递行业从古至今就是一种国民性的服务型行业，从中国古代的镖局，到如今遍布世界各地的快递运输公司，都表明人们离不开快递这一服务行业。在欧美等发达国家更是如此，快递公司在全球的运输行业公司排名中都占据着前列。

而在中国，随着电子商务的兴起，快递行业也在不断发展当中，由于中国人口基数庞大，国人对于电子消费的热度自 2008 年以来持续上升，快递的数量更是与日俱增。据国家邮政局局长在 2020 年的工作会议指出，2019 年全国快递总量达到 630 亿件，是

欧美国家的数倍之多，而且中国快递员的数量高达 320 万，但仍然是处于紧缺的状态。随着近年来劳动力成本的上涨，如此庞大的产业如果还继续依靠这种普通的人力运输来支撑，只会使其发展变缓，甚至会使用户的体验越来越差。而著名的国营企业邮政快递正是因为服务质量参差不齐，服务效率逐渐下降，导致市场份额连年下跌，大部分民营快递公司趁势崛起，以信息化和智能化的服务为方针赢得了大批的用户。由此看出，传统的人力快递运输行业急需改革。

智能化生活通常是指在数据分析，人工智能，物联网+等技术支持下，给使用者在日常生活中提供多种自动化帮助。例如现在已经推出市场的声控机器人，智能家具，无人驾驶汽车等，能极大地解放人们的双手。而我国近年来不断推动城市智能化发展和普及智慧城市的观念，但智能化生活似乎离普通百姓的生活很远，一方面是智能机器人的技术还不够成熟，另一方面是这方面的产品较少，没有涵盖重要的领域。这就证明了智能快递机器人既不仅仅是一种新型理念，也是为了填补快递行业智能化产品的空白，符合我国推行智能化生活前进的步伐。

在国家政策方面，随着第四次工业革命的呼声越来越高，而且在“中国制造 2025”政策也指出：要加快和推动智能服务型机器人的研发以及产业化，重点鼓励和培养服务型机器人对于智能型机器人的研究。所以可以体现出中国服务业数字化，网络化和智能化的改革势在必行，而传统快递业作为服务业的标杆，更应该率先打响这改革的第一枪，智能快递机器人就是首先要注重的一个研究方向。只有把快递运输与数据分析以及人工智能结合起来，争取实现互联网+机器人物流，才能达到高度去人力化，降低消费成本中的劳动力占比，加速快递业的发展。

## 2. 减轻快递员负担，提高配送效率

或许一提到快递员这个行业，不少人的第一印象可能都是感觉又脏又累，工资水平相对来说一般，工作时间比较长，有可能遭受严寒和酷暑的环境。的确，快递员这个职业从来都不和轻松挂钩，甚至有时候会因为配送时间以及地点问题与顾客产生不必要的矛盾。这时候，智能快递机器人目的就在于克服这一系列的困难，既不会受制于一般的外界环境，也不会产生任何的劳累感，可一直工作到能源耗尽，并且还能以良好的服务质量取得顾客的认可。

另一方面，传统的快递员的工作通常都是分成两大类型，一是在仓库中进行快递的挑拣，分类和储存。二是把快递从发货地运到指定地点仓库后，再分区域配送到不同的小区住宅，办公场所。由于仓库通常货物数量庞大，如果仅靠人工分拣，不仅有可能出错，对于快递员来说也是体力和精神上的挑战。而小区住宅以及各种公司的办公地点虽然是相对固定的，但存在楼层众多，取件人位置分散，收货人取件的时间无法集中在某一个时间段等等问题。这一系列问题会导致快递员工作时间变长，工作精力会被大量消耗在路途中，而且效率也不能得到稳定保证。而智能快递机器人能全程无人参与，有极低的容错率，能定时定点准确地把快递送到收货地点，依靠云数据存储用户信息，可

即时读取，节省了快递员的配送路程遥远的问题和解决了用电话通知取件人时产生的费用问题，极大地提高了配送效率。

### 3. 优化用户体验，提升快递安全

任何智能产品的设计初衷都是给用户最好的体验。相信不少人在取件时都会有过这样一些经历：取件时有些快递员会因为小区楼房没有电梯而选择在楼下派件；还有就是配送时间的不固定可能会造成自己不在家时快递刚好到了，导致无法取件。而智能快递机器人能够完美解决这些问题，一方面是能自动按电梯上下楼，遇到无电梯小区也能开启爬楼梯模式，轻松送货到家门；二是能根据收货地点的距离自动预计到达的时间，提前给收件人发送信息，决定是否派件。这样既使派件变得便捷，也可以尽可能地利用好时间。

近年来也有发生不少快递在派送过程中丢失，被盗的情况，而快递公司对于没有保价的货物一般是采取低价赔偿政策的，所以快递安全问题已经成为用户首先要考虑的问题之一。而智能快递机器人正是为解决这一问题而生。它首先采用坚硬的钢铁车架，全密闭结构的外壳，给快递提供一个安全的运输环境。其次内置安全报警装置，一旦有人尝试强制打开箱门抢走快递，它就会自动锁住箱门并把信息反馈给警察部门以及公司相关的负责机构，确保快递不会丢失。在用户取件方面，也内置先进的人脸识别模块，有快速响应的夜间识别摄像头，再结合服务器上储存的用户信息，确保把快递交到正确的用户手上。

智能快递机器人结合人脸识别技术的意义：

目前，我国的快递收件人群主要集中在社区，办公大楼等，配送方式还是以自主取件和人工配送为主。但这两种配送方式都存在一定的缺陷。

第一种自主取件也是近几年才流行的一种新取件模式，通常是指把快递放到与快递公司合作的服务点，或者是放到快递公司专门研制的无人取货柜里面，然后用户再根据自己的选择在规定时间内自行到达取件点取走快递。这从人力资源层面上分析，的确省去了快递员大部分的劳力和时间成本，也达到了集中管理模式的目的。但这种模式牺牲了用户的体验，在一定程度上给用户造成很大的不便。因为用户需要自己亲自到达取件点，一些距离取件点较远的用户会感到非常困扰，而且如果是人工服务点的话，也无法做到 24 小时取件服务，一些用户会觉得这样的服务较差，对快递公司产生排斥感。

第二种人工配送是最直接也是最普遍的快递配送方式。这种配送方式一般采用人+车，在不指定的时间内随机进行配送。这种模式同样存在以下几点缺陷：

- a) 人力成本，运输车辆成本，时间成本较高；
- b) 快递的安全保护性不高；
- c) 时间上没有系统性的规划，配送效率较低，有时候无法与用户达成统一的时间，导致产生时间上的冲突；

d) 无法达到无接触配送要求，在一些特殊时期以及场合无法进行配送；

综上所述，智能快递机器人的意义不言而喻。近看，结合当前传统快递配送方式以及用户的需求，其能解决“最后一公里”的配送问题，优化配送时间，极大地提高了取快递时的安全性和便捷性。也能保障用户的隐私，引导出一种以人为本，客户至上的服务理念，在市场价值和社会意义上有无可比拟的优点。同时，它与传统快递配送方式产生巨大的差异感，容易被年轻人带来丰富的新鲜感，而电子商务的主要消费人群也主要是年轻人，从侧面看这也是提高业务的一种方式，也是一次快递配送领域的创新尝试。

远看，随着我国现在生育率越来越低，人口老龄化越来越严重，一直以来依靠的人口红利正在逐渐消退，快递行业的劳动力缺口不断扩大，智能快递机器人不仅是给传统快递业带来一次全新的技术革命，也是一股智能化时代中的新兴力量。国家也因意识到这一点早在 2017 年提出了《新一代人工智能发展规划》，报告指出要鼓励全国企业在 2030 年抢占人工智能的领先地位，大部分国内科技企业也开始大力发展 AI 技术，对智能快递机器人的研究必然是重中之重，是顺应时代潮流，也是响应国家的号召。

其次，人脸识别技术也具有很高的应用意义。长期以来，锁的发展由传统的物理锁结构发展到以指纹识别为代表的电子信息锁结构，无数领域的学者都在研究如何更快、更准确地实现电子身份识别功能。指纹识别虽然发展较早，技术较为简单，但手指作为与外界经常接触的地方，很容易发生物理性质的改变，从而影响识别速率。而人脸识别技术是社会发展的产物，以便捷可靠的优点迅速占领了人们的日常生活，例如现在人人都在用的刷脸支付，手机的 Face ID 等。人脸识别技术具有广阔的市场前景和巨大的社会需求。

## 1.2 智能快递机器人应达到的技术要求

本课题所讨论的智能快递机器人概念设计应满足以下技术要求：

A. 搭载 GPS 自动导航和陀螺仪，能实现无人自动行驶，能实时精确地监控到快递机器人位置，并且能根据系统输入的位置自动规划出最优路线自动行驶。

B. 能根据内置的多个超声波传感器实时感应当地天气情况，侦察路径上路况信息，感应前方障碍物并进行躲避。四个车轮均配备减震系统，减少细小障碍物带来的影响。

C. 与快递公司专用 APP 平台连线，在配送快递前机器人可自动提前给用户发送信息沟通，以及可以让用户在软件上实时查看机器人的位置，还可以享受自主预约服务。

D. 搭载先进的人脸识别模块，采用红外夜视双摄像头，可在昏暗的环境中工作。可存储大量用户照片信息。基于三维特征脸方法来进行人脸识别，对比过程，然后识别成功后自动打开箱门递出快递。

E. 配置行车摄像头，对外界环境实时记录。配置自动报警装置，遇到紧急情况可

自动报警。

自主设计能达到的具体技术要求：

- A. 通过手机遥控器控制快递机器人的移动，配置超声波避障功能
- B. 采用 51 单片机为控制核心
- C. 储存箱体和机器人主体之间为可分离机构
- D. 配置人脸识别模块，可储存大量人脸信息，可实现人脸检测，人脸追踪，人脸对比，响应速率在一秒内，其中人脸识别技术采用几何特征识别方法。
- E. 车架内部为储存空间，由机械臂控制箱门的开闭；四个轮子由 tt 减速马达驱动。

### 1.3 智能快递机器人在国内外的的发展概况及存在的问题

其实智能快递机器人的本质也是属于智能机器人的一类。而科学界广泛意义的对于智能机器人的理解是：拥有一个发达的中枢大脑（中央处理器），能像人一样有独立思考的行动能力。当然，这是理论上的完美的智能机器人，而我们现在普遍的技术使得它们或多或少都需要人类的控制。智能快递机器人作为新时代的代表性产物，既符合国内外关于智能制造持续发展的理念，也符合机器为人服务的最终目的。

在国外，智能机器人的研究早已非常普遍，几乎每个国家都把智能机器人的研发当作是国家的软实力，而智能快递机器人作为智能机器人的代表性产品之一，自然也是受到广泛关注。国外的公司不仅研制出仅鞋盒般大小的仓库物流机器人，也有像自行车般大小的快递运输机器人。例如以色列有一名设计师曾设计出一款名为 Transwheel 的快递机器人。它是采用多个小机器人可组合成大机器人的模式来运载货物。它具有出色的自平衡系统，在运输小件货物时由单个机器人运输，运输大件货物时则组合成一个大机器人来共同运输。此外该机器人具备 GPS 自动导航系统，对于取件的安全性方面也采取了人脸识别系统来保证安全。此外在美国还有一款名为 Robby 的智能快递机器人，其功能与 Transwheel 相似，但它具备多个先进的传感器和激光雷达技术，可以精确地实时检测自己在地图上的位置，据说精度达到厘米之差。而在电商物流机器人配送领域较为有名气的应该就是亚马逊了，它最早采用了无人机进行物流终端配送，在当时受到不少消费者的好评，可以说是在物流的管理模式和服务模式上提出了改革性的方针。

虽然我国的智能机器人的研究时间相对欧美等发达国家来说较晚，但由于近几年国内人工智能技术的火热以及机器人技术的发展，再加上国内快递行业规模不断扩大带来的人力资源不足等问题，各快递公司不断给智能快递机器人的研究注入新的活力，在快递机器人自主专利产权方面取得丰硕的成果。比较有代表性的就是电商巨头之一——京东研发的京东配送机器人。它具备主动避障，人脸识别，GPS 定位，全景视觉系统等高科技功能，自 2017 年诞生至今，已经陆续进入到各所大学校园里进行配送服务。而更



由于今年的新型冠状病毒在我国疯狂传播，大部分居民都因担心交叉感染而不敢与快递员接触，使物流配送变得十分困难。这时京东最新研制的“铁憨憨”智能配送机器人便派上了用场，投放到武汉多个小区进行快递无接触配送服务，让附近居民能在特殊时期感受到人工智能以及互联网技术在生活上带来的便捷，可充分看出智能化的生活在中国的前景会越来越广阔，而智能快递机器人就是这智能化生活不可缺少的一环。

虽然智能快递机器人拥有许多的优点，近年来发展速度也比较快，但至今仍无法普及到大部分人的生活当中，在国内只是局限地在部分地区投入试用，经过我们的调查研究得知，其仍然存在着以下几点问题：1. 技术不够成熟，未能大面积投放到市场，在一些重要的知识产权领域仍然受制于国外科技公司，例如一些传感器技术以及自动导航控制技术。现在我国仍有大量的国外机器人科技公司占据着 80% 以上的智能机器人市场。2. 人们对于新产物的评价褒贬不一，尤其是一些快递员对其产生不满，认为机器人会剥夺他们工作机会，开始抵制智能快递机器人。而国人自古以来就不善于快速接受新的事物，对于智能机器人的了解程度不高，有相当一部分人可能还是倾向于传统的人工配送模式。这两点因素在一定程度上制约了智能快递机器人的发展，但我相信未来它会突破这些困难，最终取得成功。



图 1.1 京东物流机器人

#### 1.4 智能快递机器人应解决的主要问题

目前，智能快递机器人应解决主要问题有以下四点：

##### 1. 车辆行驶的稳定性较差，速度较慢

由于大部分智能快递机器人都是采用轮式小车为主体，而且结合体积和能耗的综合情况，一般主体尺寸较小，与一般婴儿车大小类似。这样就会导致小车在行使过程中，一旦遇到颠簸，或者有小石头等障碍物的地方，就很容易重心不平衡，发生侧翻的情况。

在速度方面，由于智能快递机器人需要携带大量运输物品，而自身由于体积的限制以及需要配置许多电子元器件而无法携带大容量的电源，所以自身的能源供给无法维持

高速的行驶速度。一般快递机器人的速度在每小时 5 公里左右，这种低速行驶虽在一定程度上保证了周围行人的安全，但带来的缺点就是速度相对于普通快递员来说肯定是比较慢的，在直线速度上比不上传统人脸运输。

## 2. GPS 信号不良，无法按规定路线行驶

智能快递机器人最重要的技术之一就是 GPS 自动导航功能。GPS 又称全球卫星定位系统，是利用远在太空的卫星发射的微波，可给使用者提供精确的三维位置信息。由于采用远距离微波传输信号，而 GPS 导航自动技术又对于信号强弱非常敏感，所以在一些密闭的场合，高楼大厦密集的场所或者离基站过远的地方，都会造成 GPS 信号弱甚至无信号。这就给 GPS 自动导航带来一定的难度。因为智能快递机器人的工作地点往往就在于居民楼内，大楼内部等。

由于自动行驶是按照地图路线规划进行的，所以在面对一些复杂的道路，室内等地图难以仔细划分的情况时，就有可能出现无法到达指定地点的情况。

## 3. 机器存在被人为损坏的风险

智能快递机器人集多种高科技技术于一身，其中不缺乏昂贵的零件。由于机器人工作的场合有可能经过人少的区域又或者是黑夜之中，即使拥有紧急报警系统和自动锁定系统，但在遭到人为破坏时是无法第一时间自行阻止的。就好像共享经济时，每个共享单车企业都必须要花费巨额的维护费用一样。这就造成了企业必须承担修理成本甚至是更换成本的风险，这也会成为快递机器人无法在企业得到重视的原因之一。

## 4. 人脸识别技术仍存在不足之处

虽然人脸识别技术在大部分理想的环境下是能取得符合人们要求的结果，但在一些采集信息环境不理想，用户自身的生物特征不明确的情况下，系统的识别成功率将大幅度下降。例如一些用户在储存人脸信息之后，脸部特征发生了细微或大幅变化，如脸上青春痘的增长，胡子长度与密度的变化，以及当时表情的不同，都会有可能使系统检测失败。如今即使是最先进的人脸识别系统准确率也只有 99%，无法做到百分百准确率。

据网络安全专家张百川介绍：人脸识别数据在经过大规模应用时，并非是万无一失的，肯定会存在一定的风险，就好像指纹识别一样，也曾有过数据泄露的安全隐患。在 19 年末，美国一家名为 Kneron 的人工智能企业宣称用自制的 3D 打印面具破解了中国的人脸识别系统。虽然种种事件的真实性还未得到官方确认，但无疑是给正在研发人脸识别技术的企业敲响了警钟，人脸识别的可靠性和安全性还需进一步得到完善。

## 2 人脸识别算法设计

在智能快递机器人课题研究中，本人主要负责人脸识别模块的算法设计及调试，辅助队友完成机械结构的设计和控制系统开发。

在充分了解其控制系统原理的前提下，采用了 SPIEED MF1 人脸识别模块为主控模块，使用 Python 语言在 OpenCV 平台的基础上对安全识别取件功能进行程序设计及优化，并且尝试与快递车的箱门机械手臂进行联动，使得人脸识别成功后可自动打开箱门，取出快递。

1) 该人脸识别模块的特性如下：

- ❖ 通过双光谱摄像头判断识别活体人脸
- ❖ 适用于无光线直射的室内和室外环境，遇到黑暗环境可自动切换至红外线识别
- ❖ 默认的有效识别距离为 40~80cm
- ❖ 开机启动时间大约为 0.3 秒，识别速度最快可在 0.5 秒左右，识别准确率为 98%@0.001FA
- ❖ 内置 16M 储存数据空间，可存储人脸特征值数量大约为 2200 张



图 2.1 MF1 模块

2) 采用 OpenCV 来进行人脸识别程序的开发及调试。OpenCV 是一个开源的计算视觉库，可以运行在 Linux, Window, Mac OS 等计算机系统上，可支持提供 Python, Ruby, MATLAB 等语言接口，实现机器视觉方面的多元化算法。其应用非常广泛，且可靠程度高，检测成功率高，是实现人脸识别功能的优秀平台。

3) Python 于 20 世纪 90 年代初诞生，到现在经过不断的发展，已经成为一门最受欢迎的编程语言之一。Python 与 C 语言, Java 等属于编程语言的一种，偏向于解释

型脚本语言。主要用于计算机程序设计，具有强大的功能和广阔的拓展性，并且优势在于简单易学，上手速度快。Python 在开始运行的时候，首先会将 py 文件中的代码翻译成字节码，然后再由 Virtual Machine 将其执行。此外，Python 还有良好的交互模式，在开源系统 Linux，Mac 系统上都可以直接运行。

此外 Python 的具体应用还有：图形处理，文本处理，数字处理以及简单的网络爬虫。Python 还具有强大的标准库和大量的第三方模块，涵盖了系统开发，数据库连接，图形系统等多个领域，是实现人脸识别功能的最优选择。

## 2.1 设计原理

首先，我们来初步了解一下人脸识别技术的概念：人脸识别技术是近年来非常火热的一项计算机与机器视觉相结合的研究领域。它与眼球识别技术，静脉识别技术等都属于生物特征识别技术，是以人本身的特有的生物特征来达到区分个体的目的。

人脸识别技术基于人的面部特征，能对录入的人脸图像或者视频进行分析处理。首先是判断是否存在空白信息，也就是是否存在人脸。然后如果存在人脸，再进一步根据每个脸部器官的位置信息，大小差异来提取每个人脸部代表的个人身份信息。最后与云端已存储的人脸信息进行对比，达到识别目的。

### 2.1.1 各部分技术原理

- 人脸检测：

参考模板法：这是最基本的人脸检测方法，首先准备好一个人脸的标准模板，然后计算测试中采样的信息与模板之间的匹配程度，并通过阈值计算来判断人脸是否存在。

肤色模型法：不同的人肤色会有所差异，这种方法主要是根据肤色在色彩范围内分布的值的不同来进行检测。

可视特征法：这种方法是根据特征之间反射的信息之间的差异来进行。将所有的面集成为一个整体，基于检测物在投影上的距离来判断是否存在

另外一个检测步骤是人脸跟踪：

人脸跟踪是指在检测到目标为有效面部后，对目标进行动态追踪。具体采用三维模型与运动仿真相结合的方法，此外也有利用肤色模型的方法来进行跟踪。

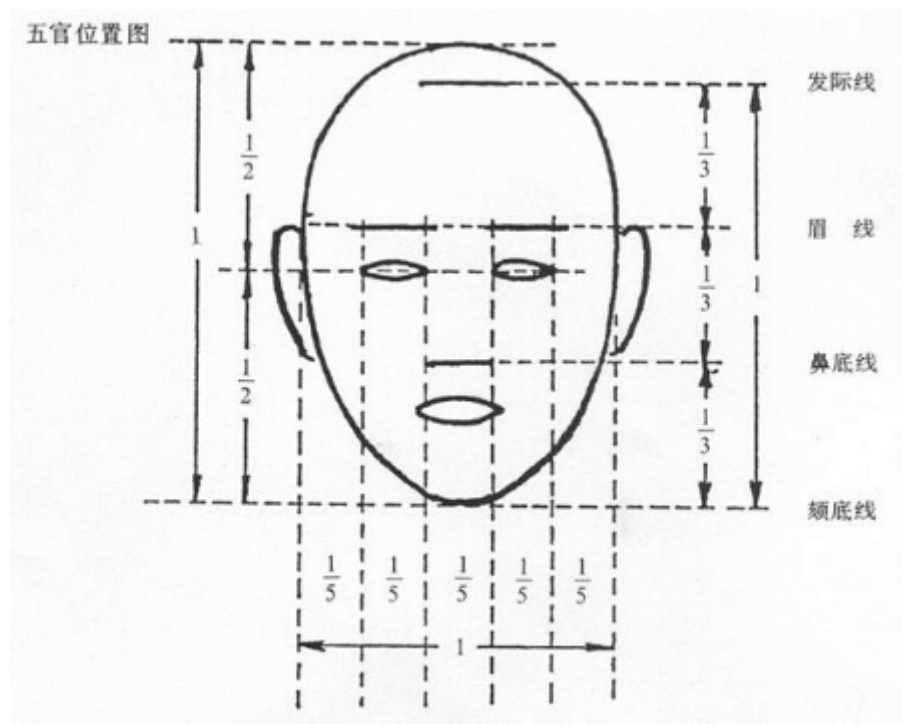


图 2.2 人脸特征分布关系图

### ● 人脸特征采集

人脸特征采集步骤就是提取人脸上的细微特征。主要流程在于输入图像后，与识别库中的一张已存人脸图片进行联系，形成一个识别系统。具体方法如下：

**几何特征法：**总的来说就是把人脸看似成一个几何特征总量，并根据这些特征去设计分类器。在选取几何特征矢量是要尽可能采用差异较大的部分，同时要考虑光线变化产生的问题。

**代数特征法：**主要是把图像上的一个个像素点替换成空间上的投影点，从而对各个像素用基本图像来编码。常采用 PCA 采集法，仅采用一种少量特征描述方法来达到降维特征空间的目的。其基本公式是采用 K-L 展开公式，可以使所有图像都被投影到同一空间上，能很好地达到降维目的。

**基于人工神经网络：**机器视觉近几年发展迅速，在人脸识别上的应用也开始研究。神经网络是一个庞大的可自我计算的一个系统，能处理复杂的算法。主要特点在于具备学习和分类的能力，也可以通过大量训练来达到自我适应的能力。

### ● 人脸识别对比

人脸识别对比就是对检测成功的完整的人脸进行识别，也就是最后一步。而人脸识别又主要分为以下两种算法原理：1. 基于几何特征，2. 以特征脸为基础。

首先基于几何特征方法的原理在于：将重要特征的位置信息及形状参数作为一个分类，因为人五官的轮廓，角度，之间的距离是有很大差异的。通过设计一个参数可自

行调整的五官函数模型，就可以作为这个人脸的几何特征的对比向量。

特征脸方法也是基于PCA算法的一种方法，只不过这种算法相对简单且速度比较快。主要就是利用一个二维矩阵，将检测好的人脸图像投影到空间当中，通过对特征的加权运算和描述，将其与已存的图像权值进行对比即可。



图 2.3 人脸识别原理流程图

### 2.1.2 人脸识别模块参数

SIPEED MF1 人脸识别模块是基于 M1W 模块，以 K210 为核心的 AI 模块。它搭载了一对在黑暗环境中也能清楚识别人脸的双红外摄像头，一块用于显示人脸画面的 1.3 寸 TFT 屏，一个 RGB LED 指示灯。这个模块主要结合 Opencv 平台来实现人脸识别功能，用途相当广泛。其具体参数如下表：

表 2.1 MF1 具体参数表

K210 芯片参数	1.内核：RISC-V Dual Core 64bit, with FPU
	2.主频率：400MHz
	3.SRAM：内置 8M Byte
	4.语音识别：麦克风阵列（8mics）
	5.图像识别：QVGA@60fps/VGA@30fps
	6.深度学习框架：支持 TensorFlow/kears/Darknet/Caffe 等框架
	7.网络模型：支持 YOLOv3/TinyYOLOv3/Mobilenetv2/人脸识别等
	8.外设：FPIOA. UART. GPIO. SPI. TIMER

	<p>9.视频处理:</p> <ul style="list-style-type: none"> <li>➤ 神经网络处理器 (kpu)</li> <li>➤ FPU 满足 IEEE754-2008 标准</li> <li>➤ 音频处理器 (APU)</li> <li>➤ 快速傅里叶变换加速器 (FFT)</li> </ul>
模块参数	<p>1. 板载配件: 按键*1</p> <p>双摄像头*1</p> <p>RGB LED*1</p> <p>1.3 寸 TFT 屏幕*1</p> <p>128mbit Flash*1</p>
	<p>2. 接口类型: USB Type-c 接口</p> <p>音频接口 (可外接扬声器)</p> <p>有 IO 排针引脚引出</p>
	<p>3. 尺寸大小: 46.3*37.1mm</p>
	<p>4. 工作电压: <a href="#">4.0-5.2V @ 300mA</a> (电流大于 800mA)</p>
	<p>5. 工作温度: -30° C-85° C</p>
开发环境	<p>芯片操作系统: FreeRTOS. RT-Thread 等</p>
	<p>开发环境: MaixPy IDE. Python IDE. opencv 等</p> <p>编程语言: C. C++. Python</p>

## 2.2 方案选择

经调查发现, 现在主流的人脸识别算法方案主要有以下几个选择:

### 1. MTCNN 算法

MTCNN 主要由 PNet, RNet, ONet 组成, 其中 PNet 是一种可获得人脸区域大小与边界的回归向量, 并且能通过非极大值抑制来合并多个重复的窗口。PNet 可输入任意大小的图片, 并且对于图片会有一个循环缩放的过程。一张  $6*6*1.5$  的图片最终经过循环可输出为  $0.5*0.5*15$  的特征图, 再分成三个方向用于人脸分类, 特征定位。

而 RNet 的网络结构与 PNet 存在差异，多了一个全连接层，能更好地抑制 false-positive。RNet 在生成特征量过程中会进行检测，避免特征向量的坐标超出原始图片。由于采用全连接层结构，所以输入的图片统一为 24\*24 大小，最终输出结果为 3\*3\*64 特征图。

ONet 则相比 R-Net 多了一层卷基层，所以能提供更加准确的处理结果。将 48\*48\*3 的图片输入后可输出 3\*3\*128 的特征图，同样是经过三条支路用于特征定位。

## 2. 虹软人脸识别算法

虹软属于一种人脸检测技术，作用是能检测并定位照片或视频中的人脸。主要以三个库为基础，实现整个人脸识别的流程。

首先是人脸注册，主要包含 byte 数组，类 FaceDB，用于储存用户数据，其次是初始化引擎，需要定义一些有参数值的构造函数以及析构函数，以此来实现人脸信息的增加和读取功能，示例部分代码如下：

```
if (requestCode == REQUEST_CODE_IMAGE_OP && resultCode == RESULT_OK) {  
    mPath = data.getData();  
    String file = getPath(mPath);  
    //TODO: add image coversion }  
}
```

最后调用 AFD\_FSDK\_StillImageFaceDetection 来反馈检测信息。

## 3. 基于 Python 和 opencv 库的人脸识别

这是一种基于特征分析的方法，首先是进行局部人体面部特征分解，再用图形+训练数据识别算法，将其几何关系数据与数据库参数进行比较和判断。主要运用 Python 库中包含 numpy 扩展库和 opencv 自带的 EigenFaces 人脸识别器。这种方法比较依靠测试集图的训练数据成果，也对测试图像和训练集图的相似性要求较高。但这种指导思想不同之处在于不用对图片进行过于复杂的处理，也不用采取抽调模板的方法，能综合运用统计学，信息学等知识，相对于我们初学者来说较为简单。

经过小组讨论，对每个人脸识别模块功能的探索以及算法的分析，本次课题研究主要选择最后一种方案。

### 2.2.1 存在的影响因素

针对在开发人脸识别模块过程中遇到的问题，个人认为有以下几点因素，结合参考文献对其进行了个人的分析。

#### i. 光照的变化



在人脸识别过程中，由于人体表面会分泌出油脂，且复杂的 3D 结构会形成多个反光层。光线的变化导致的折射率的变化都会使识别速率受到影响。

对于光线问题进行分析可知，一方面可以通过记录可见光谱的变化来减小受到的影响；另一方面也可以使用主成像技术来在固定照明条件下采集图像。

### i. 表情变化

在识别过程中发现，如果使用与一开始录入的图像不同的表情，识别准确率就会有所浮动。我觉得是因为人脸的空间维度就像正方体一样，有六个方向。其中沿着 X, Y 方向的变化为位置变化，而沿着 Z 轴的变化为大小比例变化，绕着这三个轴旋转的变化则是脸部角度变化。

上述的变化由于会产生不同的数据，给系统传递错误的信息，所以会影响人脸识别精度。我觉得其实可以通过计算机先推算出某张人脸可能变化出的姿态范围，再把属于这个变化范围内的人脸归纳成统一的一张人脸信息，再进行识别；又或者是把多种表情记忆下来，建立一个专门的表情库，不过这种方法会产生大量信息。

### ii. 年龄变化

据我调查的资料显示，一些人的外形及皮肤纹理的变化，也会带来人脸识别精度下降。因为随着年龄增长，人的外形轮廓和皮肤纹理深沟也会带来像第一种光线变化的情况。解决方法也类似于第二种，即是在图像中识别人脸年龄，再设法推算出随着年龄增长的脸部变化，将其整理成统一数据。在特征训练中，也可以建立其人脸年龄函数，自动计算出当前人脸处于哪一个年龄段。

## 2.3 设计过程

首先明确人脸识别系统的步骤组成，第一步是先输入人脸图像，并设计程序让系统能检测到人脸的存在；第二步是检测人脸特征数据，也就是对五官等三维特征数据进行收集；第三步是对不同的素材进行模型训练，最后一步就是根据以上步骤识别出人脸身份。

### 2.3.1 人脸检测

这是比较重要且困难的一步，因为要利用 Python 中许多不同类型的库，如果缺少任何一个程序都进行不下去。以下结合几个重要步骤的程序来讲解我的思路。

1. 先调用 Python 里含有的识别分类器，主要运用一般常用的人脸的识别分类器—haarcascade\_frontalface\_default 以及识别双眼的 haarcascade\_eye 分类器。而 haar 特征也是人脸检测中常用且有效的一种方法。代码如下：

```
#人脸识别分类器
```

```
faceCascade=cv2.CascadeClassifier(r'C:\Users\leo12\AppData\Local\Programs\Python\Python38\Lib\site-packages\cv2\data\haarcascade_frontalface_default.xml')
```

```
# 识别眼睛的分类器
```

```
eyeCascade=cv2.CascadeClassifier(r'C:\Users\leo12\AppData\Local\Programs\Python\Python38\Lib\site-packages\cv2\data\haarcascade_eye.xml')
```

而我在调用分类器时也出现过一点问题，就是分类器的路径一定要是完整路径，不然它自己寻找不到，我当时只是填了 python 的安装路径，导致程序出错。

1. 接下来就是人脸框架的检测，这部分就是识别出脸和眼睛的大致方位，并用矩形框标出来，主要就是把检测到的各个坐标换成绝对位置的过程，部分代码如下：

```
# 人脸检测
```

```
faces = faceCascade.detectMultiScale(  
    gray,  
    scaleFactor=1.2,  
    minNeighbors=5,  
    minSize=(32, 32)
```

```
# 在检测人脸的基础上检测眼睛
```

```
for (x, y, w, h) in faces:
```

```
    fac_gray = gray[y: (y+h), x: (x+w)]
```

```
    result = []
```

```
    eyes = eyeCascade.detectMultiScale(fac_gray, 1.3, 2)
```

```
# 眼睛坐标的换算，将相对位置换成绝对位置
```

```
for (ex, ey, ew, eh) in eyes:
```

```
    result.append((x+ex, y+ey, ew, eh))
```

附上检测效果图：

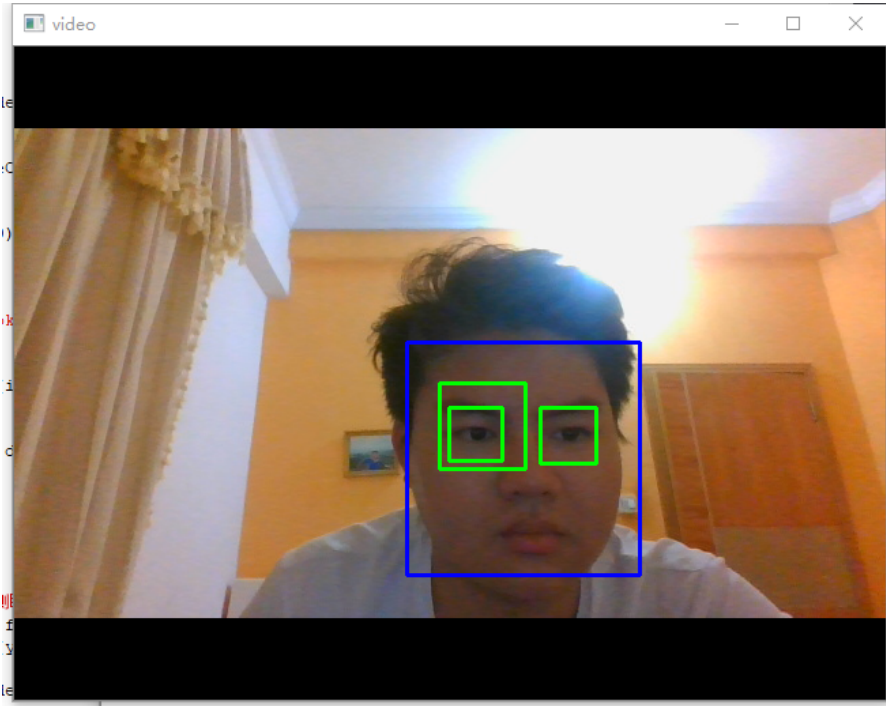


图 2.4 人脸检测图

### 2.3.2 人脸特征数据采集

这一部分其实只是为接下来的模型训练作素材准备。主要是先从摄像头中读取图片，然后经过灰度转换，然后和上面同样的检测人脸，但这个时候只需要简单的整脸采集，不必做单独的眼睛定位，在持续拍摄 500 张图片样本后就完成采集工作了。

✓ 首先新建存放样本的文件夹，我这里选择把分类器也一起存放，以免出错。

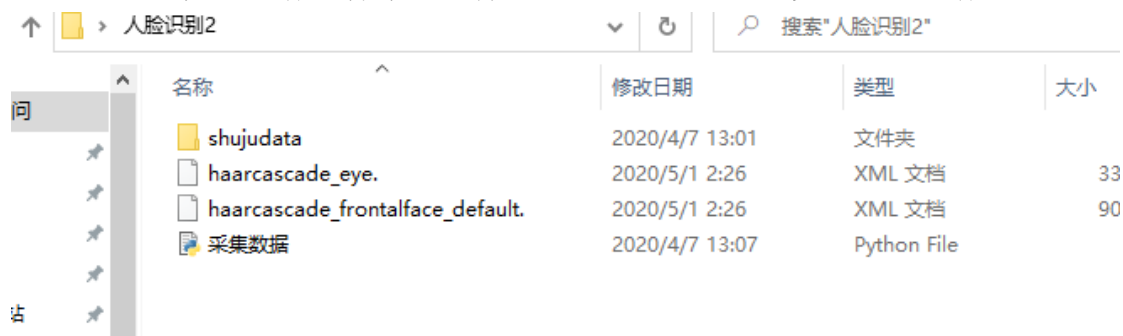


图 2.5 创建文件夹

✓ 然后运行采集程序，这里设定采集数量为 500，也就是采集 0-500 号数据图片，存放到 shujudata 文件夹中。经过大量的数据采集之后能提高人脸识别精度，所以结合家庭电脑工作速度我选择采集 500 张。部分代码如下：

```
sucess, img = cap.read()#摄像头工作照相
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)#转为灰度
```

```
faces = face_detector.detectMultiScale(gray, 1.3, 5)#人脸检测
for (x, y, w, h) in faces:
cv2.rectangle(img, (x, y), (x+w, y+w), (255, 0, 0))
count += 1
cv2.imwrite("shujudata/User." + str(face_id) + '.' + str(count) +
'.jpg', gray[y: y + h, x: x + w])#设定存储路径, 放入 shujudata 文件夹
cv2.imshow('image', img)
k = cv2.waitKey(1)
if k == 27:
break
elif count >= 500: # 设定为取 500 张样本
break
附上过程图:
```



图 2.6 人脸采集过程图

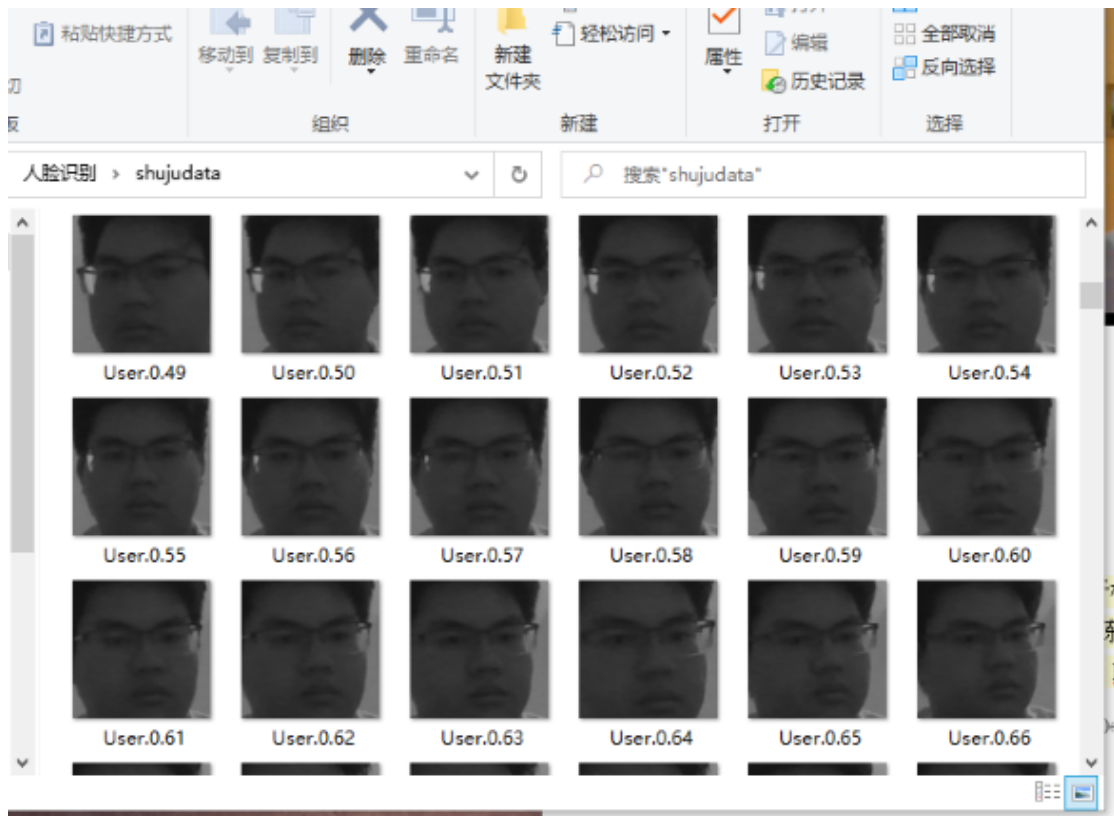


图 2.7 不同角度的采集图

### 2.3.3 人脸训练

根据上面采集得到的 500 样本，进行模型训练。这里主要运用 contrib 库中的 FaceRecognizer\_create 函数，在训练前先新建一个文件夹，用来存储训练完毕后得到的 trainer.yml 文件。这里的 FaceRecognizer 其实是现在主流的人脸识别方法的一个大类，主要包括基于 PCA 方法，基于 Fischer 变换和基于二值模式方法。我这里使用的方法的主导是基于 PCA 方法。部分代码如下：

```

path = 'shujudata' #上面所采集样本提取路径
recognizer = cv2.face.LBPHFaceRecognizer_create()
detector= cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
def getImagesAndLabels(path):
    imagePath = [os.path.join(path, f) for f in os.listdir(path)]
    faceSamples = []
    ids = []
    for imagePath in imagePath:
        PIL_img = Image.open(imagePath).convert('L') # convert it to
        grayscale

```

```

img_numpy = np.array(PIL_img, 'uint8')
id = int(os.path.split(imagePath)[-1].split(".")[1])
faces = detector.detectMultiScale(img_numpy)
    for (x, y, w, h) in faces:
        faceSamples.append(img_numpy[y:y + h, x: x + w])
        ids.append(id)

```

过程截图如下：

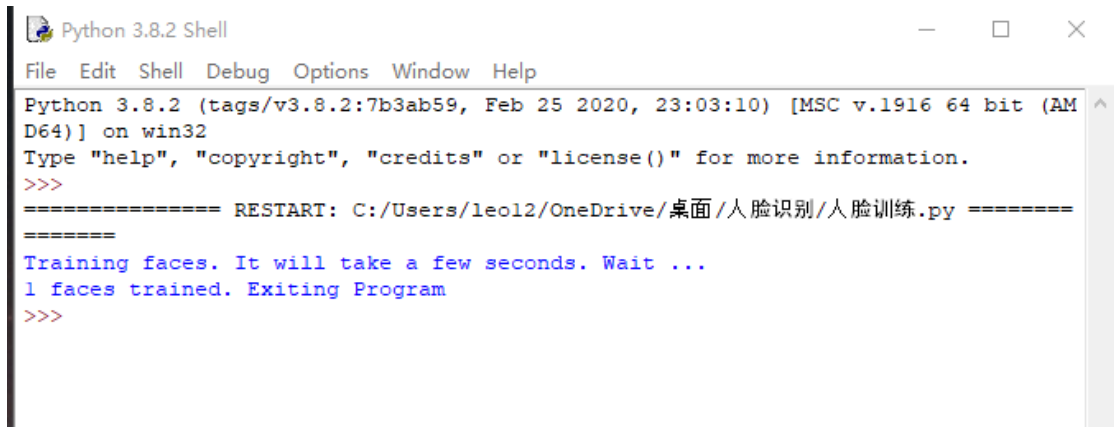


图 2.8 训练运行结果图

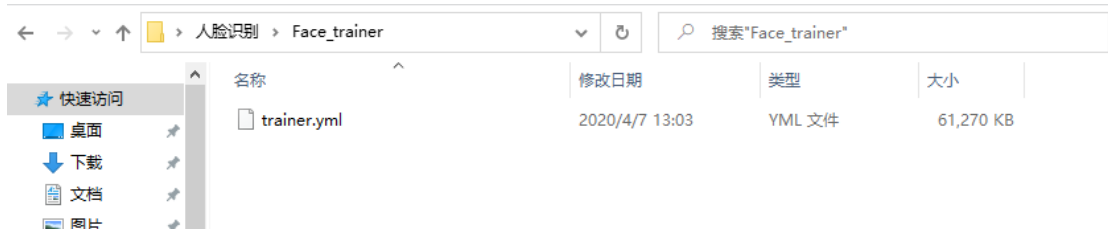


图 2.9 产生的训练 YML 文件

## 2.3.4 人脸识别

最后一步就是实现人脸识别的对比结果，得出当前识别人脸的正确身份。同样是采用 `cv2.face.LBPHFaceRecognizer_create` 语句，调用 `facerecognizer` 库，然后选择我们之前的训练库进行读写，然后根据现在显示的人脸与之前训练结果进行比较，得出识别结果。由于采用中文定义名字时会显示问号出错，所以这里我给识别结果命名为 `jie` (名字拼音)，左下角会有系统判断正确的概率的黑字体显示。部分代码如下：

```

recognizer = cv2.face.LBPHFaceRecognizer_create() #这里仍然调用各种库
recognizer.read('face_trainer/trainer.yml')
cascadePath = "haarcascade_frontalface_default.xml"

```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要  
下载或阅读全文，请访问：

<https://d.book118.com/708023000033006052>