

摘要

人脸识别是本世纪的一个热门技术话题之一。随着科技飞速发展。人脸识别技术已渗透进社会中的各个领域，例如：军事、医疗、教育、娱乐等。类似于人脸识别支付、人脸识别登陆、人脸识别认证等功能，我们几乎每天都在使用。

本设计主要运用 Python 语言、Opencv（计算机视觉库）、Tkinter(Python 的 Tk GUI 工具包)制作的一个含有人脸识别的一个信息系统。其工作原理为：以含有人脸的图片为素材，从图片中提取人脸数据的特征值作为识别的标准后，与人脸系统库中人脸比对，从而识别出图片中的人物，并执行与该人物相关的一系列功能。

关键词：人脸识别 Opencv GUI 检测

Abstract

Face recognition is one of the hot technical topics in this century. With the rapid development of technology, Face recognition technology has been in various fields in society, such as military, medical, education, entertainment, etc. Similar to face recognition payment, face recognition login, face recognition authentication and other functions, we use it almost every day.

This design mainly uses the Python language, Opencv (The vision library of computer), and Tkinter (Python Tk GUI toolkit) to design an information system containing face recognition. Working principle: It uses the picture containing the face as the material, extracts the feature value of the face data in the picture as the recognition standard, and then compares it with the face in the face system library. Thus, the person in the picture is recognized, and perform a series of functions related to the character.

Key words: Face recognition Opencv GUI Detection

目录

第一章 绪言	1
1.1 选题目的与意义	1
1.2 研究现状	2
1.3 主要内容	3
第二章 人脸识别的相关理论和技术	4
2.1 脸部识别原理	4
2.1.1 采集人脸图像	4
2.1.2 人脸检测	5
2.1.3 人脸信息对比（人脸识别）	5
2.2 人脸识别算法	6
2.2.1 几何特征识别法	6
2.2.2 特征脸识别法（PCA）	8
2.2.3 线段 Hausdorff 距离识别法（LHD）	9
2.2.4 支持向量机(SVM)的识别法	9
2.3 本章小结	9
第三章 人脸识别系统的软件设计	10
3.1 人脸识别的算法设计	10
3.2 软件功能的设计	12
3.3 软件功能的实现	13
3.4 本章小结	16
第四章 功能测试和系统调试	17
4.1 人脸识别系统的功能测试与结果	17
4.2 实验结果测试与对比	18
总结与展望	20
课题工作总结	20

未来展望	20
参考文献	21
致谢	22
附录	23

第一章 绪言

1.1 选题目的与意义

如今时代发展飞速，从汽车到火车，再从飞机到火箭，科技的进步令人瞠目结舌。从农耕社会到现在的信息社会，我们的生活也逐步迈向小康。但凡事都有其两面性。在科技进步下，带来了信息化的社会，但就是在这个信息化的社会下，防止信息的泄露也演变成了国际性的难题之一。

据不完全统计，全世界 80%以上的人，他们的个人信息都保存在服务器中；92%以上的企业信息都保存在服务器主机内。出于安全性的考虑，国家机密资料更是有很大一部分都需要用加密硬盘进行保存。这些数据可充分体现出人们对信息安全性的重视。

如今，人们都喜欢使用电子设备存储信息。第一、现在的电子设备越来越小巧，可随身携带；第二、电子设备体积虽小却能装载下大容量的东西；第三、电子设备不仅存储空间大，其存储的内容更是多种多样，不但可以存储文本、图片、视频，还可以存储三维模型、软件等。真因为电子设备的存在，同时也大大提高了信息在人与人之间传递的速率。从另一个方位思考，可发现这些信息同时也蕴含了极高的价值，例如个人的隐私信息、公司的机密信息、国家的机密信息等，所以，人们都很希望这些信息得到严密的保护，希望信息能够层层加密，就像是保险箱一样。也正因人们的需求太大，大大促进了如今的信息技术行业。

如今的加密方式的种类也很多，我们使用得最多的就是密码加密，这是利用了混淆技术原理^[8]，把一些可识别的信息经过某些手段处理后，变成无法识别的信息。这种密码加密方式仅适用于一些使用率较少的秘密信息，对于一些使用率较高的秘密就显得不太合适，因为这些秘密每天都要被使用，所以这样每天反复的输入密码，会显得十分繁琐，同时还加剧密码泄露风险。总而言之，类似于密码保护等身份识别工具已经无法满足社会的要求^[15]，研究新的身份识别工具是一个必然的趋势。本次毕业设计选择了人脸识别，以保证用户的真实性，从而进一步提高管理效率以及管理水平。

人脸识别技术早在 20 世纪 60 年代已有很多科学家在研究了。但是碍于当时的计算机处理速度，人脸识别这一技术才没办法发挥它的真正的“实力”。在 80 年代后，计算机的硬件方面大大提高，其处理速度方面也大大提升，人脸识别技术又被广大科学家重新研究，因此，人脸识别有了重大突破，进入真正的自动识别阶段。这一技术经过一段时间的成长后，在 90 年代后期，便涉及到各个领域，例如军事、教育、医学等。

从人脸识别的发展历程上可以看出，人脸识别技术的发展饱经风雨，但是也影响不了它飞速发展的速度。从 20 世纪 60 年代至今不过就是几十年的时间，人脸识别技术已从简单的二维平面模型识别到现在的三维立体模型识别、多姿态识别等。

人脸识别技术在学术上的解释是：人脸识别是基于人的脸部特征信息进行身份识别的一种生物识别技术。为什么能通过人脸识别某个人的身份呢？是因为每个人脸都有一个共性——都是由眼、鼻、口组成，并且它们的大致位置是不变的。但是在这些共性中又存在许多的不同点。例如：有些人是双眼皮有些人，是单眼皮；每个人的瞳距不一样；每个人嘴巴大小不一样；每个人鼻子的大小不一样等。就算是双胞胎都是会存在些许差别。人脸识别技术就是根据这些共性与差异辨别出身份的。

1.2 研究现状

人脸识别是基于人的脸部特征信息进行身份识别的一种生物识别技术^[16]。并且已涉及生活的各个领域，例如：军事、教育、购物等。其发展具有很大的潜力，就算在这个技术还未成熟的时候，在一些科技电影中就对这一项技术描述得淋漓尽致，也对其进行“华丽的预言”。正因这一技术的潜力极大，所以，我国也出台了许多相关政策支持这一技术的研发。

2015 年以来，我国相继出台了《安全防范视频监控人脸识别系统技术要求》、《信息安全技术网络人脸识别认证系统安全技术要求》等法律法规，为人脸识别技术在社会的发展，以及在各个领域的普及奠定了非常重要的基础^[1]。

在 2017 年的人工智能被写入全国政府报告的同年 7 月，国务院发布了《新一代人工智能发展规划》，同年的 12 月，工信部出台了《促进新一代人工智能产业发展三年行动计划(2018-2020 年)》^[1]。由于人脸识别也属于人工智能的一部分，所以在这几个出台的规划中，也对人脸识别的有效检出率以及正确识别率作出了一些标准规定。显而易见，人脸识别的发展是必然的，是这个时代的选择。

人脸识别最早在 20 世纪 60 年代就已有科学家在研究了，只不过是受制于当时的科技技术不够先进，计算机处理速度缓慢，才导致人脸识别这一技术迟迟无法面向社会。后来计算机的发展渐渐有了起色，人类也因此进入了计算机的信息时代，计算机的各个方面都突飞猛进地增长，有了良好的硬件基础后，人脸识别这一技术也伴随着计算机的增长而逐渐走向成熟。人脸识别的整个发展过程可以分为五个阶段，它们分别是机械识别阶段、半自动化识别阶段、非接触式识别阶段及互联网应用阶段。

与其他生物识别方式相比，人脸识别优势在于更具有自然性、具有不被察觉性等特点^[16]。

但与此同时人脸识别的研究难度也会比其它生物识别方式高。第一、人的脸部异变性比较大，换句话说就是外形的容易变化，容易影响识别的准确率。第二、环境的光度也会影响识别的准确率，由于如今的人脸识别大多数还是使用图像处理的方式识别，光的角度对于图像人脸的识别精确度的会有一定的影响。第三、人脸包含了许多姿态，例如开心、难过、憔悴、笑等，也给人脸识别造成了更大的研究难度。人脸识别其优势虽明显，未来必将会成为生物识别的主导技术之一，但其对于技术的要求也非常高。如果人脸识别在精度问题上能够得到有力的解决的话，人脸识别估计会替代指纹识别，成为市场应用规模最大的生物识别技术之一。

从申请专利数量来看，2007-2017年，我国人脸识别专利申请数量总体上呈上升的趋势，尤其是2014年后，计算机人脸识别专利申请数量大幅增多，到2017年，人脸识别专利申请数量达到2847项。

从公开专利数量来看，2007-2017年，我国人脸识别专利公开数量总体上呈稳定趋势。2017年，我国人脸识别专利公开数量为2698项，达到近年来最大值；截至2018年7月，专利公开数量为2163项。

从专利数据上可知，人脸识别技术的发展趋势一直在快速增长。同时也说明了，人脸识别技术还可能继续成为下一个世纪的热门技术之一。在市面上的与人脸识别技术相关的书籍、文献、资料也不少，所以当前人脸识别技术的研究现状是非常不错的。

1.3 主要内容

本文将设计人脸识别系统为主，对人脸识别技术进行深度的研究、学习。在设计开展之前，首先介绍一下人脸识别的相关知识、具体原理、以及算法知识。人脸识别算法有许多种，本文将不一一具体说明。在人脸识别算法知识中，会以阐述几何特征人脸识别算法为主。几何特征人脸识别算法也是本次设计的核心。除此之外，本文还会着重对以下几点进行了较为深入的探讨：

- (1)采集人脸图像过程
- (2)人脸检测过程
- (3)人脸信息对比过程

第二章 人脸识别的相关理论和技术

2.1 脸部识别原理

计算机识别人脸与人识别人脸的方式存在很大的差异。人是通过肉眼的视觉影像获取到这个人的脸部信息，进而识别出这个人的身份信息。如果是个双胞胎的话，还可以通过这个人的肢体动作感知来区分。但对于计算机而言，就会显得相对困难，因为计算机不像人一样有感知，计算机需要将这些信息转化成数据，再对这些数据进行一系列的处理后，才能达到识别的效果。所以，人脸识别会涉及许多领域的知识点，其中最主要的就是图像处理。

计算机需要通过大量数据才能识别出一个人脸。在这个地球上一共有 70 多亿人，因此一个精确的人脸识别背后需要大数据的支撑。在人脸识别算法中，有一个基于卷积神经网络人脸上识别算法，它拥有极其庞大的人脸库，涉及到大数据、深度学习、以及人工智能等相关知识。

经过多重考察后，最后选用了 Opencv 作为设计的工具。因为利用 Opencv 对图像操作非常方便。Opencv 可对图像进行剪辑、加载、拷贝等操作，还可以对图像进行基本的处理、例如去噪点、色彩处理等。不仅如此 Opencv 里还有摄像头的功能，方便本次设计制作以视频流为基准的人脸识别。而且 OpenCV 还可以进行一些线性代数上的计算，以内置函数模块实现计算特征值、特征向量、特征矩阵等。OpenCV 还提供了图像连通域、霍夫变换、图像轮廓处理、连通分支的分析、多项式的逼近、图像的模板匹配、曲线拟合、狄劳尼三角化等目标识别，以及多种特征提取方法、HMM 模型^[4]。

人脸识别的过程主要分为采集人脸图像、人脸检测、人脸信息比对三个过程。

2.1.1 采集人脸图像

采集人脸图像的渠道有很多种（下面列举三种常用的）：

- ①利用电子机器摄像头拍摄
- ②互相网上的人脸识别库
- ③扫描仪

采集对象是一些含有人脸的 2D 图像、3D 图像、视频等。不同的类型的图像，识别的方式也会有所不同。

例如：3D 人脸图像识别就有两种，一种是在三维的基础上，另一种则是在 2D 图像的基础上。3D 图像识别之所以可以在 2D 图像识别的基础上识别，是因为 3D 图像只是比 2D 多出一个维度的数据，所以可以通过处理 2D 图像再加上一个深度数据的方式来识别 3D 的人脸图像。以目前的调查数据显示，3D 的人脸识别准确度会比 2D 的人脸识别准确度会更精准。

2.1.2 人脸检测

人脸检测是人脸识别的前提和基础^[12]。这个过程主要检测的是图像中是否含有人脸。由于人的脸部器官之间大致位置关系是固定的，并且这些脸部器官也会有所差异，例如瞳距、大小、角度等。可以通过这些共性与非共性的集合，在人脸图像寻找出一个特征点，通过这些能够标识人脸的特征点数据追踪到人脸图像中的人脸位置，人脸图像是由多个像素点组合而成，计算机一般会用数据去记录每个像素点，对这些像素点进行分析之后就可以得到人脸数据特征矩阵，从而得到特征值与特征向量等数据。如今的人脸识别技术已成熟，所以市面上已有不少人脸检测工具，Opencv 就是其中之一，在人脸检测这一模块可以使用 OpenCV 中的人脸模型去定位人脸位置之后再对人脸进行进一步的解析。但需要注意，在图像检测中，一般会先使用灰度处理过图像后，再进行人脸检测，因为这样能够提高人脸识别的性能。当然，彩色也是可以的。本次设计将使用的是灰度处理过的图像进行识别。

人脸检测这一部分主要有图像的预处理、特征提取与选择两部分。

①预处理

影响图像的因素会有很多，例如：光线亮度、拍摄角度等。当图像被影响了之后就容易产生许多噪点，这些噪点会影响我们人脸识别的各种检测。所以图像处理是很有必要的。一般是把图像中的噪点去除掉，还有把图像灰度化。

②特征提取

特征提取是整个人脸识别里面很关键的一部分。很多识别方法都是以特征点为参照的基准才能达到检测/识别的效果。在 OpenCV 里，含有许多器官的特征值，利用这些特征值可以定位出图像中是否含有人脸，如果含有人脸的话，就会返回给系统其上下左右四个点的位置。再利用这四个点就找出人脸的具体位置了。

2.1.3 人脸信息对比（人脸识别）

仅仅识别出人脸还不够，还需要识别出具体的人物。所以需要人脸信息对比。人脸识别肯定离不开脸部识别算法。在开发设计过程中，可以自己设计属于自己的人脸识别算法，但这样设计难度也会随之增加，也可以利用开源的人脸识别库，调用其中的脸部算法函数，这样会方便自己的初次学习，而且自己写的脸部识别算法还可能不如别人的算法好，导致系统设计缺陷多。而且人脸识别的算法质量对人脸识别的准确率会有一定的影响。

人脸识别的算法有很多，主要的就是几何特征识别、以统计学为原理的 PCA 识别算法、神经网络识别算法、局部特征算法、弹性匹配算法等。本次人脸识别设计采用的算法是几何特征识别算法，利用 OpenCV 与 face_recognition 的结合制作的人脸识别系统。

这部分主要检测图像中的人脸到底是谁。如果要应用在市场应用上，在这个过程中需要建立一个人脸数据库。在图像检测完毕后，将图像中的人脸与人脸数据库的人脸进行比对，从而识别出这个人脸。一般人脸信息相似度不会设置为百分之 100，脸部表情也会影响人的脸部的相似度，一般设置相似度为 60%左右。

2.2 人脸识别算法

在众多人脸识别算法中，特征分析也是人脸识别的核心之一。经过多个的人脸模型的训练后，所有的特征点组合起来后就能建立出人的脸部模型。通过这个脸部模型就可以定位到人脸图像中的人脸位置。但是还不能识别出这个人脸到底属于谁，还需要对识别出来的人脸作进一步再分析，以某一相似值为标准，与人脸库中的人脸信息作对比。

其中，主要的识别算法有：几何特征识别法、局部特征识别法、特征脸识别法（PCA）、神经网络识别法（CNN）、弹性匹配识别方法、线段 Hausdorff 距离(LHD)识别法、支持向量机(SVM)的识别法^[2]。

2.2.1 几何特征识别法

基于几何特征的提取方法主要是人脸的主要器官进行定位^[3]。由于脸部器官的结构、大小等的不同，所以可以对脸部以几何进行描述，构成脸部的几何特征。这些几何特征主要包括特征点的位置、特征点之间的距离、特征点之间的角度等^[3]。通过对这些特征点进行解析，利用特征点与特征点之间的距离构造出特征向量^[16]，再利用该特征向量进行人脸识别^{[11][15]}。

本次设计的特征值提取是基于灰度图像的一个特征值提取。

假设输入的图像 $g(x,y)$ ，其大小为 $e * f$ 。图像 g 在点 (x,y) 的灰度值 $I(x,y)$ 。那么，该图像的 $V(x)$ 垂直积分投影函数， $H(y)$ 平积分投影函数如下^[5]：

$$V(x) = \frac{1}{y_2 - y_1} \sum_{y_1}^{y_2} I(x, y), \quad x \in [1, e] \quad (\text{公式 2-1})$$

$$H(y) = \frac{1}{x_2 - x_1} \sum_{x_1}^{x_2} I(x, y), \quad y \in [1, f] \quad (\text{公式 2-2})$$

垂直积分投影函数是指输入图像的一个垂直区域的灰度值之和，它表示图像的灰度值在垂直方向上的变化。水平积分投影函数则是指输入图像水平区域的灰度值之和，表示图像的灰度值在水平方向上的变化。

这是一个经过灰度处理后的人脸图像（如图 2.1）：



图 2.1 灰度处理后的人脸图像

其垂直积分投影函数如下图所示：

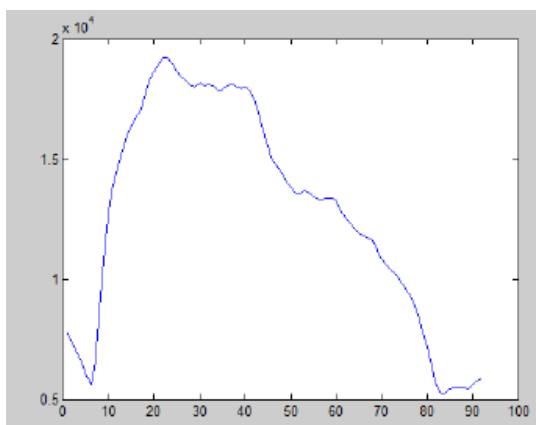


图 2.2 垂直积分投影函数

其水平积分投影函数如下图所示：

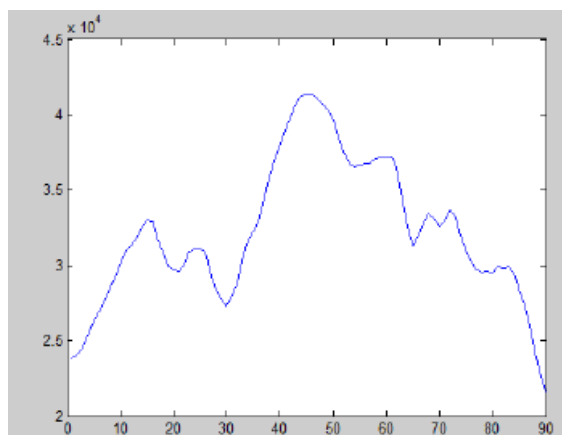


图 2.3 水平积分投影函数

像素点越暗，其灰度值就越低。相反，像素点越亮，其灰度值越高。

根据垂直积分投影函数与水平积分投影函数，可以定位出人脸部的眼睛位置。因为人的眼睛一共有两个，并且具有对称性，所以在水平积分投影函数的两个波谷就是眼睛的位置的。紧接着就是定位出鼻子的位置，鼻子在两个眼睛的中间，并且灰度值比较高，也就是灰度值最大的那个点，就是脸部的鼻子的特征点。紧接着就能定位出嘴巴的位置，就是在积分函数投影的灰度值比较小，并且在眼睛附近的地方。每个脸部器官都取 3 个特征点，共 12 个，并且构造出其标准特征向量。

2.2.2 特征脸识别法（PCA）

PCA（Principal Component Analysis）方法也就是主成分分析法。该方法是一种以 K-L 变换为基础的统计分析方法^[5]。每一个人脸都是由许多成分组成的，例如：痘痘，皮肤，黑色素等，在这些成分中有些成分主要的，例如：眼睛形状，嘴巴的形状等，并且这些主成分组合起来可以唯一标识出特定的人，在这种情况下，就可以不加以考虑除了主成分之外的元素，这样可以大大减少程序的运算量。在识别人脸的过程中，首先要把主要人脸部分截取出来，再进行识别。截取出来的人脸图像可以看作一个高维度数据的产物，通过对这个高维数据进行分析得到一个低维数据，方便程序计算。虽然在这个过程中图像存在细微的亏损，但是其主成分是不发生改变的。利用这些主成分特征值与现在的人脸对比，进而达到识别的效果。

在这一算法中，图像训练是这一算法中很重要的一部分。

多个低维数据，可以反推出一个标准的新的的人脸图像（如图 2.4），根据这个图像就能准确地识别出当前人脸对应人脸库中的哪个人脸，进而达到识别的效果。



图 2.4 反推的新的的人脸图像

PCA 的主要步骤如下：

- (1) 先把人脸数据样本截取出来。
- (2) 将人脸图像的高维图像矩阵进行排序、计算，得出低维的数据矩阵。

计算公式如下^[5]：

$$\mu = \frac{1}{N} \sum_{i=1}^N X_i$$

（公式 2-3）

(3) 紧接着计算出其协方差矩阵，公式如下^[5]：

$$S = \frac{1}{N} \sum_{i=1}^N (X_i - \mu)(X_i - \mu)^T \quad (\text{公式 2-4})$$

(4) 将计算出来的 K 个最大特征值与其对应的特征向量进行组合，变成新的变换矩阵，接着采用 K-L 变换式对计算出来的变换矩阵进行计算，就可以得到该图像的低维特征向量。

(5) 最后利用这些低维数据反推出新的图像投影到子空间中，就可以得到该人脸的人脸模型，利用这个模型的数据就可以达到人脸识别的目的。

2.2.3 线段 Hausdorff 距离识别法（LHD）

心理学研究表明，人在识别类似素描的边缘图上的速度和准确度丝毫不比识别灰度图的能力差^[13]。LHD 识别就是基于人脸图像的灰度图轮廓，识别的是两个线段集合之间的距离，通过计算 Hausdorff 距离进行识别两张人脸之间的相似度^[14]，再根据相似度的大小来判断是否为同一个人。由于线段集之间的线段不存在对应关系，所以，LHD 识别对微小变化的处理远比其他识别算法要好，但是其缺点是在大表情下的处理会存在较大的缺陷。在本次设计中没有用到此算法。

2.2.4 支持向量机(SVM)的识别法

支持向量机（Support Vector Machine，SVM）目前是 Corinna Cortes 和 Vapnik 等人在研究统计学习理论的基础上提出的新的方法^{[6][10]}。SVM 算法主要分为线性与非线性两种方法，在进行识别时基本上不受样本的质量与数量的影响^[9]。通过在训练集合里提取出特征向量值，结合一定的算法得到结果。在本次设计中没有用到此算法。

2.3 本章小结

本章主要概述了人脸识别的具体过程和不同人脸识别算法的原理。人脸检测是所有算法的前提、基础。一般情况下，都是检测出人脸位置，以该图像区域为基础进行人脸识别。不同的人脸算法的依据也是不一样的，例如：PCA 识别是依据统计学的，线段 Hausdorff 距离识别是依据心理学的某些原理。不仅如此，由于原理的改变，其运算的时长也不一样，所需的空间也会不一样。综上所述，要想设计出一个性能较好的人脸识别系统，必须完善其人脸识别算法，可通过研究不一样的算法，综合其优点改善。

第三章 人脸识别系统的软件设计

3.1 人脸识别的算法设计

本次设计尝试了 2 种人脸识别方法。一种是 PAC 算法，另一种几何特征算法，这两种算法都有其优点和缺点，最后经过多次考量之后，决定选择用简单的几何特征算法。因为 PAC 需要训练大量的人脸图像训练样本，经过统计、分析之后，得到一个人脸模型，再由这个人脸模型与当前人脸对比，才识别出人脸。这个过程相对复杂，导致最后在嵌入进窗口界面的时候，容易出现卡顿严重，所以最后选用的是几何特征识别的方式。

本次人脸识别系统设计是主要基于 Opencv、face_recognition 两个库。

几何特征识别流程（如图 3.1）：

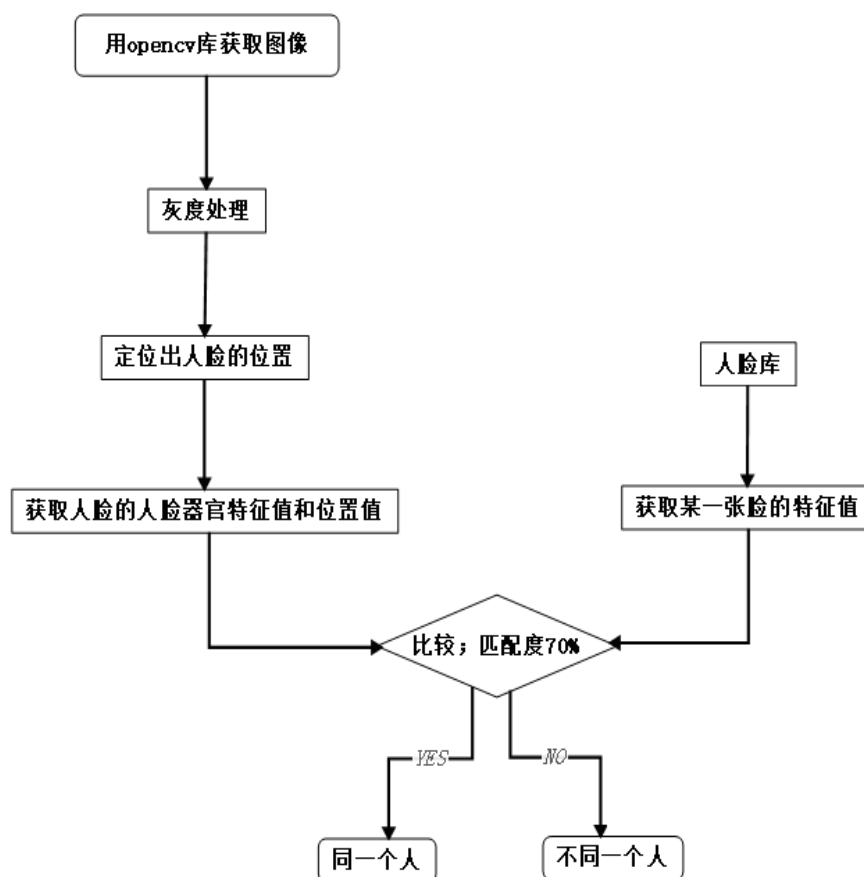


图 3.1 几何特征识别流程

PCA 算法识别流程：

在用 PCA 识别算法中，可使用 sklearn 模块进行计算，这样会大大提高识别的效率。

①降维过程（如图 3.2）：

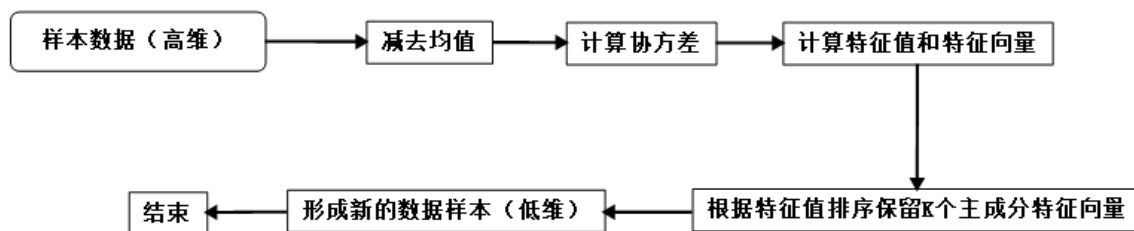


图 3.2 降维过程

②训练与识别流程（如图 3.3）：

在用 PCA 识别算法的时候，可使用 sklearn 模块进行计算，这样可以大大提高识别的效率。

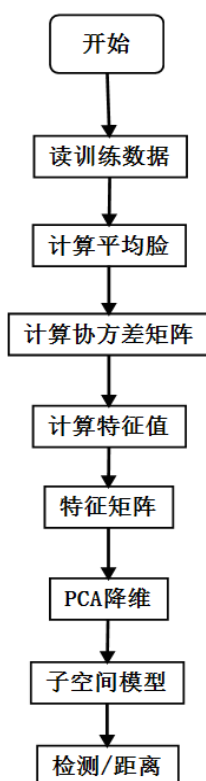


图 3.3 训练与识别流程

（1）PCA 识别样本训练的部分代码：

```

# 获取训练集
train_face, train_face_number, test_face, test_face_number = loadDataSet(3)
# PCA 训练训练集，用 pca 将数据降到 30 维
pca = PCA(n_components=30).fit(train_face)
# 返回测试集和训练集降维后的数据集
x_train_pca = pca.transform(train_face)
x_test_pca = pca.transform(test_face)
# 逻辑回归训练
  
```

```

classirfier = LogisticRegression()
lr = classirfier.fit(x_train_pca, train_face_number)
# 保存模型
joblib.dump(lr, 'lr.model')
# 计算精确度和召回率
accuray = classirfier.score(x_test_pca, test_face_number)
recall = accuray * 0.7
    
```

（2）PCA 识别的部分代码:

```

# 预测的人脸
predict = img2vector(string)
# 加载模型
LR = joblib.load('lr.model')
predict_people = LR.predict(pca.transform(predict))
string1 = str("编号: %s 精确度: %f 召回率: %f" %
              (predict_people, accuray, recall))
    
```

3.2 软件功能的设计

本次人脸识别系统的功能主要以登陆为主，分为密码登陆与人脸登陆两个部分。

密码登陆部分流程（如图 3.4）：

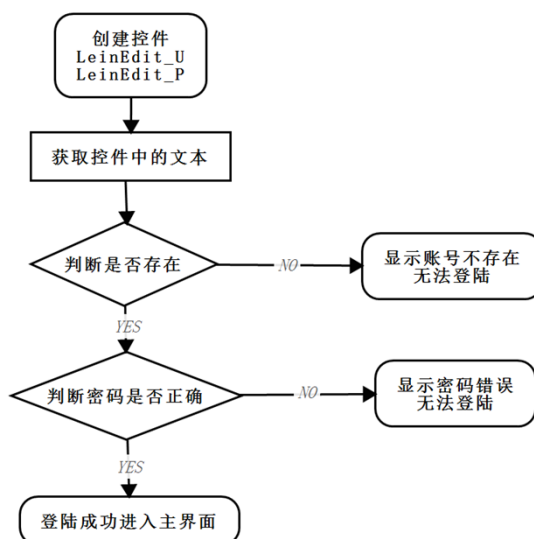


图 3.4 密码登陆部分流程

人脸登陆部分流程（如图 3.5）：

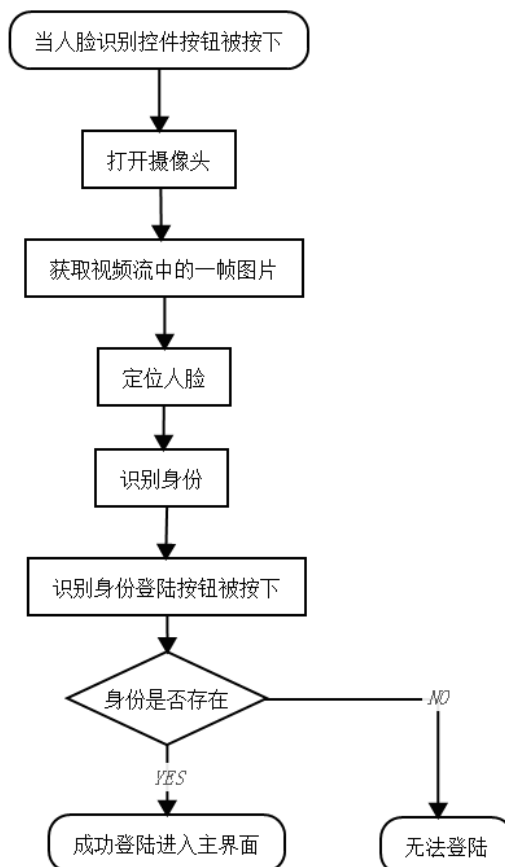


图 3.5 人脸登陆部分流程

3.3 软件功能的实现

详细代码见附录。

第一步：环境配置

下载 PyCharm，配置好 python3.6 的环境。

安装 opencv 库、tkinter 库、face_recognition 这些库的环境安装相对复杂，需要的时间会比较长。

第二步：各个子模块准备（人脸定位、人脸识别、界面设计、摄像头）

人脸定位子模块：

利用 opencv 库加载一张含有人脸的图像，将图像里的成分进行解析，利用人脸器官的特征值，定位出人脸的大致位置后，对图像进行标记，最后显示标记过的人脸图像，达到人脸定位的效果，具体的人脸定位过程（如图 3.6）。

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/427022034042006065>