 **哈尔滨理工大学**

**HARBIN UNIVISITY OF SCIENCE AND TECHNOLOGY**

**技 能 训 练 Ⅱ**

**实 践 报 告**

**实践名称： 人脸指纹混合识别系统设计**

**学院名称： 自动化学院**

**专业班级： 机器人21-1**

**姓名学号： 李嘉昱 2112010209**

**刘子硕 2112010215**

**陈冠宇 2112010901**

**刘晴 2102070112**

**余凯 2112010421**

**指导教师： 曹宇 刘美军**

**成 绩 评 定**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **考核**  **内容** | **人员**  **名单** | **实践**  **表现** | **实践**  **报告** | **实践成果**  **或答辩** | **综合评**  **定成绩** |
| **成绩** |  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**2023 年 12 月 29 日**

**实践纪律要求和成绩考核办法**

1．实践期间必须听从教师指导，严格遵守安全操作规程。不准违规操作，未经指导教师允许不准启动任何非自用设备、仪器、工具等；操作项目和内容必须按实习要求进行，特别要注意防止电烙铁烧烫伤、不用时要及时切断电源。

2．实践教室内不准吸烟、吃食物（含零食），不准带无关人员到实习教室活动，否则扣平时表现分。

3．参加本次实践时间不足三分之二或旷课1天以上者，不得参加本次考核，按不及格处理。

4．病事假必须有请假条，需经导员或有关领导批准，否则按旷课处理。

5．实践过程考核和实习成绩在教师手册中要有记载。

6．实践成绩的考核由指导教师根据实践表现（出勤、遵守纪律情况等）、实践报告（格式规范、叙述详实）、实习成果、现场操作、口试或笔试等几个方面，给出各项成绩或权重，综合后给出实践总成绩。

7．成绩评定采用五级分制，即优、良、中、及格、不及格。

8．实践结束一周内，指导教师提交实践成绩和实践总结。

目录

[《人脸指纹混合识别系统设计》实践报告](#_Toc731124385)

[一、实践目的与意义](#_Toc200113979)

[1.1人脸和指纹混合识别算法分析对比](#_Toc618110197)

[1.2人脸和指纹混合识别目的及意义](#_Toc180825386)

[二、 国内外研究现状](#_Toc1342578883)

[三、实践内容](#_Toc1736378877)

[3.1图像预处理](#_Toc1092021081)

[3.1.1功能分析和方案论证说明](#_Toc965168351)

[3.1.2人脸和指纹图像预处理程序设计](#_Toc2118944626)

[3.1.3程序模块与调试说明](#_Toc1969801197)

[3.1.4存在的问题和解决措施](#_Toc2067729630)

[3.2人脸检测](#_Toc260522057)

[3.3人脸和指纹特征提取](#_Toc1695686220)

[3.3.1指纹特征提取](#_Toc372512781)

[3.3.2人脸特征提取](#_Toc572749973)

[3.4人脸和指纹混合匹配识别](#_Toc1090032933)

[3.4.1指纹特征点匹配](#_Toc710947721)

[3.4.2人脸特征点匹配](#_Toc321200962)

[3.5 GUI界面设计](#_Toc547555450)

[四、实践总结](#_Toc252021242)

[五、主要参考文献](#_Toc2100996473)

[附件：](#_Toc1065419494)

《人脸指纹混合识别系统设计》实践报告

一、实践目的与意义

1.1人脸和指纹混合识别算法分析对比

（1）卷积神经网络（CNN）是人工神经网络（ANN）和人工智能发展的突破之一。它是深度学习中最流行的算法之一，深度学习是一种机器学习，模型学习直接对图像、视频、文本或声音执行分类任务。该模型在多个领域显示出令人印象深刻的结果：计算机视觉、自然语言处理 (NLP) 和最大的图像分类数据集 (Image Net)。CNN 是一个普通的神经网络，带有新的层——卷积层和池化层。CNN 可以有几十个和几百个这样的层，每个层都学会检测不同的成像特征。

（2）特征脸：是一种人脸检测和识别方法，用于确定图像数据集中的人脸方差。它使用这些差异通过机器学习对人脸进行编码和解码。一组特征脸是通过对大量人脸图像的统计分析确定的“标准化人脸成分”的集合。面部特征被分配了数学值，因为这种方法不使用数字图片，而是使用统计数据库。任何人脸都是这些值以不同百分比的组合。

（3）LBPH：LBPH算法即局部二值模式算法，是一种用来识别图像中人脸的算法，由于其简单易懂，被广泛应用于人脸识别领域。LBPH算法将图像划分为若干个小块，通过对这些块的像素值进行特征提取和组合，得到一个二进制数，用于表示这个图像块的特征。基于这些特征，进行人脸分类。

1.2人脸和指纹混合识别目的及意义

（1）目的：人脸指纹混合识别旨在提高身份认证系统的准确性和安全性。传统的人脸识别和指纹识别技术各有其局限性，比如人脸识别可能受到光线、角度和遮挡等因素的影响，而指纹识别可能受到指纹质量和损伤的影响。然而，通过结合人脸识别和指纹识别技术，可以在一定程度上克服各自的缺点，提高识别的准确性和鲁棒性。

人脸指纹混合识别可以利用人脸和指纹两种生物特征的互补性，增加多个特征源的信息，从而提高识别的可靠性。通过同时采集和比对人脸和指纹数据，可以大大降低识别的误报率和漏报率，增强系统的安全性。此外，在一些高安全性要求的场景下，比如边境检查、金融机构等，人脸指纹混合识别可为身份认证提供更强的验证手段，防止欺诈和冒充行为的发生。

总的来说，人脸指纹混合识别的目的是为了提高身份认证系统的准确性、安全性和可靠性，以适应多种复杂应用场景的需求。

（2）意义：人脸和指纹混合识别是一种生物特征识别技术，它综合利用了人脸识别和指纹识别两种技术的优势，旨在提高身份认证的准确性和可靠性。这种识别技术的目的是识别身份，以确保安全和防止欺诈行为。

在实际应用中，人脸和指纹混合识别可以被广泛运用于身份验证和门禁控制等场合。例如，在银行等金融机构，采用这种生物特征识别技术可以大大增强客户的账户安全性，有效地防止账户被盗刷；在企事业单位的门禁管理中，也可以通过人脸和指纹混合识别技术对员工进行身份验证，确保办公区域的安全。因此，人脸和指纹混合识别可以提高安全性和便利性，为人们的生活带来更多的便利和保障。

1. 国内外研究现状

2.1人脸识别国内外研究现状：

人脸识别技术是一种通过从图像或视频中提取和识别人脸的特征，并将其与数据库中的样本进行比对来验证或识别身份的技术，近年来随着计算机视觉和深度学习技术的快速发展，人脸识别技术得到了广泛的应用和发展国内方面，中国政府在2015年提出了“人工智能+”战略，将人脸识别技术列为其中的重点领域之一。随着政策的支持和资金的投入，国内的人脸识别技术得到了快速的发展，应用场景也不断扩大。目前，国内的人脸识别技术已广泛应用于公安、金融、零售、交通、教育等领域，成为了智慧城市建设和社会治理的重要工具。

国外方面，人脸识别技术的应用也非常广泛。在西方国家，人脸识别技术主要应用于安全领域，如机场、边境、监狱等场所的安全检查和监控。此外，人脸识别技术也被应用于零售、金融、医疗等领域。在发达国家，人脸识别技术的应用已经进入到智慧城市、智能家居等领域。

2.2指纹识别国内外研究现状：

目前，指纹识别的研究现状主要包括以下几个方面：

1. 基本概念和方法：指纹识别是通过分析和比对指纹图像中的纹线和纹谷等特征来进行身份认证的技术。常用的指纹识别方法包括特征提取、特征匹配和决策等步骤。

2. 低质量指纹的识别：对于低质量指纹的识别率仍然有待提高。尤其是在公安领域，由于指纹采集条件的限制，低质量指纹较多，对它们的识别率需要进一步改进。

3. 大规模指纹数据库的管理和查询：随着指纹识别技术的广泛应用，大规模指纹数据库的管理和查询成为一个重要的问题。如何高效地存储和检索大量的指纹数据，是当前研究热点之一。

4. 多模态生物特征融合：为了提高识别的准确性和可靠性，研究人员开始将指纹识别与其他生物特征识别技术进行融合，如人脸识别、虹膜识别等。这种多模态生物特征融合的方法可以提高识别的鲁棒性和安全性。

5. 深度学习在指纹识别中的应用：近年来，深度学习技术在图像识别领域取得了显著的进展。研究人员开始探索将深度学习应用于指纹识别中，以提高指纹识别的准确性和鲁棒性。

三、实践内容

训练图片集

图像

分割

特征

选择

特征

信息

特征

选择

训练

结果

分类

结果

测试图片集

生成灰

度矩阵

生成特

征矩阵

生成特

征向量

分类器

训练

分类器

信息

分类器

运算

预处理

特征选择

训练、分类

图像

分割

生成灰

度向量

图1 人脸和指纹识别系统体系结构

3.1图像预处理

3.1.1功能分析和方案论证说明

（1）功能分析

设计制作任务：设计和制作一个指纹人脸检测系统需要考虑到传感器的选择和集成、算法的设计和优化、用户界面的设计、系统的联网和数据管理，以及系统的性能和稳定性测试。只有在这些方面都得到仔细考虑和设计，才能制作出一个高效、稳定和可靠的指纹人脸检测系统。

功能及要求：指纹人脸混合检测功能的设计和开发需要综合考虑传感器集成、算法设计、用户界面设计、数据管理和安全性，以及性能和稳定性测试。只有在这些方面都得到仔细考虑和设计，才能实现一个高效、稳定和可靠的指纹人脸混合检测功能。

1. 方案论证说明

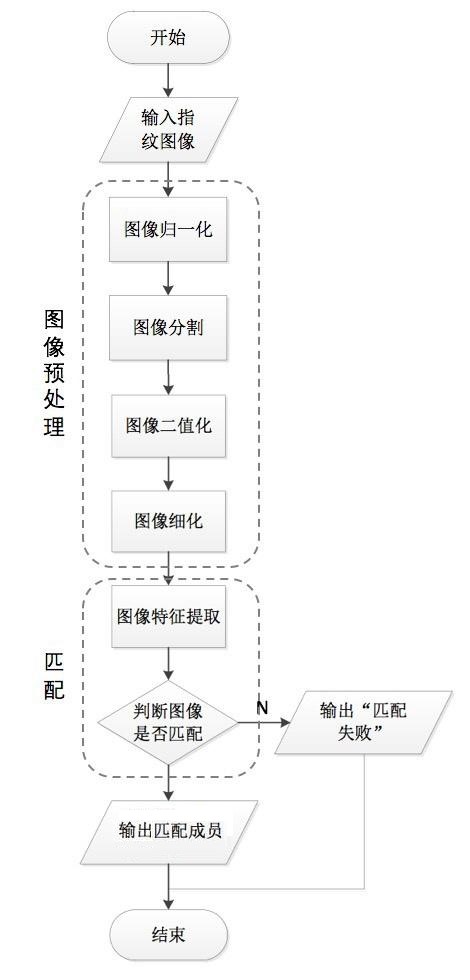


图2指纹识别流程图 1

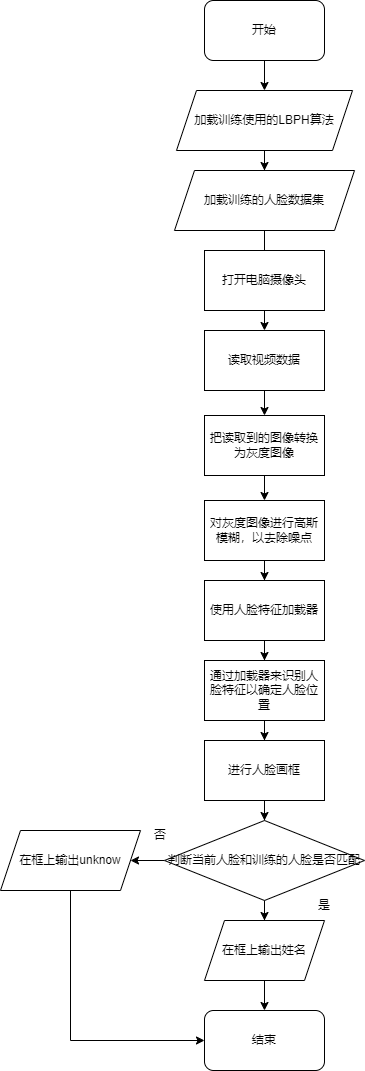


图3人脸识别流程图 1

3.1.2人脸和指纹图像预处理程序设计

1.指纹图像预处理

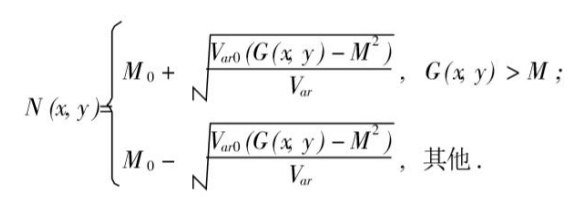
（1）图像灰度的规格化和指纹与背景的分割

因为各种指纹图像的对比度，图像灰度的差别，我们需要将上述做图像规格化处

理，处理成在一定灰度范围内的图像。

· 图像灰度规格化的方法如下：

对图片中的每一个像素作如下的公式转换：



其中，N(x, y)是新生成的像素灰度值，G(x, y)是原像素的灰度值，M0 和 Var0 分别是

整个图像灰度值的均值与方差的期望值，这里分别取固定值 150 和 2000，而M 和 Var 分别是整个图像灰度值的实际均值与方差。由于在指纹图像中，往往背景是作为不需要的部分，因此，我们需要将指纹图像与背景分割开来。

· 指纹图案与背景分割的方法如下：

将整个图像分割成若干个 24\*24 的小块区域，对每块区域中的所有像素的弧度值做方差运算，如果方差小于一定阈值，就代表这块区域的灰度值区别不大，是背景，于是就将这块区域的所有像素设置成白色。

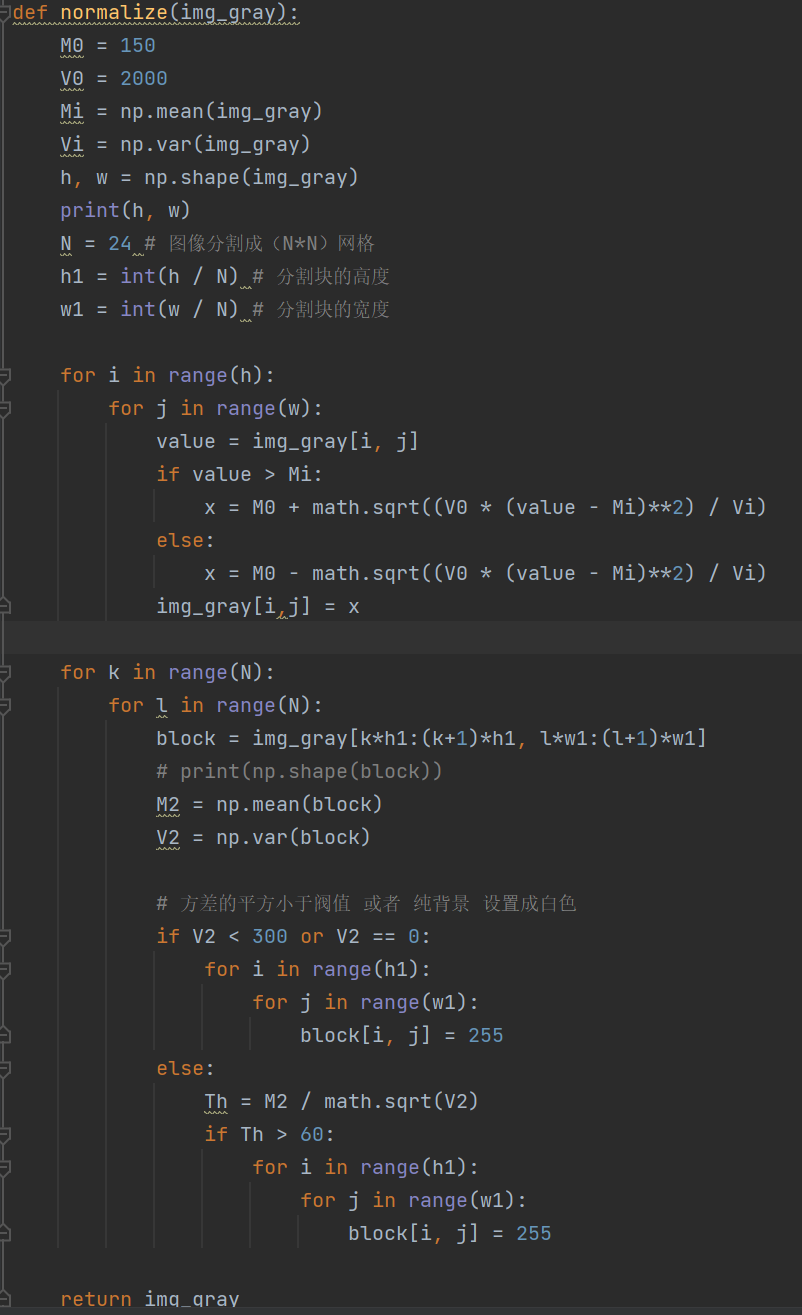


图4指纹图像预处理代码1

（2）图像的二值化

·基于方向场的二值化

指纹图像一般都有着清晰的方向场，如果我们能够利用这种方向性，对图像进行方向滤波处理，就能得到较为清晰纹路图像。

·基于方向场的二值化方法：

首先，对整个图像进行 3\*3 的均值滤波，然后对每个像素的八邻接块的八个方向进行灰度值的累加计算，然后去掉最大值 summax 与最小值 summin，如果满足 summax+summin+4\*(该像素的灰度值) > (3\*summ/8)，其中 summ 为所得八个方向灰度累加值之和，则该脊线方向为 summin 所对应的方向，否则为 summax。

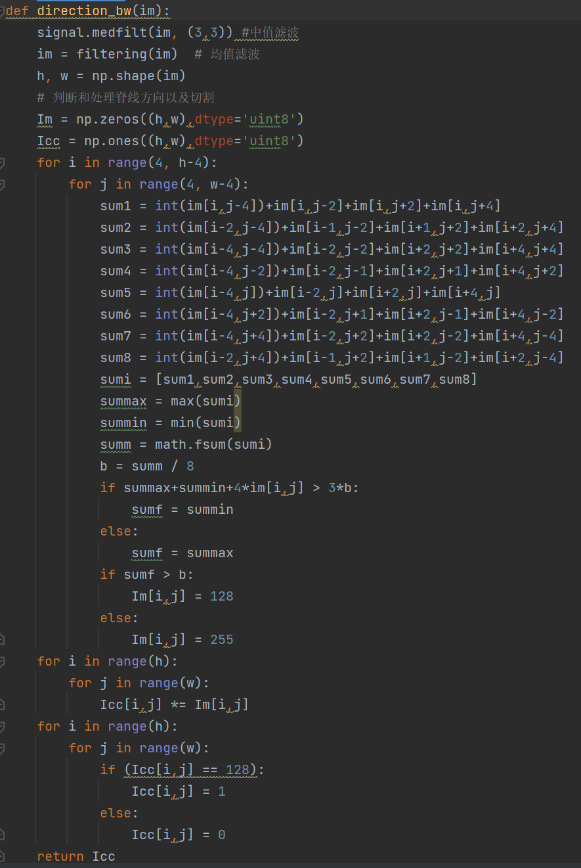


图4指纹图像预处理代码2

（3）去除指纹中的空洞和毛刺

原指纹在进行二值化后，指纹上的空洞和毛刺等一些为伪特征点在抽取特征点的阶段都有可能被错误地抽取为特征点，从而影响特征的匹配。因此，去除指纹上的空洞和毛刺就变得十分必要。空洞的判断条件是：如果该点的背景在二值化后是白色，即图像矩阵上的值为 1，并且八领域的值和为 0，即八零域的点均为黑色，这样的点就是空洞。毛刺的判断条件是：如果该像素点的值为 0（黑色），并且该像素点的上下左右的像素点的值加起来大于 3，即上下左右的像素点中有多于 3 个是白色，就是毛刺。去除空洞只要让该像素点变为黑色即可，而去除毛刺只要将该像素点变成白色即可。

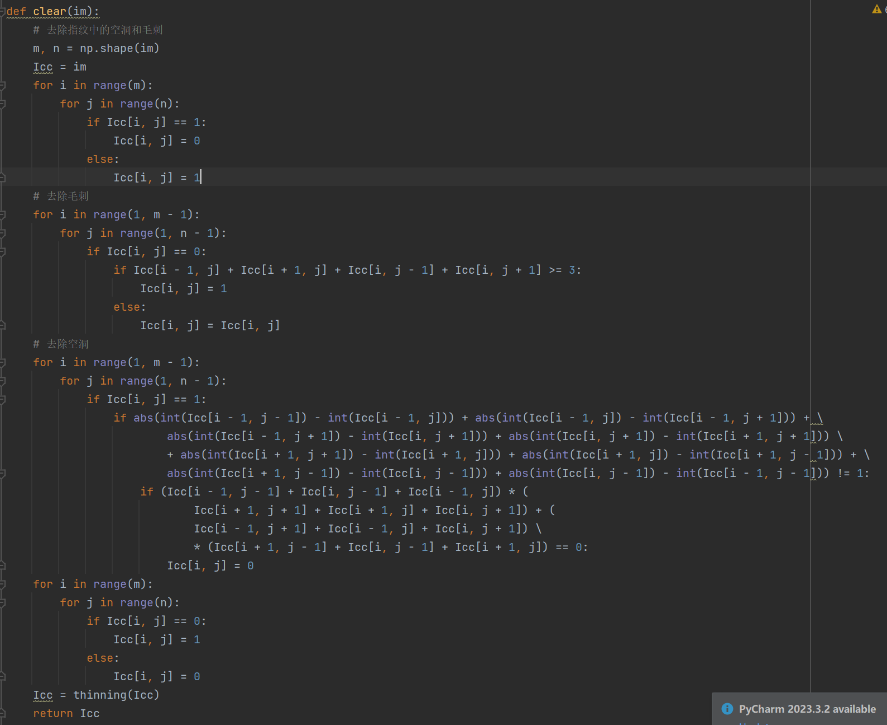


图5指纹图像预处理代码3

（4）图像的细化

在去除了空洞和毛刺之后，有可能还存在很多小的由图像形成的斑点。同时，可能因为前面去除空洞和毛刺的过程导致很多本来连接的部分被误分成许多小块的对象。在此，采用形态学中的开闭运算进行处理。开运算的过程：先定义一个3\*3 的矩形结构元素，然后对图像的每个像素点进行腐蚀操作，即让每个结构元素和与其覆盖的二值图像的 3\*3 子块进行与运算，如果结果的矩阵里的元素都为1，则该像素点的值就为 1，否则置为 0。然后再对每个像素点进行膨胀操作，即用结构元素和与其覆盖的二值图像的 3\*3 子块进行与运算，如果结构都为 0，则该像素点的值就为 0，否则置为 1。而闭运算的过程和开运算的相反，即先进行膨胀操作，再进行腐蚀操作。通过开运算，可以移除掉由噪声引起的斑点，通过闭运算，可以连接起分开的小块。

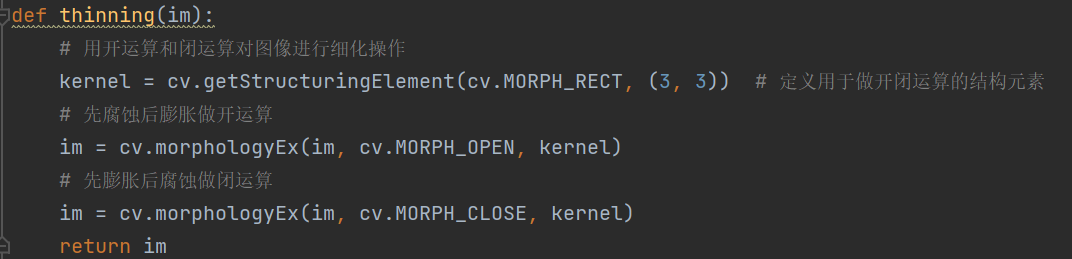


图6指纹图像预处理代码3

2、人脸图像预处理

（1）图像灰度的规格化

为了避免在人脸图像分类中的干扰项，使用黑白图片是个好主意。为了得到灰度版本，我们只需要在图像加载函数中指定，将适当的值作为第二个参数传递。

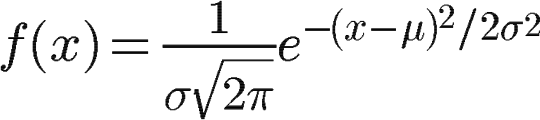
使用加权平均法：将红、绿、蓝三个颜色通道按照不同的权值进行加权平均，得到一个灰度值。这种方法的公式为：Y = 0.3R + 0.59G + 0.11B1。这种方法的权值选择是基于人眼对不同颜色的敏感度。

（2）图像的高斯模糊

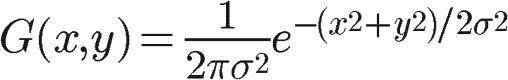
在本项目中，对输入图像进行高斯模糊处理是非常重要的一步。高斯模糊可以帮助我们减少图像中的噪声和细节，使得图像变得更加平滑，这对于后续的特征提取和人脸识别是非常有帮助的

高斯模糊本质上是低通滤波器，输出图像的每个像素点是原图像上对应像素点与周围像素点的加权和，就是用高斯分布权值矩阵与原始图像矩阵做卷积运算而已。

正态分布的密度函数叫做"高斯函数"（Gaussian function）。它的一维形式是：



其中，μ是x的均值，σ是x的方差。因为计算平均值的时候，中心点就是原点，所以μ等于0。根据一维高斯函数，可以推导得到二维高斯函数：



有了这个函数 ，就可以计算每个点的权重了。

使用cv2做高斯模糊，只要一行代码调用GaussianBlur函数，给出高斯矩阵的尺寸和标准差就可以：dst = cv2.GaussianBlur(gray, (3, 3), 0)

这里(3, 3)表示高斯矩阵的长与宽都是3，标准差取0时OpenCV会根据高斯矩阵的尺寸自己计算。概括地讲，高斯矩阵的尺寸越大，标准差越大，处理过的图像模糊程度越大。

（3）图像的人脸特征提取

这是识别过程中的关键步骤，因为图像质量会影响特征提取的精度1。常用的特征提取方法包括Haar特征、LBP特征、HOG特征等。

本项目使用的是Haar特征和LBP特征同时使用，Haar特征来确定是否是人脸，而LBP特征是用来确定是谁的人脸，从而完成人脸的识别和显示

LBP（Local Binary Pattern，局部二值模式）是一种用来描述图像局部特征的算子，具有灰度不变性和旋转不变性等显著优点。

原始LBP特征描述及计算方法：原始的LBP算子定义在像素33的邻域内，以邻域中心像素为阈值，相邻的8个像素的灰度值与邻域中心的像素值进行比较，若周围像素大于中心像素值，则该像素点的位置被标记为1，否则为0。这样，33邻域内的8个点经过比较可产生8位二进制数，将这8位二进制数依次排列形成一个二进制数字，这个二进制数字就是中心像素的LBP值1。

改进一：圆形LBP特征：为了适应不同尺度的纹理特征，并达到灰度和旋转不变性的要求，研究人员对LBP算子进行了改进，将3×3邻域扩展到任意邻域，并用圆形邻域代替了正方形邻域1。

改进二：旋转不变的LBP：通过对得到的LBP特征进行旋转，得到一系列的LBP特征值，最终将特征值最小的一个特征模式作为中心像素点的LBP特征1。

改进三：等价模式和混合模式：当某个局部二进制模式所对应的循环二进制数从0到1或从1到0最多有两次跳变时，该局部二进制模式所对应的二进制就称为一个等价模式。除了等价模式之外的称为混合模式1。

特征图可视化：可以将一幅图片划分为若干的子区域，对每个子区域内的每个像素点都提取LBP特征，然后，在每个子区域内建立LBP特征的统计直方图3。

3.1.3程序模块与调试说明

1.指纹识别

* contrast模块 增强对比度
* filtering模块 3\*3 均值滤波
* normalize模块 归一化 实现对原始数据的等比例缩放
* direction\_bw模块 二值化 沿脊线方向增强指纹纹路，采用的方法为基于脊线方向场的增强方法。
* clear模块 去除指纹中的空洞和毛刺 如果当前位置点值为0（背景）该点的四邻域点（上下左右）的和大于3则为毛刺，空洞的判断方法为该点为白色（背景）的四周为黑色（前景）八领域点两的和为0，则为空洞。
* thinning模块 细化 用开运算和闭运算对图像进行初步细化操作，先腐蚀后膨胀做开运算，先膨胀后腐蚀做闭运算。
* thin模块 进一步细化，进行骨架化处理。骨架化是一种细化的方法，可以将图像中的线条保留下来，而去除其他部分。
* point模块 找出细化后图像的所有端点，将一个点的八个邻域依次两两相减并取绝对值，并将所有结果相加，从细化图像的特征来说，和为2时为端点 和为6时为交叉点
* guanghua模块 纹线光滑处理.原理：找到每个端点，使其沿着纹线的方向移动num个像素，如果在num个像素之内遇到交叉点，则认为此端点为毛刺，去除此点
* Walk模块 判断离端点num距离内是否有另一端点
* removeTooClose模块 该函数的主要目的是提高特征点的独立性，移除距离过于接近的点，以确保特征点在相邻区域不过于密集。
* Feature模块 特征点提取

使用point模块 将所有端点分为端点和交叉点

使用guanghua模块 去除毛刺

使用removeTooClose模块 移除距离过于接近的点

* getBasicPoint模块 通过迭代txy1中的点，使用findPoint函数找到附近的两个点，并计算这三个点的角度。

在txy2中找到与当前点具有相同属性的点，同样使用findPoint函数找到附近的两个点，并计算这三个点的角度。

比较这两组角度，如果差异在误差阈值(b)以内，则返回第一个找到的匹配点对。

* getThetas模块 给定三个点，计算这三个点构成的三角形的三个角度，并返回排序后的角度列表。
* findpoint模块 寻找离指定点最近的num个点，并返回这些点的坐标集合。
* pointMatch模块 判断两组特征点集是否匹配，通过对两组点集进行比较，判断匹配率是否达到阈值，从而判断匹配成功或失败。

使用 getBasicPoint 获取两组特征点的基本匹配点

根据基本匹配点，使用 findPoint 函数找到附近的若干个点。

计算匹配率，并根据阈值判断匹配是否成功

* matchAll模块 将目标图片与指纹库中所有图片进行匹配

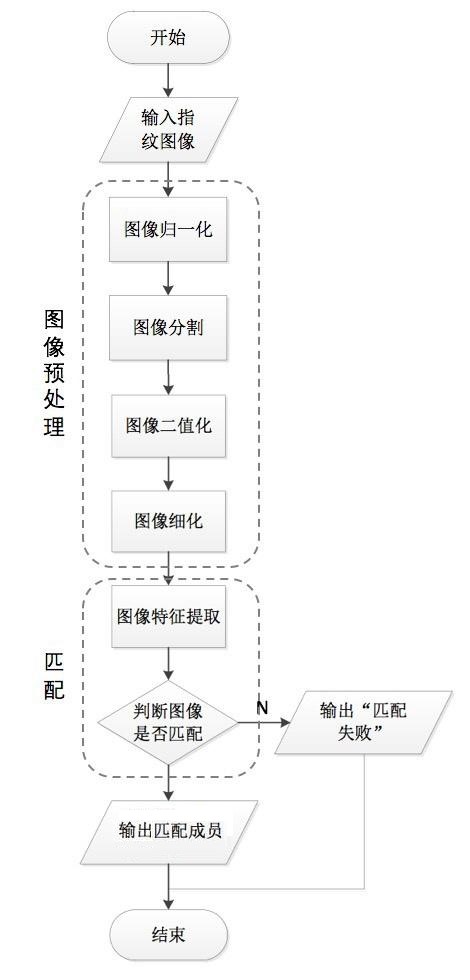


图7指纹识别流程图2

2.人脸识别

* gray 是通过 cv2.cvtColor函数把原图像转换为灰度图像数据
* dst 是通过cv2.GaussianBlur函数把灰度图像数据进行高斯模糊后的数据
* face\_detect 是通过cv2.CascadeClassifier函数将人脸识别器加载的数据
* face 是通过face\_detect.detectMultiScale函数检测进行灰度和高斯处理后的图像进行人脸识别后人脸的数据
* cv2.rectangle是在图像上根据人脸位置进行画图后的图像数据
* idnum, confidence是人脸通过recognizer.predict来进行人脸判断的数据
* 检测的人脸相关参数有：眼睛、嘴巴、鼻子
* 通过读取这些人脸相关的参数和已有的数据库进行比对完成人脸的识别

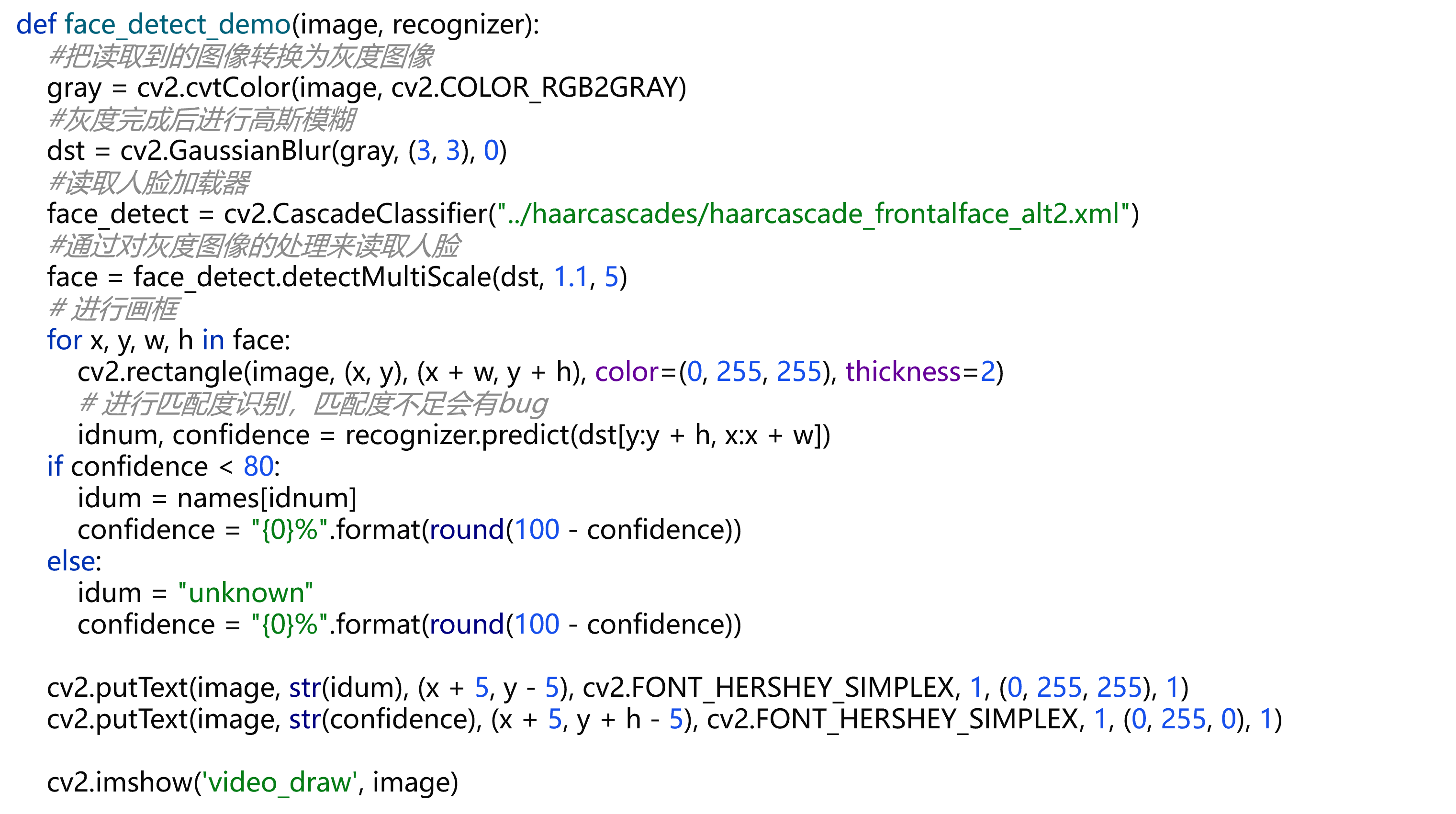


图8人脸识别处理代码

3.1.4存在的问题和解决措施

1. 当人脸识别未识别到人脸时，程序将报错并退出

当遇到这个问题的时候，我没有任何的头绪，因为当它检测到人脸的时候工作是完全正常的，只有当检测不到人脸的时候才会报错，为此，我专门研究了报错的代码，

TypeError: '<' not supported between instances of 'str' and 'int'

这段报错的意思是：str类型和int类型之间不支持<运算。也就是说，你不能直接比较一个字符串和一个整数的大小，因为它们是不同的数据类型。

后来经过检测后发现，当程序没有检测到人脸相关的数据时，他将会输出一段str类型的字符串，当它检测到人脸的时候，他将输出与人脸匹配的id号，因此，在这段程序之间我加上了一段检测输出数据是str或者int类型的代码，当数据类型为int时，它将会输出与人脸匹配的id号，当输出为str时则不进行输出，从而解决了这个错误

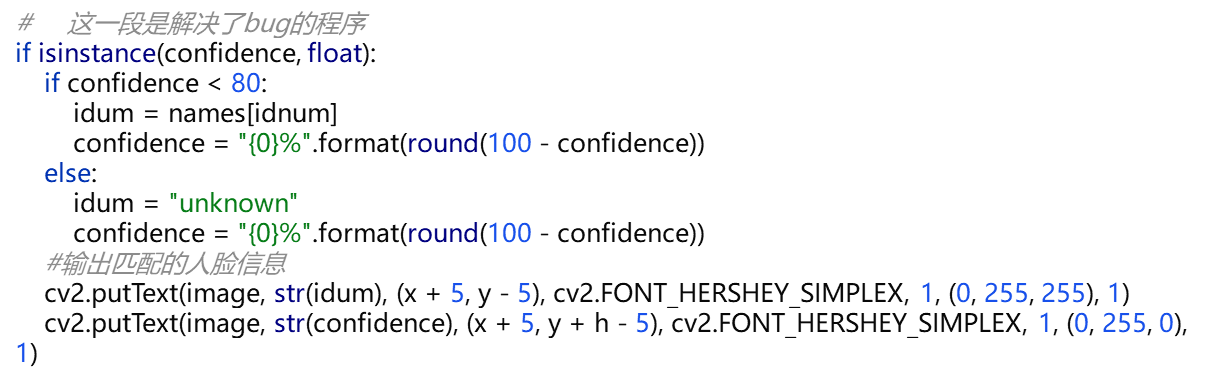


图9人脸识别代码

3.2人脸检测

（1）特征提取

Haar特征和LBP（局部二值模式）特征都是用于图像处理和计算机视觉中的特征描述子，它们都可以用于描述图像的局部纹理特征

Haar特征提取过程：

1、Haar特征：Haar特征分为三类：边缘特征、线性特征、中心特征和对角线特征。特征模板内有白色和黑色两种矩形，并定义该模板的特征值为白色矩形像素和减去黑色矩形像素和。Haar特征值反映了图像的灰度变化情况。

2、积分图：积分图是一种能够描述全局信息的矩阵表示方法，其构造方式是位置 (i, j)处的值ii (i, j)是原图 (i, j)左上角方向所有像素灰度值f (k, l)的和。积分图能够在多种尺度下，使用相同的时间（常数时间）来计算不同的特征，因此大大提高了检测速度。

3、特征值的计算：利用积分图计算Haar特征值，例如计算边缘矩形特征：区域A的像素值之和为：Sum (A) = ii (5)+ii (1)−ii (2)−ii (4)，区域B的像素值之和为：Sum (B) = ii (6)+ii (2)−ii (5)−ii (3)，该矩形特征值为：Sum (A)-Sum (B)。

4、特征值归一化：由于Haar特征值的范围可能非常大，因此需要进行归一化，压缩特征值范围。

LBP特征提取过程：

1、原始LBP特征描述及计算方法：原始的LBP算子定义在像素33的邻域内，以邻域中心像素为阈值，相邻的8个像素的灰度值与邻域中心的像素值进行比较，若周围像素大于中心像素值，则该像素点的位置被标记为1，否则为0。这样，33邻域内的8个点经过比较可产生8位二进制数，将这8位二进制数依次排列形成一个二进制数字，这个二进制数字就是中心像素的LBP值。

2、改进一：圆形LBP特征：为了适应不同尺度的纹理特征，并达到灰度和旋转不变性的要求，研究人员对LBP算子进行了改进，将3×3邻域扩展到任意邻域，并用圆形邻域代替了正方形邻域。

3、改进二：旋转不变的LBP：通过对得到的LBP特征进行旋转，得到一系列的LBP特征值，最终将特征值最小的一个特征模式作为中心像素点的LBP特征。

通过这两种特征提取方法，我们可以提取出人脸的边缘特征、线性特征、中心特征和对角线特征，而且改进后的LBP算法可以对旋转后的人脸图片进行识别。

3.3人脸和指纹特征提取

3.3.1指纹特征提取

（1）特征点的选取

根据 Crossing Number（CN）理论，取某个实心像素 P（黑色像素）为中心，建立 3\*3 的像素区块，绕边缘环绕一周分别为 P1,P2,…,P8根据公式：



算出 CN 值，其中实心像素的值为 1，空心像素的值为 0。根据 CN 理论，当 CN 值等于 1 时，该像素 P 为端点，当 CN 值等于 3 时，该像素 P 为分叉点。按照这个理论，以指纹的中心点为中心，周围一定区域内计算出所有的特征点。

（2）特征点的筛选

因为图像本身以及处理过程不是百分之百精确的，所以，我们找出的特征点中必定有许多“伪特征点”，因此，我们要筛选这些特征点。

· 根据所得特征点去掉细化图像的毛刺：

所得特征点中，有许多端点，这些端点有可能是细化图中毛刺的端点。对每一个端点，顺着纹路向前走若干个像素（比如 5 个），如果遇到了分叉点，那么就说明这个端点就是毛刺的端点，将其去除，同时也在顺着纹路走的同时，去除走过的黑色像素点，从而对图像起到光滑的作用。

· 去除相距过近的特征点

相距太近的特征点也可能是图像中某个短线的所造成的为特征点，应该予以去除。

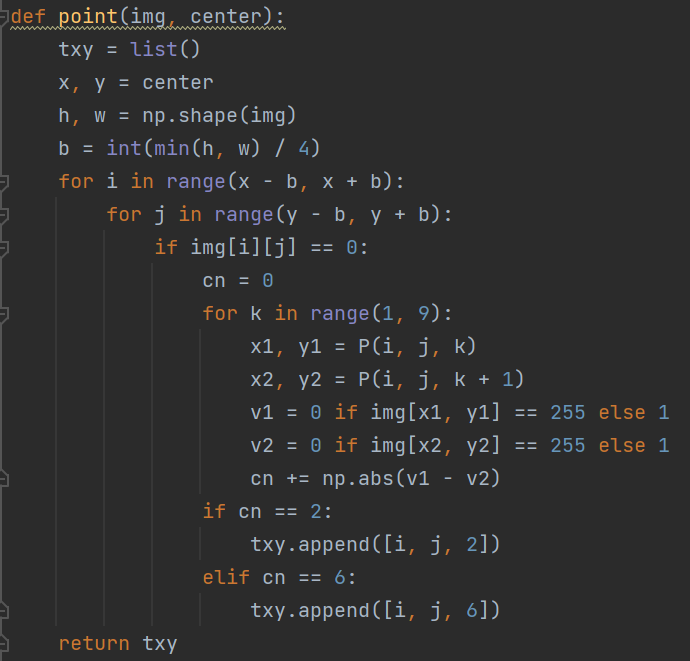


图10 指纹识别代码1

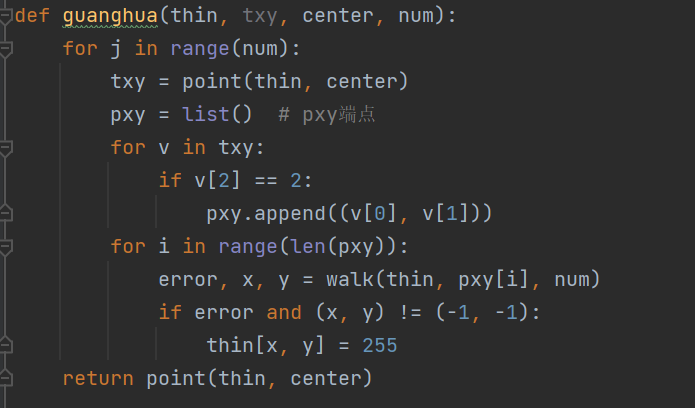


图11指纹识别代码2

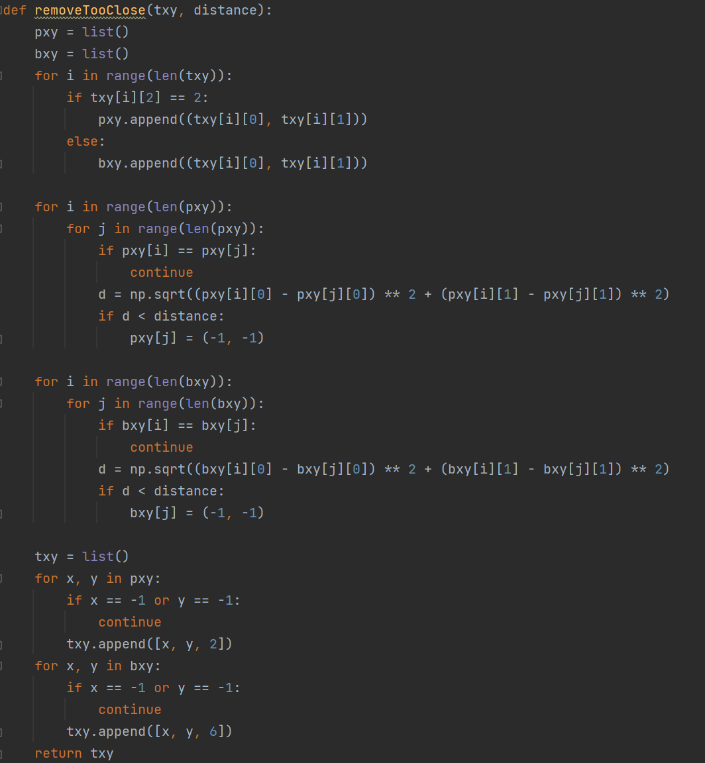


图12指纹识别代码3

3.3.2人脸特征提取

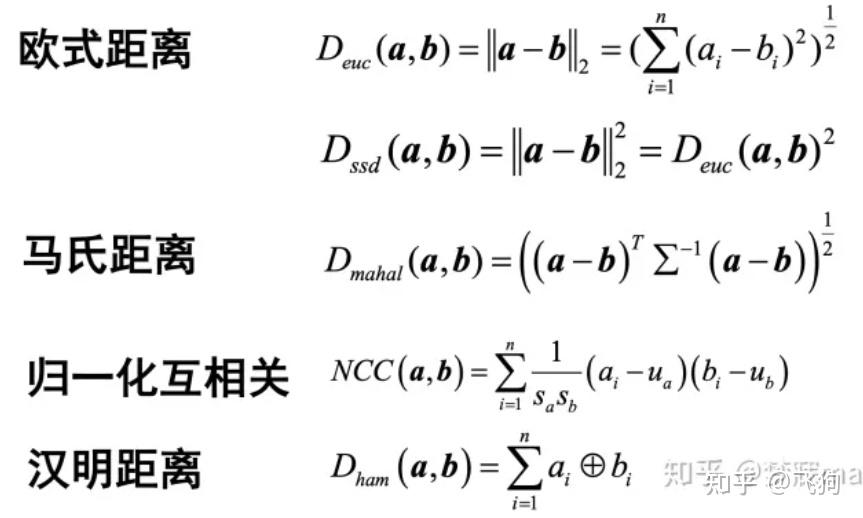
人脸特征点匹配是人脸识别中的一个重要步骤，主要包括以下几个步骤：

1、人脸检测：从图片中准确定位到人脸。

2、人脸对齐：自动定位出面部关键特征点，如眼角、鼻尖、嘴角和脸部轮廓等1。这些点的位置一是用来做人脸的几何校正，即把人脸通过缩放、旋转、拉伸等图像变化变到一个比较标准的大小位置。

3、特征提取：从人脸区域提取各种特征，包括LBP、HOG、Gabor等。

4、特征匹配：对两张人脸图像的特征向量进行对比，计算相似度。特征匹配的方法主要有最近邻匹配和归一化互相关匹配。最近邻匹配是通过计算特征向量之间的距离来找到最相似的特征，而归一化互相关匹配则是通过计算特征向量之间的相关性来找到最相似的特征。



3.4人脸和指纹混合匹配识别

3.4.1指纹特征点匹配

参考了资料《基于全等三角形的指纹匹配算法的研究与实现.pdf》。资料中提到，可以利用全等三角形法寻找两图的基本点，从基本点出发，用点匹配法对两图像的特征点进行匹配。

·具体方法：

取得两个图像的特征点序列，若其中一个序列中有特征点 P 与其最临近的两个与 P 类型不同的特征点构成一个三角形，然后，在另一序列中也能找到一个特征点 Q，也与其最邻近的这种特征特征点构成一个相似的三角形，那么这两个特征点对应，可以作为两个图的基本点。

而点匹配法稍作了简化：以这两个基本点 P、Q 出发，在其周围找出若最邻近的若干个点，计算其中的端点个数，若端点个数两边相差小于一定的阈值，那么就判定这两个指纹图像匹配。

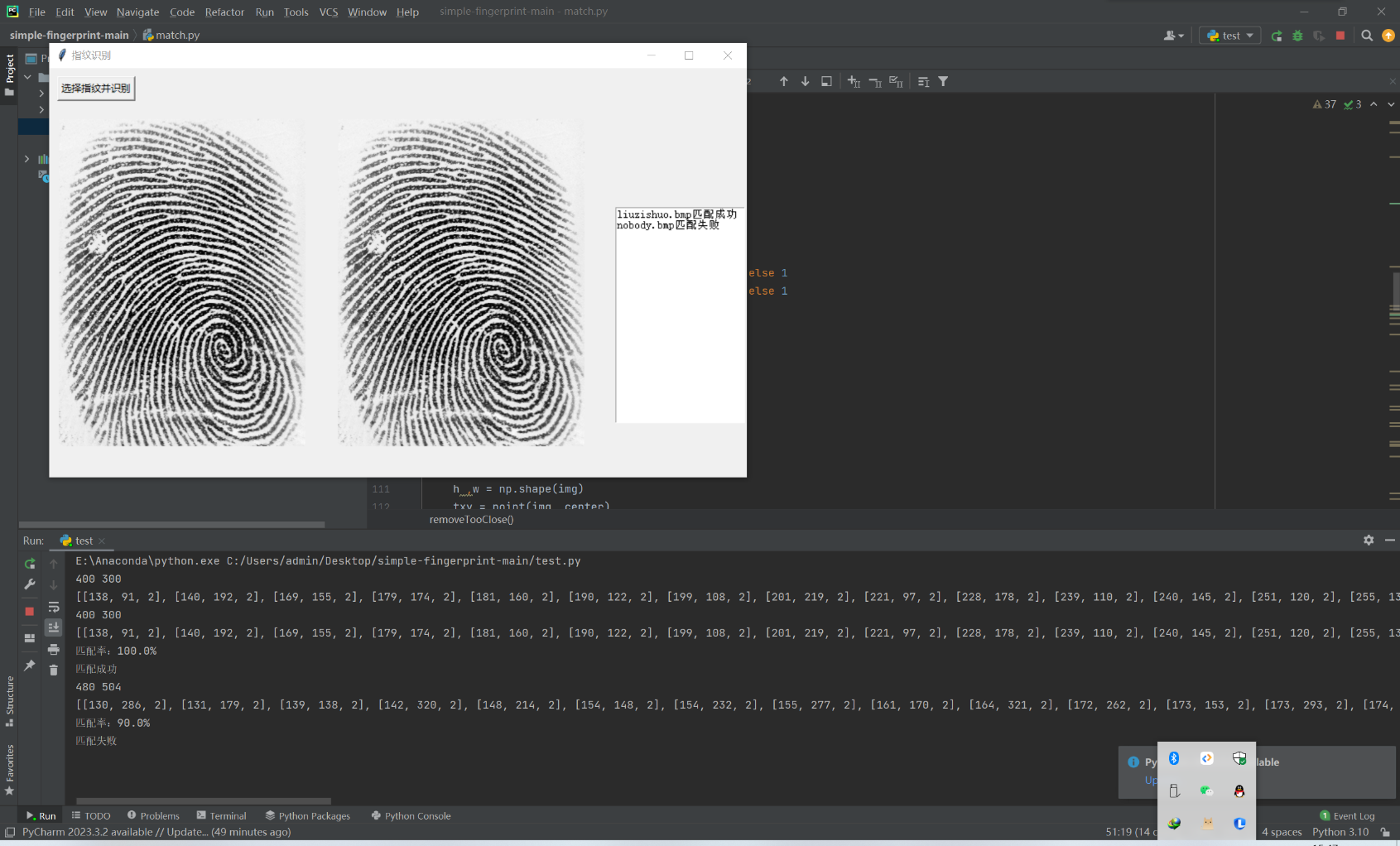


图12指纹识别匹配1

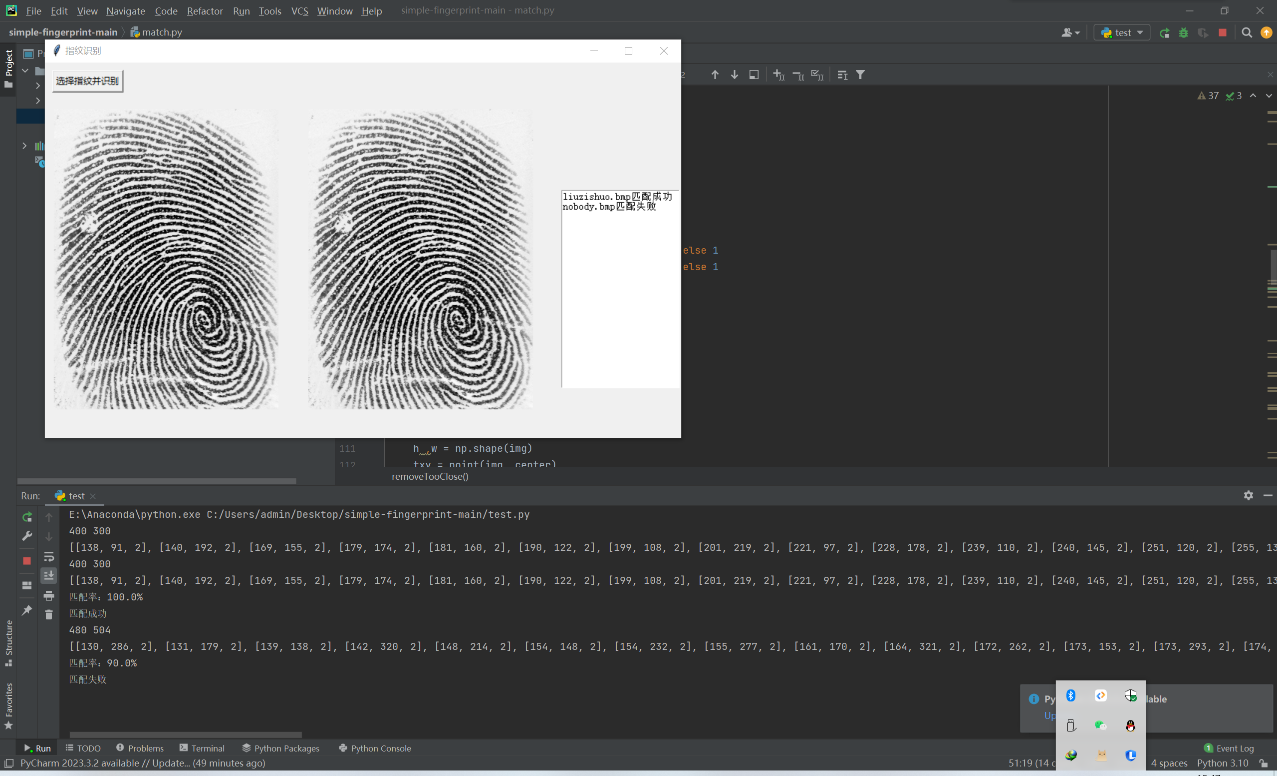


图13指纹识别匹配2

3.4.2人脸特征点匹配

通过LBP（局部二值模式）算法匹配人脸特征点主要包括以下步骤：

1、图像分割：将图像的被检测区域分割成一个个格子，例如16×16大小。

2、计算LBP特征：比较一个像素值与其周边8个相邻格子的大小。在一个3×3的区域中，最中间的像素值相比于其他像素的大小。如果某个相邻格子的值比中间值小，那么它会被记为0；相反地就会被标注为1。这样一来，3×3大小的框一共可以产生8个二进制值。

3、生成LBP值：沿着正方向或者反方向来组装这8个二进制数，那么将得到一个新的数值。这个新的数值就是中心像素的LBP值。

4、重复计算：重复以上步骤，直到处理完所有像素点，得到完整的LBP结果。

5、特征匹配：对两张人脸图像的LBP特征进行对比，计算相似度。特征匹配的方法主要有最近邻匹配和归一化互相关匹配。最近邻匹配是通过计算特征向量之间的距离来找到最相似的特征，而归一化互相关匹配则是通过计算特征向量之间的相关性来找到最相似的特征。

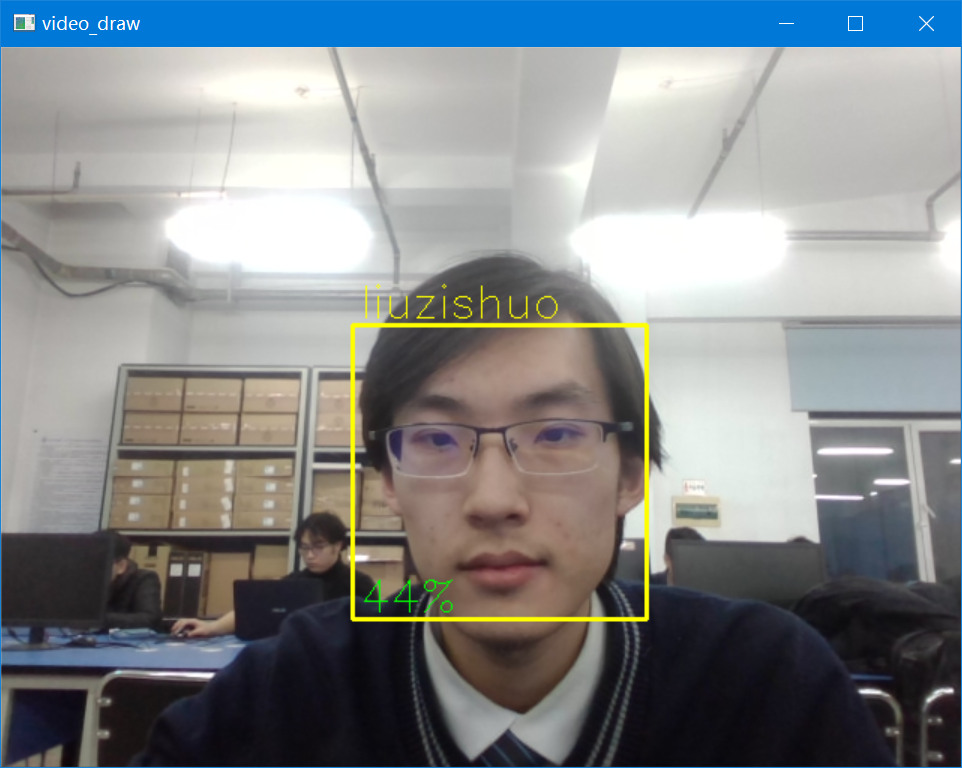


图14人脸识别效果

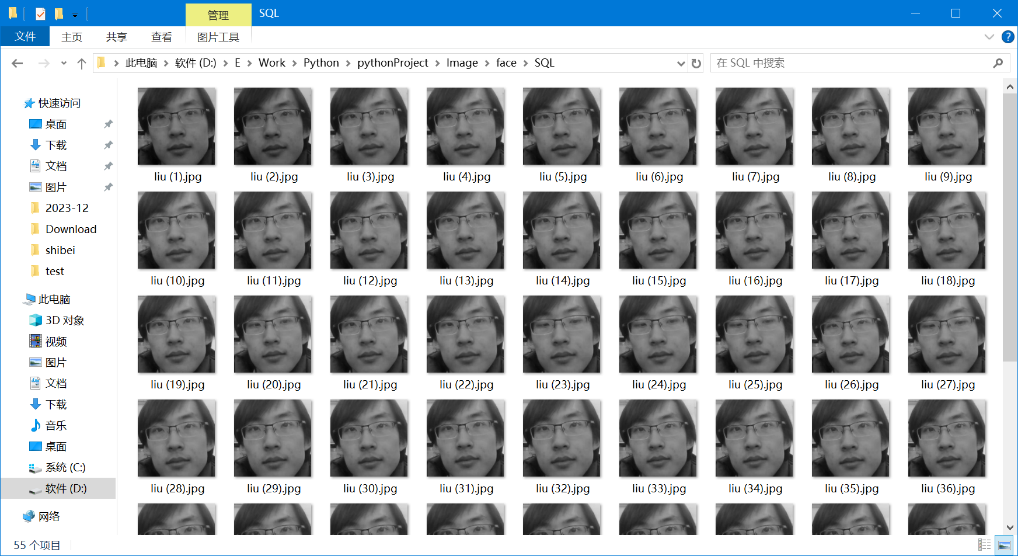


图15训练数据集

3.5 GUI界面设计

首先确定项目需求，项目是人脸指纹混合识别系统，需求如图----，了解需求，通过pyside6与qt designer进行项目GUI的编写，在designer中分别绘制出主材单与各级子菜单并通过pyside6 Uic转化为py文件，建立项目主文件并在主文件中调用各级菜单的py文件，并在主文件中写出各层之间的逻辑关系，显示各级菜单，并把各个触发函数绑定在相应的按键下方，完成整个项目GUI的编写。

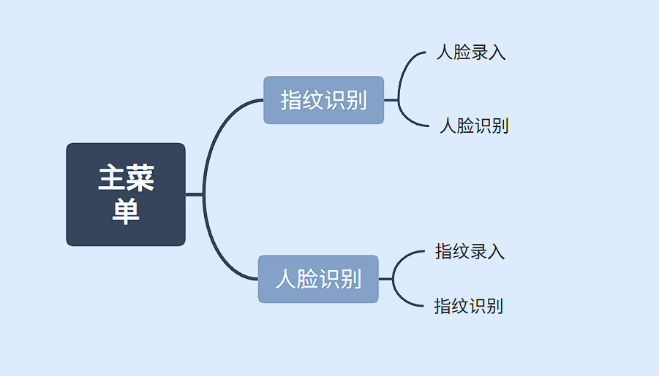
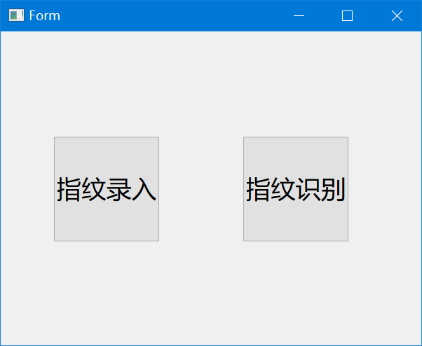
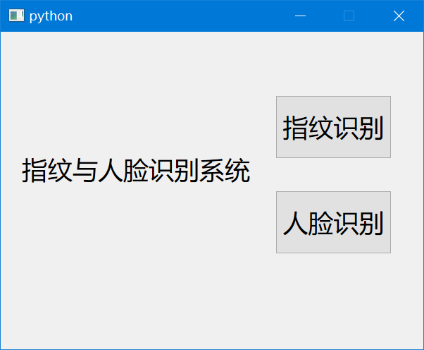


图16 GUI流程图



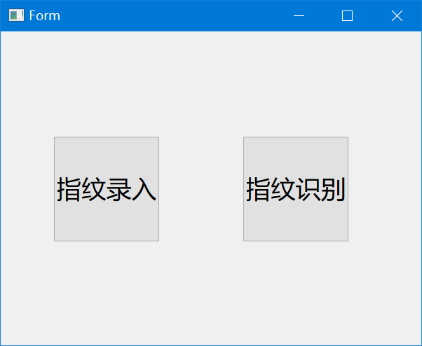
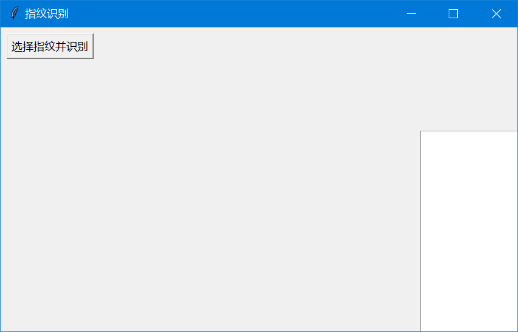
 

图17 GUI运行界面



图18 GUI运行代码

四、实践总结

1．实践过程中遇到的问题及解决措施、学到了那些知识。

2．组长给出各成员的实践建议得分（百分制）。

李嘉昱：指纹识别程序通过对指纹规格化处理，分割，然后进行基于方向场的二值化处理，之后对图像细化，特征点采集等预处理操作，得到一串基于特征点的指纹模型，之后根据所写的匹配算法进行匹配，所得到的结果是能大概区分出相同的指纹与不同的指纹，但精度上还有待加强。本程序所含的代码中图片的读取与展示，细化过程的轮廓提取，对卷积核的卷积操作，中值与均值滤波等调用了外部库（opencv、numpy、skimage、scipy 等）

刘子硕：人脸识别程序通过将读取到的视频数据进行图片化，然后进行灰度处理和高斯模糊去除图片中的噪点，之后通过Haar算法得到人脸的数据，然后通过HBP算法得到人脸的特征点数据，从而确定人脸的相关位置和参数，Haar特征级联分类器是一种基于机器学习的方法，可以根据一组预先训练好的特征来快速检测出图像中的目标对象，HBP算法是一种基于直方图的方法，可以根据图像的颜色分布来计算出图像的相似度，从而实现人脸识别

陈冠宇：通过实验，我们的人间指纹识别系统表现出了优异的准确性、响应速度和安全性，这些结果为系统的实际应用提供了强有力的支持。首先，选择适合的算法对于系统的性能至关重要。在人脸识别方面，我们采用了LBP（Local Binary Patterns）算法作为特征提取的方法。LBP算法具有简单高效的特点，并且对光照和表情变化有较好的鲁棒性。而在指纹识别方面，CN理论基于指纹纹线和纹谷之间的空间夹角，能够有效地提取指纹特征，具有良好的识别性能。

其次，数据的质量对于识别系统的性能具有重要影响。在实践过程中，我们注重了数据采集的质量控制，进行了图像预处理确保图像清晰，并消除了噪声和失真。同时，为了增加系统的鲁棒性，我们还通过多样性的数据采集，尽可能的覆盖不同的人群和场景，以提高系统的泛化能力。实验得出的经验教训也将有助于我们进一步完善系统的设计和技术，从而更好地满足用户需求并应对实际应用中的挑战。

余凯：通过参与这个实验，我对人脸识别和指纹识别这两种生物识别技术有了更详细的了解，并对它们的不同特点和应用场景有了更清晰的认识。人脸识别相对而言，更加直观和易于接受，因为人脸是每个人身体特征中最易于获取和识别的。然而，其对于环境因素的敏感性以及在一些特殊情况下的准确率下降，仍然需要进一步的优化和改进。指纹识别在个体辨识和安全性方面具有明显优势，因为指纹的唯一性和稳定性。然而，指纹的采集过程可能引起隐私问题，并且不适用于需要非接触的场景。

刘晴：人脸识别和指纹识别各自具有不同的应用场景和优势。例如，人脸识别适用于无接触场景，如门禁考勤、手机解锁等场景；而指纹识别则适用于需要高安全性和个体辨识的场景，如银行卡支付、国家安全等场景。通过本次实验，我深入了解了人脸识别和指纹识别这两种生物特征识别技术的优势与劣势，并加深了对于其在应用中的适用性的理解。随着技术的进步，对于人脸识别和指纹识别技术的继续研究和优化，将对于增强其准确性、鲁棒性和隐私保护等方面，都具有重要的意义。在实际应用中，需要根据实际需求和场景进行选择，以确保其效果和安全性。

五、主要参考文献

参考博客：<https://blog.csdn.net/MrCharles/article/details/79300671>

参考博客：<https://blog.csdn.net/weixin_42121843/article/details/100124486>

参考博客：<https://blog.csdn.net/MrCharles/article/details/79300671>

参考博客：https://blog.csdn.net/xidianzhimeng/article/details/19634573

参考博客：https://zhuanlan.zhihu.com/p/651523263

[1] 袁东锋,杜恒,秦小铁.基于三角形局部特征点模型指纹匹配算法[J]. 重庆师范大学学报.2013,3,16

附件：

