基于生物特征以及弱口令

密码生成器

当前形式分析

- 生物特征识别
- 互联网发展动态
- 密码学
- 同类产品分析

生物特征识别

01

背景

生物特征是唯一的，每个人都会有仅属于他的独特性。

在当今信息化时代，如何准确鉴定一个人的身份、保护信息安全，已成为一个必须解决的关键社会问题。传统的身份认证由于极易伪造和丢失，越来越难以满足社会的需求。

02

定义

通过计算机技术对生物特征进行统计分析，便能确定如身份、位置等信息。

通过计算机与光学、声学、生物传感器和生物统计学原理等高科技手段密切结合，利用人体固有的生理特性和行为特征来进行个人身份的鉴定。

03

应用领域

声纹识别除了可以用于识别说话人，还可以进行语意的判断。

计算机视觉、图象处理与模式识别、计算机听觉、语音处理、多传感器技术、虚拟现实、计算机图形学、可视化技术、计算机辅助设计、智能机器人感知系统等其他相关的研究。

04

技术优势

生物特征识别技术的出现，让我们不再需要实体验证工具。

每个人的生物特征具有与其他人不同的唯一性和在一定时期内不变的稳定性，不易伪造和假冒，所以利用生物识别技术进行身份认定，安全、可靠、准确。

互联网发展动态

网站的爆发性增长

我们都无可避免的注册大量的网站，此时账号和密码的管理就成了一个很严重的问题。

01

02

移动端的普及

移动端的方便使得移动互联网的快速普及，但是随之而来的是各种各样的应用程序；而附带着的则是一个密钥管理问题。

互联网与现实联系

随着科技高速发展，我们的网络生活已经逐渐与现实生活逐渐交融，成为一体。

03

04

网络安全

网络安全已经不单纯是虚拟空间的安全，它已经延伸到个人安全、社会安全、经济安全，乃至国家安全。

同类产品分析

方便、安全的密码生成器

本项目实现的密码生成器通过生物特征，网站名称，便于记忆的弱口令生成一个复杂度较高、安全性较强、毫无规律的强口令，当你需要再次使用密码时，只需要输入网址和弱口令即可等到强口令。

项目实现过程

- 项目分析与设计
- 项目总体实现过程
- 算法设计及实现过程
- 后台框架设计
- 后台实现过程

项目分析与设计

需求出现

我们需要大量方便使用，同时具备较强安全性的密钥。

继续深究

也许会有人提出可以使用相同的密钥，或者是一些具有逻辑记忆的密钥，但是这些密码毫无疑问是不安全的。

需求分析

即需要安全性也需要方便使用，若要使项目实现，我们需要找到一种能加入到强口令中的特征，并且这种特征方便携带。

想法初现

在这个互联网高速发展的时代，每个人都无可避免的注册大量的网站和手机APP，这些账号和密码的管理就成了一个严重的问题。

需求实现

生物特征刚好是一种随身携带且唯一的特征，我们将生物特征及弱口令融合其中，生成强口令。

项目总体实现过程

需求分析

完成需求分析
程序总体结构设计

算法设计及实现

面部识别算法设计以及实现
单向加密算法设计以及实现

前/后端搭建

前/后端搭建
前后端连通使用
算法置于后端运行
及返回数据予前端

项目测试

算法功能及运行效果测试
应用测试

算法设计及实现过程

面部识别

将图片数据中的面部识别出来，并进行简单标识。

第 一 项 工 作

面部特征

将第一项工作中标记的面部数据，提取面部特征。

第 二 项 工 作

单向加密

将面部特征以及弱口令加入到加密算法中，计算出口令。

第 三 项 工 作

前端设计

SplashActivity

作为APP首次冷启动的启动页，提高用户体验。

workActivity

用户输入网址口令界面，同时接收返回的密码提供给用户

loginActivity

用于用户登录注册以及忘记密码找回的操作。

加密传输

APP与服务端的通信数据均采用AES进行加密，保证信息的安全。

CameraActivity

调用安卓设备的前摄像头进行拍照提取人脸的信息。

邮箱找回密码

用户忘记密码可通过邮箱进行验证并修改密码。

后台框架设计

Controller层

后台的主要功能实现

Util层

项目中所需方法类

Repository层

继承JPARepository接口，实现数据库操作

Webadd实体类

描述网址的类

User实体类

描述用户的类

综合技术分析

- 面部识别算法的主要技术
- 加密算法主要技术
- 后台实现过程

面部识别算法的主要技术

本项目中面部识别算法主要使用的是一个Dlib的框架，该框架中包含了大量的机器学习算法；所有这些都旨在通过干净和现代的API 实现高度模块化，快速执行和简单易用。

面部检测

```
img = cv2.imread(img_path, cv2.IMREAD_COLOR)          /*图片读入
b, g, r = cv2.split(img)                              /*三原色转换
img2 = cv2.merge([r, g, b])
dets = detector(img, 1)                               /*使用模型对面部的检测
```

面部识别

```
for index, face in enumerate(dets):/*提取出刚刚面部检测的每一张脸
shape = shape_predictor(img2, face) /*提取出面部特征点的坐标
/*将面部特征点重新计算出128维向量值
face_descriptor = face_rec_model.compute_face_descriptor(img2, shape)
```

同人判断

```
/*我们计算距离的方式为，将两张图片的128维坐标一一取出
此外，计算其欧拉距离
   for i in xrange(len(data1)):
       diff += (data1[i] - data2[i])**2
diff = numpy.sqrt(diff)
```

加密算法主要技术

弱口令的加入

/*将弱口令加入，并更新
hc = hmac.new(sys.argv[2])
hc.update(str3)

增长输出

/*使用base64作为增长输出
base64_str =base64.urlsafe_b64encode(hash_str)

面部数据的加入

/*其中data1为面部128维向量值
str3 = bytes(data1)
h = hmac.new(str3)

网址的加入

/*加入网址，作为密钥，并以之前的加密消息作为消息加密
hash_bytes = hmac.new(sys.argv[3],hc.hexdigest())
hash_str = hash_bytes.hexdigest()

前端设计

后台实现过程


需求分析
分析项目所需要实现的功能点
数据库设计
设计数据库的表以及表与表之间的关联
代码实现
根据需求分析及具体要求，实现代码
测 试
测试代码，连通前端

实现效果与测试

- 面部特征点检测
- 欧拉距离分界点确定
- 欧拉距离评判机制测试
- 输出密码重复字串测试

面部特征点检测

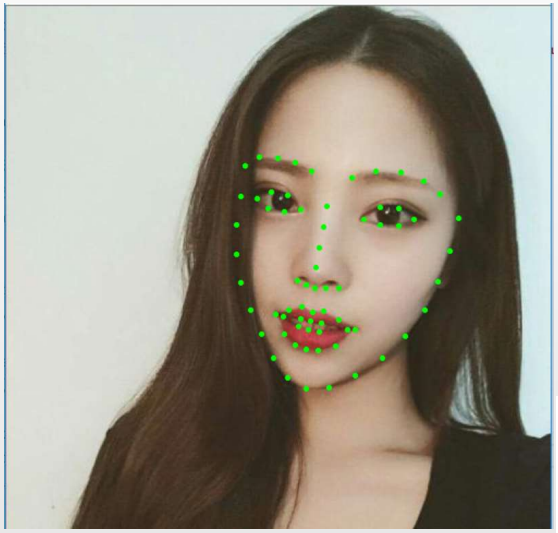
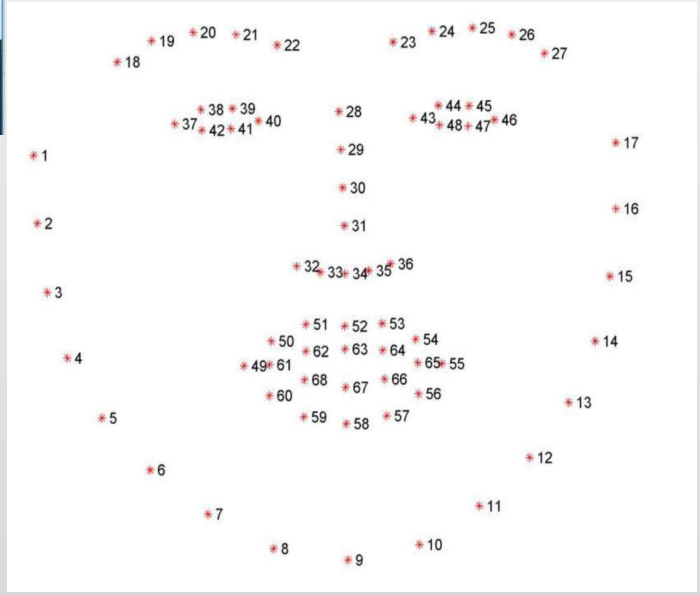


图 1

图 2

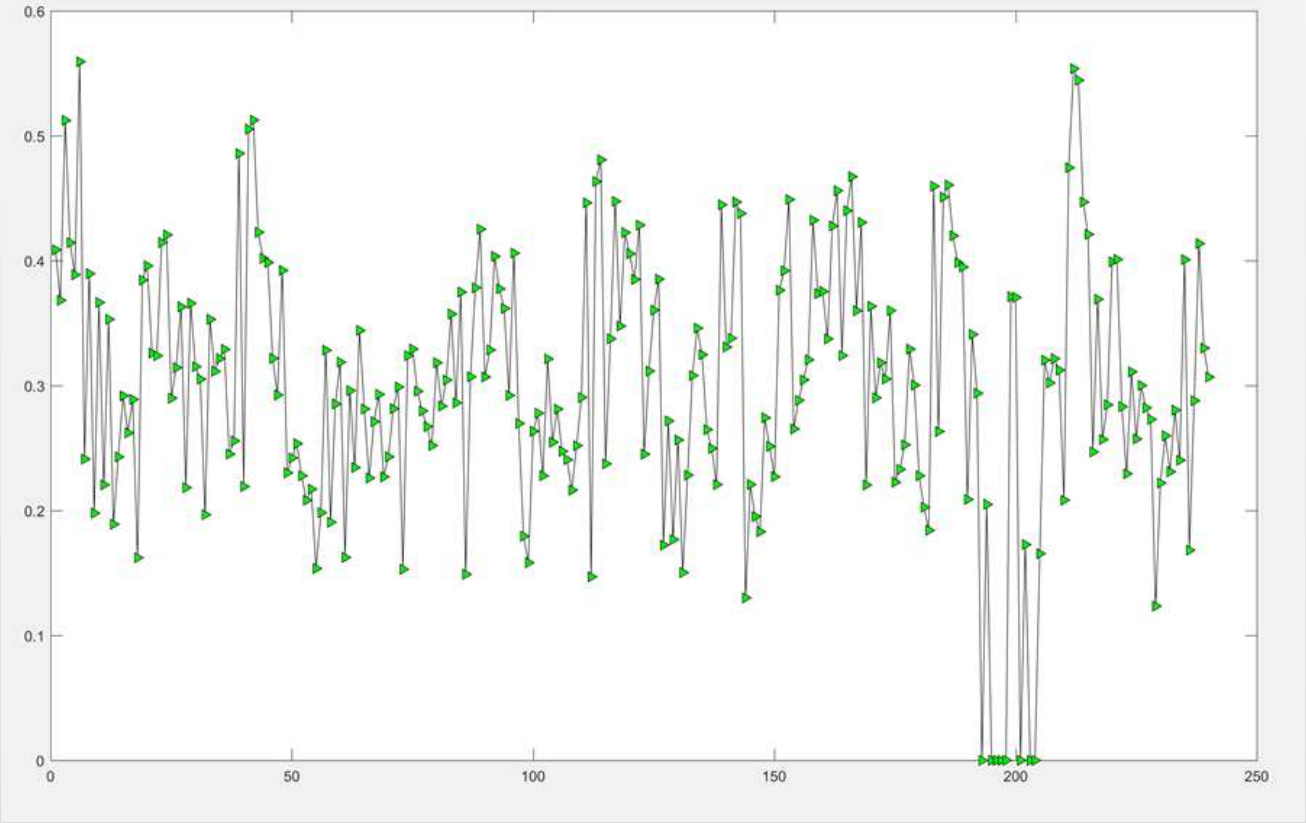


面部特征点提取

通过特征提取器，提取面部的68个特征点。

- 面部识别效果（图1）
- 面部68个特征点（图2）
- 通过比对可得，面部68个特征点提取完整。

欧拉距离分界点确定

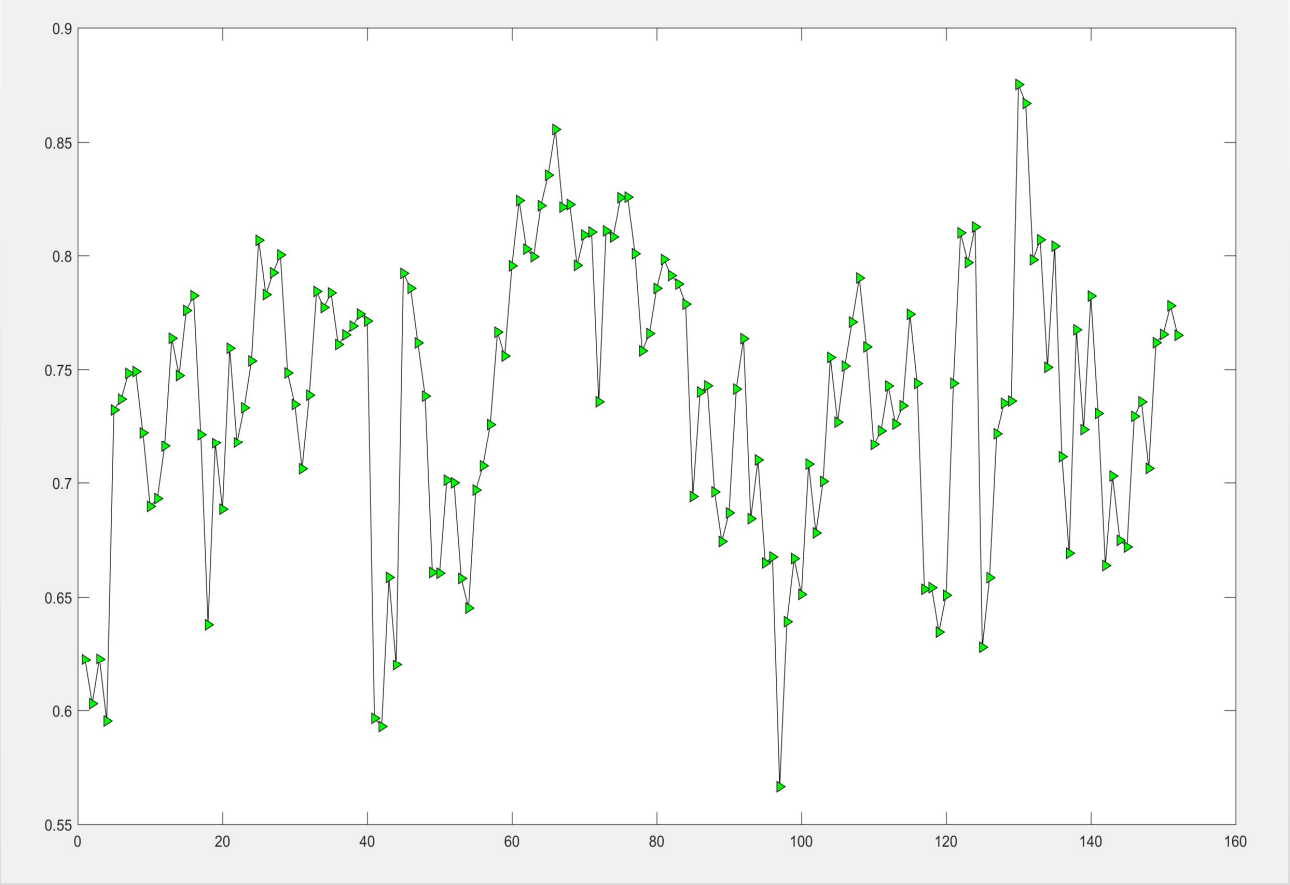


在同一个人不同照片的测试中，其特征点之间的欧拉距离都小于0.6，故我们使用的判别标准分界点为0.6。

240

- 相同人在不同角度
- 240 组测试图片
- 欧拉距离均小于0.6

欧拉距离评判机制测试



我们以之前的分界点为标准，在159组不同人的面部识别中，结果如下图所示，准确率高到97.5%(即判定结果为不是同一个人)，由此可见，我们的面部识别算法具有很高的准确率。

159

- 0.6 为分界点
- 159组测试图片
- 准确度为97.5%

输出密码重复字串测试

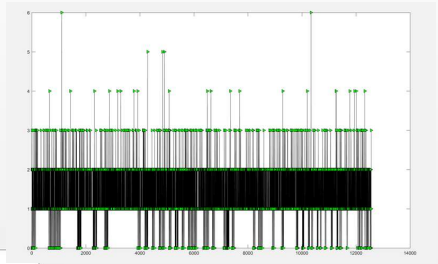


图　一

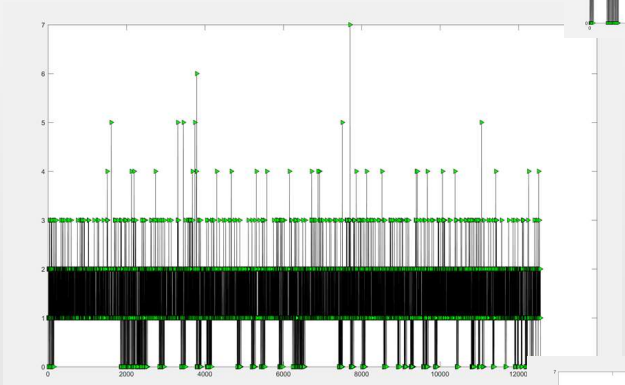


图　二

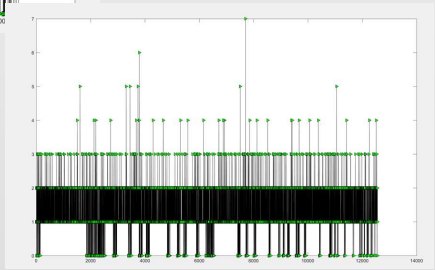


图　三

本次使用的测试用例为12561个，在经过多情况测试后可得，无论是在相同的面部数据或相同的弱口令下，均有很强的随机性，符合需求。

不相同的面部数据及相同的弱口令

最长子串6位

全长44位

如　图　一

相同的面部数据及弱口令

最长子串7位

全长44位

如　图　二

不相同的面部数据及不相同的弱口令

最长子串5位

全长44位

如　图　三

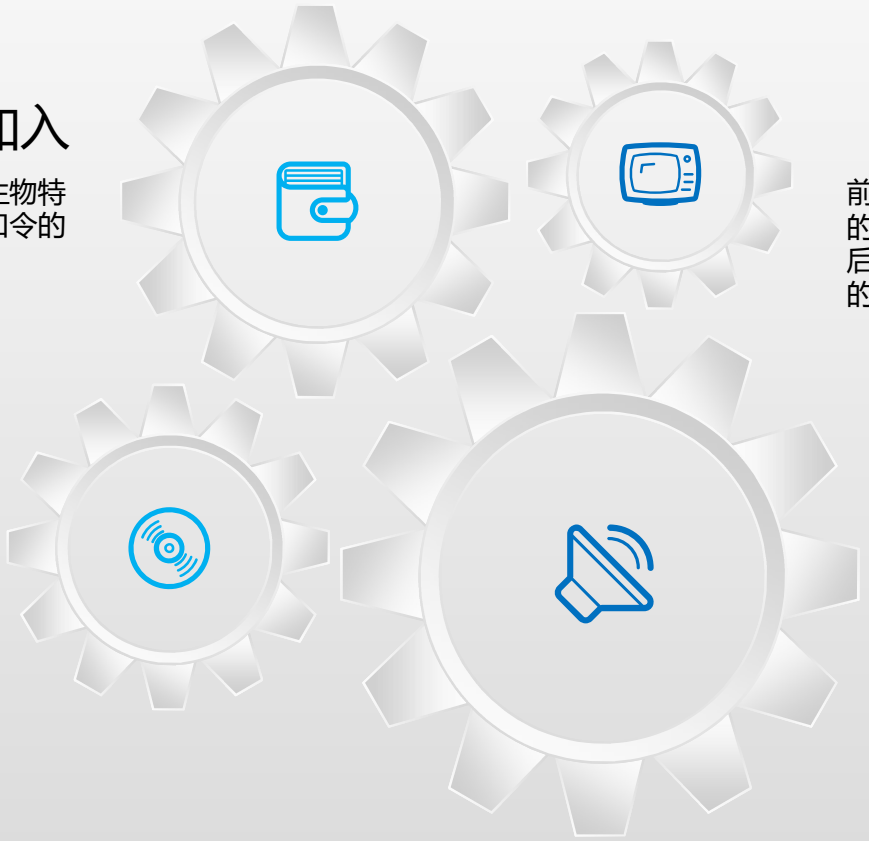
总　　结

- 实现功能简述
- 总 结
- 仍存在的一些问题

实现功能简述

生物特征数据的加入

我们通过Dlib的两个模型，将生物特征数据得到，并成功加入到强口令的计算中。



前/后端搭建

前端应用界面的编写及接收发送数据的处理已经完成。
后端接收数据并运算后重新发送数据的处理也已经完成。

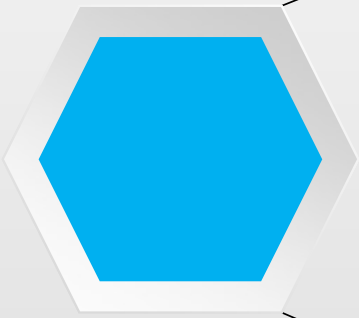
加密算法的实现

通过使用生物特征及弱口令，进行单向加密，产生出高度随机，重复性低的强口令。

由弱口令计算出强口令

由前端进行注册登陆，而后拍照得到面部数据，将数据打包发送至后端并进行处理，将得到的强口令返回前端。

仍存在的一些问题



处理时间相对较长

由于数据集的加载需要一定的时间，所以在进行单次的识别、计算密码的时间相对增长，在往后的工作中，我们将继续优化算法，减少数据加载时间，增加计算密码的效率，提升用户体验。

识别精准度有待提升

在本次使用的模型中，我们在测试中发现，识别精准度仍在一定程度上受到了外界因素的影响（如光线，像素），在往后的工作中，我们将会继续训练新的模型，减少外界因素的影响。

增加更多的生物特征

在本项目中，我们使用的生物特征仅有面部数据，比较单一。在往后的工作中，我们将继续努力加入其他的生物特征（如声音，瞳孔）等，增强密码生成的安全程度，以部署在不同安全强度的地方。

最后

学习的能力

在通过本次的项目中，我们在查询文档、博客等的过程中，提升了我们自身的学习能力，增强了解决问题的能力，积累到了更多的知识。

计划的安排

在项目进行中，项目的开发，算法的实现与学习糅合在一起。通过本次的经历，学习到了计划的重要性，为以后的学习之路有了更好的规划，也提升了自己对于计划的指定能力。

交流与协作

在项目设计和实现过程中我们均有自己的想法，在前后端的对接中也屡次出现问题，但是最终我们都通过交流沟通解决了这些问题，使得我们更加明白倾听与协作的道理。

THANKS

演讲完毕 感谢聆听