

教育部高等学校大学计算机课程教学指导委员会

中国大学生计算机设计大赛



软件开发类作品文档简要要求

作品编号： 61757

作品名称： 基于人脸识别的智慧医疗预约挂号平台

作 者： 文杰、周文志、杨昊凡

版本编号： V1.0

填写日期： 2019/04/30

目录

1.需求分析.....	3
1.1 项目背景.....	3
1.2 用户分析.....	3
1.3 主要功能.....	3
1.4 竞品分析.....	3
2.概要设计.....	4
2.1 模块设计及接口.....	4
2.2 模块调用关系.....	4
3.详细设计.....	5
3.1 界面设计.....	5
3.2 数据库设计.....	9
3.2.1 数据字典.....	9
3.2.2 概念模型设计.....	11
3.3 关键技术.....	12
3.3.1 人脸信息比对.....	12
3.3.2 地图路径规划.....	13
3.3.3 摄像头调用.....	14
3.3.4 ECharts 可视化图表.....	14
3.3.5 网站部署.....	14
3.3.6 真实数据爬取.....	14
3.4 前后端设计.....	15
3.4.1 前端设计.....	15
3.4.2 后台设计.....	15
3.5 重要功能函数说明.....	15
3.5.1 __init__.....	15
3.5.2 face_detection.....	16
3.5.3 score.....	16
3.5.4 getMD5String.....	17
3.5.5 upload.....	17
3.5.6 send.....	17
4.测试报告.....	18
4.1 测试过程.....	18
4.2 技术指标.....	18
5.安装及使用.....	18
5.1 环境要求.....	18
5.2 安装流程.....	19
5.2.1 Python 端环境安装.....	19
5.2.2 Java 端环境安装.....	20
5.2.3 数据库导入.....	20
6.项目总结.....	20
6.1 项目优化.....	20
6.1.1 后续升级.....	20
6.1.2 功能扩展.....	21

6.1.3 商业推广21

6.2 小组感悟21

1.需求分析

1.1 项目背景

随着互联网大潮的推进,互联网技术逐渐进入医疗领域,各大医院也在积极进行互联网相关的改造,切实解决患者的就医问题。为帮助患者进行就诊挂号,节省挂号时间,方便患者快速就诊,本项目从生物特征识别技术中最实用、应用最广泛的人脸识别技术入手,旨在开发一款基于人脸识别的智慧医疗预约挂号平台。

1.2 用户分析

- 预挂号病患;
- 科室医生;
- 医院后台管理人员;

1.3 主要功能

- 1、人脸识别信息注册。挂号时运用“人脸识别注册”通过身份证+人脸识别,这种验证系统精准、科学地防止“号贩子”恶意注册并占用挂号资源,真实有效的方便实际病患挂号需求;
- 2、精准信息检索。用户可通过点击相应科室、疾病或模糊搜索,实现预约挂号;
- 3、地图路径规划。页面显示医院精准定位,用户可根据当前定位选择合适的驾车、公交、步行路径规划方式。

1.4 竞品分析

表 1. 竞品分析表

项目名称	功能特色	市场布局	数据情况	操作情况	界面情况
浙江在线预约诊疗服务平台	基本功能齐全	浙江省范围医院	数据齐全	查询复杂,页面刷新良好	页面信息繁杂
乌镇互联网医院	地图显示	全国部分医院	实际数据缺乏,部分查询无法显示	挂号流程较为繁琐	页面信息良好
绍兴市预约诊疗平台	基本功能齐全	绍兴市区医院	较为齐全	页面相应时间较长	页面体验感不佳

通过与表 1 所示三款同类型竞品分析得出，本作品具有以下优势：

- 1、基于人脸识别，采用最新的生物识别技术；
- 2、医院定位显示与路径规划；
- 3、页面简洁，操作方便；
- 4、应用对象范围偏小，为用户提供精准服务。

2.概要设计

2.1 模块设计及接口

根据需求分析结果将项目内容分为 7 大页面模块，内容及其相应接口划分如表 2 所示：

表 2. 模块设计表

模块名称	模块功能简述	模块接口
主页面	显示科室，疾病，以及主要信息，提供其他页面入口链接。	无
注册登录页面	用户进行注册，登录	注册、登录接口
信息管理页面	用户，医生，管理员的信息管理	用户、医生、管理员信息管理接口
科室选择页面	用户选择挂号所需的科室	科室选择接口
疾病选择页面	用户选择自身的疾病，引导至相应科室进行挂号	疾病选择接口
预约挂号页面	用户进行预约挂号	预约挂号、医生列表、排班信息接口
帮助页面	显示常见问题，注册登录指南，预约指南	常见问题、注册指南、预约指南接口

2.2 模块调用关系

根据所设计的功能模块，对各模块部分进行合理的逻辑流程划分，如图 1 所示：

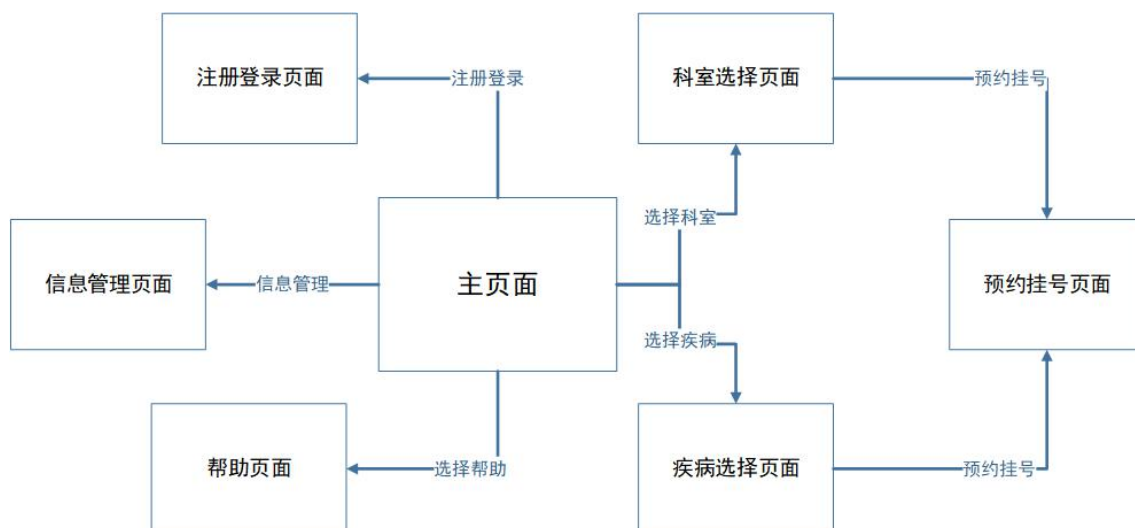


图 1.模块逻辑流程图

3.详细设计

3.1 界面设计

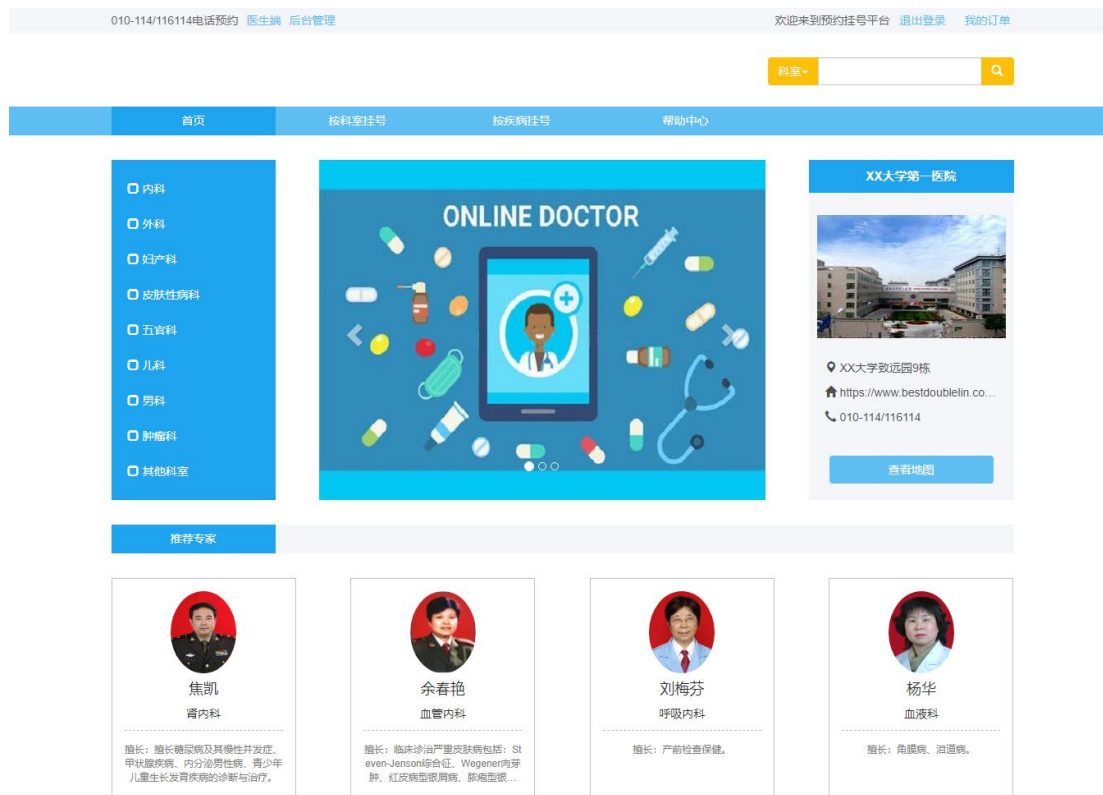


图 2.主界面图



图 3.按科室挂号图



图 4.按疾病挂号图



图 5.后台信息管理图

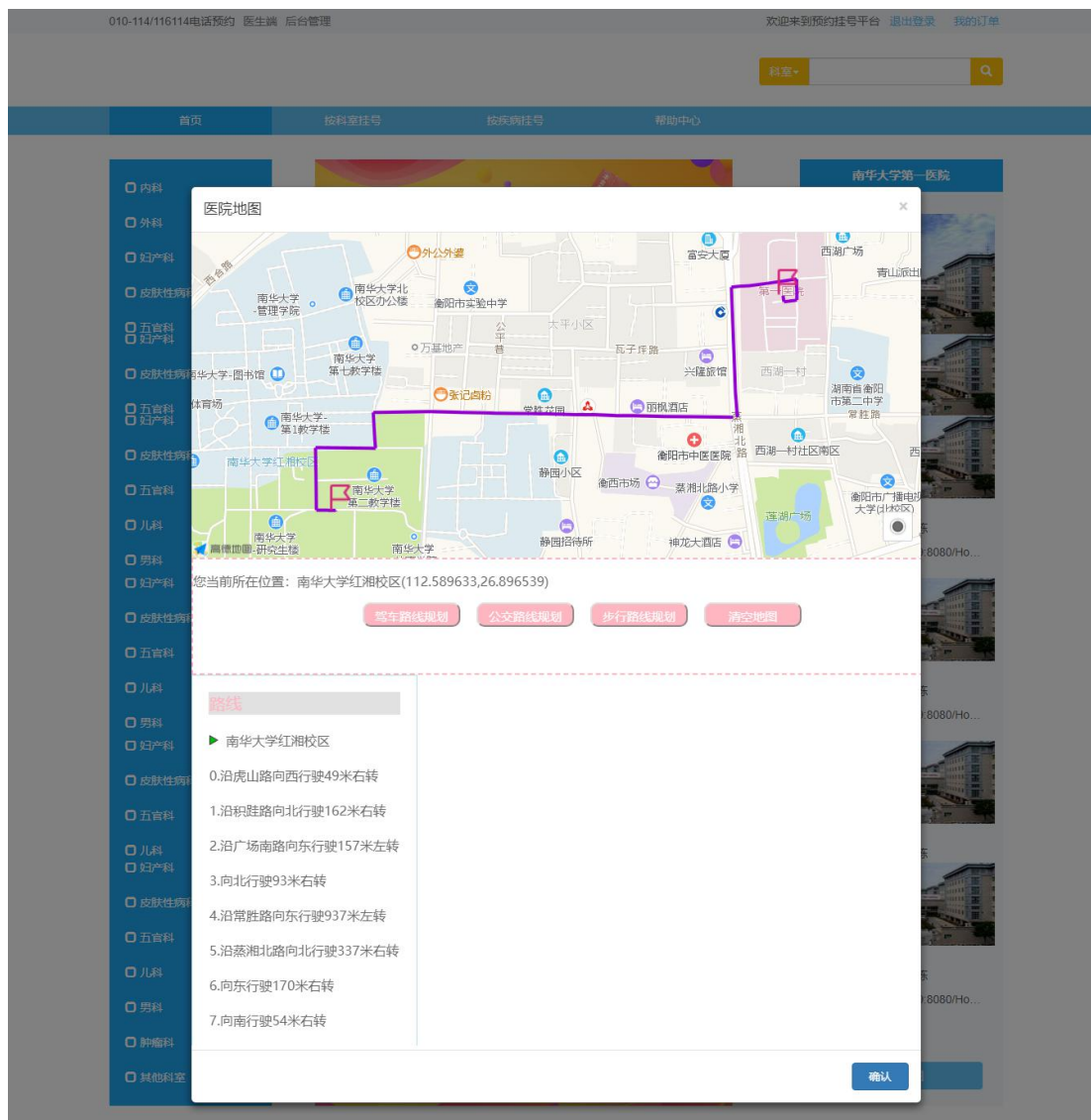


图 6.地图路径规划图

科室

Q

首页

按科室挂号

按疾病挂号

帮助中心

注册新用户

填写注册信息

确认注册信息

注册成功

身份验证：未通过身份验证

开始验证

姓名：

性别：

☒ 男 ☐ 女

身份证件：

身份证

证件号码：

确认号码：

设置密码：

请输入6-12位的字母或数字

确认密码：

现居住地：

请选择省份

请选择城市

请选择区县

手机号码：

邮箱地址：

邮箱验证码：

获取验证码

注册

图 7.账号注册图

3.2 数据库设计

3.2.1 数据字典

本项目数据分为 7 大部分，其中具体数据项的数据字典如下表所示：

表 3.用户表

表名	hos_user			
列名	数据类型	空/非空	约束条件	中文名称
usid	int	非空	主键	用户编号
uname	varchar (20)	非空		姓名
sex	varchar (20)	非空		性别
cardtype	varchar (20)	非空		证件类型
cardid	varchar (100)	非空		证件号码
pwd	varchar (100)	非空		密码
address	varchar (20)	非空		地址
tel	varchar (100)	非空		手机号
email	varchar (100)	非空	唯一	邮箱
photo	varchar (100)	非空		照片

表 4.科室表

表名	hos_ks			
列名	数据类型	空/非空	约束条件	中文名称
kid	int	非空	主键	科室编号
kname	varchar (20)	非空		科室名称
ktype	varchar (20)	非空		所属大类
hot	Int	非空		热门度

表 5.疾病表

表名	hos_jb			
列名	数据类型	空/非空	约束条件	中文名称
jid	int	非空	主键	疾病编号
kid	int	非空	外键	科室编号
jname	varchar (20)	非空		疾病名称
stype	varchar (20)	非空		小部位
btype	varchar (20)	非空		大部位
hot	Int	非空		是否常见

表 6.医生表

表名	hos_doctor			
列名	数据类型	空/非空	约束条件	中文名称
did	int	非空	主键	医生编号
kid	int	非空	外键	科室编号
msg	varchar (200)	空		简介
pic	varchar (100)	非空		照片

表 7.日程表

表名	hos_schedule			
列名	数据类型	空/非空	约束条件	中文名称
sid	int	非空	主键	日程编号
did	int	非空	外键	医生编号
sdate	date	非空		日期
timetype	varchar (20)	非空		所属时间段
rescount	Int	非空		可预约总数

表 8.黑名单表

表名	hos_blacklist			
列名	数据类型	空/非空	约束条件	中文名称
bid	int	非空	主键	黑名单编号
photo	varchar (100)	非空		照片
btype	varchar (20)	非空		类型

表 9.订单表

表名	hos_order			
列名	数据类型	空/非空	约束条件	中文名称
oid	int	非空	主键	订单编号
uid	int	非空	外键	用户编号
sid	int	非空	外键	日程编号

3.2.2 概念模型设计

对网站使用用户所需的现实使用情况，通过对其中所处分类、聚类和概括，建立抽象的概念数据模型。网站全局 ER 图如图 8 所示：

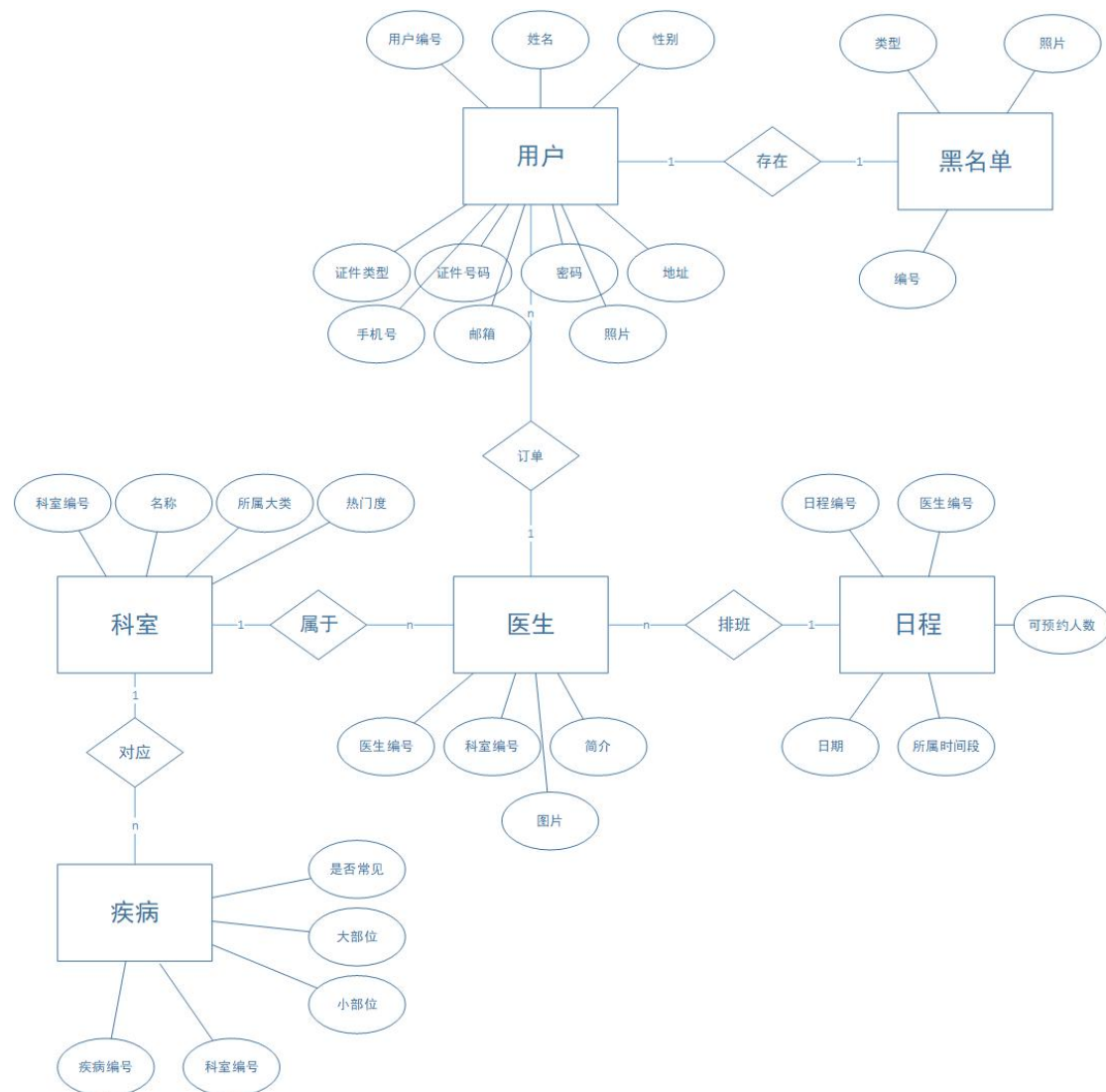


图 8.全局 ER 图

3.3 关键技术

3.3.1 人脸信息比对

本项目采用三种模型集成测试的思路，确保人脸识别的精确性。

1、首先采用 dlib 模型和感知 hash 模型进行人脸比对。使用 dlib 中提供的人脸检测方法（使用 HOG 特征或卷积神经网络方法），并使用提供的深度残差网络（ResNet）实现实时人脸识别，不过本项目的目的不是构建深度残差网络，而是利用已经训练好的模型进行实时人脸识别，实时性要求一秒钟达到 10 帧以上的速率，并且保证不错的精度。

人脸识别分为人脸检测和识别两个阶段，人脸检测会找到人脸区域的矩形窗口，识别则通过 ResNet 返回人脸特征向量，并进行匹配。

（1）人脸检测阶段。人脸检测算法需要用大小位置不同的窗口在图像中进行滑动，然后判断窗口中是否存在人脸。dlib 中使用的是 HOG（histogram of oriented gradient）+ 回归树的方法，使用 dlib 训练好的模型进行检测效果要好很多。dlib 也使用了卷积神经网络来进行人脸检测，效果好于 HOG 的集成学习方法。

（2）识别阶段。识别也就是我们常说的“分类”，摄像头采集到这个人脸时，让机器判断是张三还是其他人。分类分为两个部分：

- 特征向量抽取。本文用到的是 dlib 中已经训练好的 ResNet 模型的接口，此接口会返回一个 128 维的人脸特征向量。
- 距离匹配。在获取特征向量之后可以使用欧式距离和本地的人脸特征向量进行匹配，使用最近邻分类器返回样本的标签。

根据上述内容，识别的大致过程如图 9 所示：

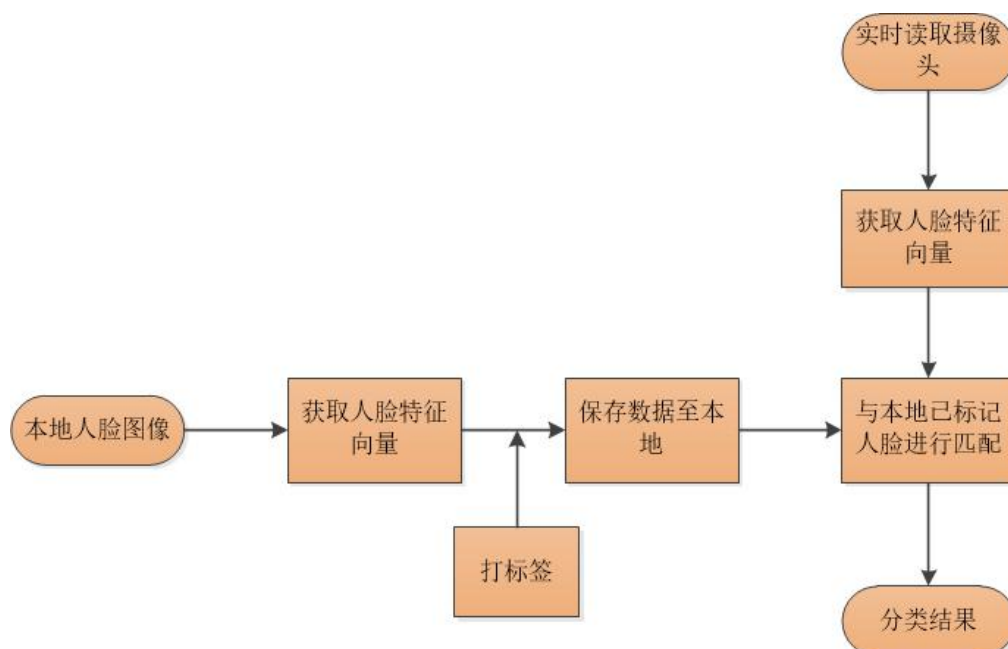


图 9.人脸识别分类过程

对于图 x 中的获取人脸特征向量，其过程如图 10 所示：

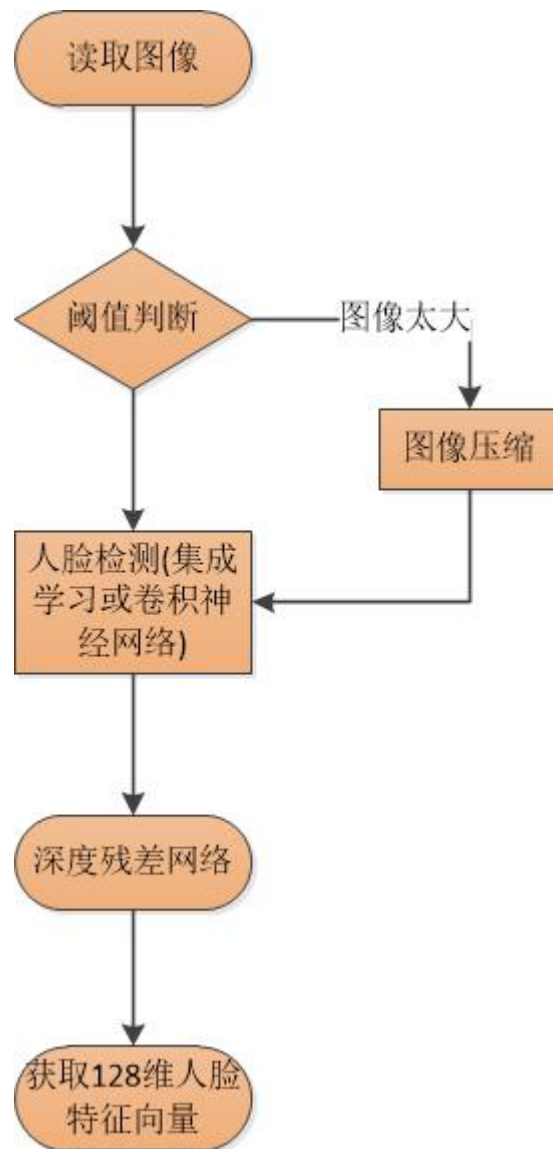


图 10.获取人脸特征向量过程

2、采用百度 api 人脸比对，将摄像头所拍摄的照片与数据库中黑名单人员进行人脸比对，缩短检测运行时间。

- 两张人脸图片相似度对比：比对两张图片中人脸的相似度，并返回相似度分值；
- 多种图片类型：支持生活照、证件照、身份证芯片照、带网纹照四种类型的人脸对比；
- 活体检测控制：基于图片中的破绽分析，判断其中的人脸是否为二次翻拍（举例：如用户 A 用手机拍摄了一张包含人脸的图片一，用户 B 翻拍了图片一得到了图片二，并用图片二伪造成用户 A 去进行识别操作，这种情况普遍发生在金融开户、实名认证等环节。）；
- 质量检测控制：分析图片的中人脸的模糊度、角度、光照强度等特征，判断图片质量；

3.3.2 地图路径规划

路径规划 API 是一套以 HTTP 形式提供的步行、公交、驾车查询及行驶距离计算接口，返回 JSON 或 XML 格式的查询数据，用于实现路径规划功能的开发。由于道路/数据/算法

的变更，很可能存在间隔一段时间后请求相同起终点的经纬度返回不同结果。

第一步，申请“Web 服务 API”密钥（Key）；

第二步，拼接 HTTP 请求 URL，第一步申请的 Key 需作为必填参数一同发送；

第三步，接收 HTTP 请求返回的数据（JSON 或 XML 格式），解析数据。

如无特殊声明，接口的输入参数和输出数据编码全部统一为 UTF-8。

针对步行、公交、驾车这三种路径规划的接口，实现批量请求，使用批量请求接口 (<https://lbs.amap.com/api/webservice/guide/api/batchrequest>)

3.3.3 摄像头调用

由于本项目需要将网站部署至云服务器上，所以此处摄像头的调用不能直接使用 eclipse 中的 jsf 依赖包，而是需要使用 webcam-capture 访问网络摄像头，通过用户所使用的网络摄像头来实现在线拍照功能。

webcam-capture 可以通过 JavaScript 直接使用小型 API 访问网络摄像头- 或者更确切地说是 jQuery。因此，可以将图像置于 Canvas（回调模式），将图像存储在服务器上（保存模式），并在 Canvas（流模式）上流式传输 Flash 元素的实时图像。

3.3.4 ECharts 可视化图表

ECharts 一款基于 HTML5 的图形库，图形种类丰富，风格搭配美观。

引用 ECharts 首先需要下载 ECharts.js 文件。ECharts 代码的编写基于 JavaScript,本项目通过 ajax 异步请求获取数据库数据作为可视化图表的数据来源，从而创建可视化图表统计医院一些相关数据在某一段时间内的变化，同时当数据库中数据更新的时候，图表中数据也能随之动态变化。

3.3.5 网站部署

本项目将 javaweb 端配置在华为云的云服务器 ubuntu16.04 系统下，同时申请域名 uscrzwj.cn 将域名与公网 ip 地址绑定，实现网站的域名访问，构造安全链接 https 形式。将后代源代码通过 Eclipse 打包为 war 形式上传至 tomcat8 服务器环境下的 webapps 文件夹中，实现 0.0.0.0:8080 端口可直接访问网站。其中所需要涉及的环境包括但不限于：JDK8 环境安装，mysql 数据库安装，python 相关版本依赖包以及工具安装。

3.3.6 真实数据爬取

本项目所有医生数据均由某医院信息库爬取得到，其中所用到的技术点包含但不限于：

- 浏览器分析 ajax 请求
- requests 模拟请求
- pyquery 解析
- pymysql 存储数据库
- Pool 进程池多线程爬取

开始爬取前安装好相应的依赖包，确定抓取目标的地址，选取所需要爬取的信息。首先抓取第一页的内容。我们实现了 `get_one_page()` 方法，并给它传入参数。然后将抓取的页面返回，通过 `main()` 方法调用，将所提取的结果写入文件，这里直接写入到一个文本文件中，通过 JSON 库的 `dumps()` 方法实现字典的序列化。最后将文本文件中的医生信息数据导入数据库，如前端代码相结合，实现前后端连接。

3.4 前后端设计

3.4.1 前端设计

前端倾向于呈现，着重处理用户体验相关的问题，前端使用 HTML/CSS/JavaScript 等开发技术，实现前端页面设计，具体前端页面设计如表 10 所示：

表 10.前端页面表

表名	说明
Homepage	主页面
Index	医生端信息管理页面
Index	管理员信息管理页面
Register	注册页面
Ks	科室查询页面
Jb	疾病查询页面
Search	简单预约页面

3.4.2 后台设计

后端倾向于业务逻辑、数据处理和持久化等。后端进行的业务逻辑升级，数据持久方案变更，只要不影响到接口，前端可以毫不知情。后台将前端人机交互所使用及产生的数据，通过 java 后端代码形式逻辑处理之后，存入 mysql 数据库，实现操作的方便性以及实用性。

3.5 重要功能函数说明

3.5.1 __init__

表 11.__init__ 表

<code>__init__</code>
函数说明：初始化模型配置环境

参数	参数说明	返回值	返回值说明
self	函数本身参数	NoneType	无
predictor_path	模型地址	NoneType	无
face_rec_model_path	模型地址	NoneType	无

3.5.2 face_detection

表 12.face_detection 表

face_detection			
函数说明：人脸相似度对比			
参数	参数说明	返回值	返回值说明
self	函数本身参数	NoneType	无
url_img_1	图片 1	NoneType	无
url_img_2	图片 2	NoneType	无

3.5.3 score

表 13.score 表

score			
函数说明：人脸相似度得分			
参数	参数说明	返回值	返回值说明
self	函数本身参数	NoneType	无
url_img_1	图片 1	NoneType	无
url_img_2	图片 2	NoneType	无

3.5.4 getMD5String

表 14.getMD5String 表

getMD5String			
函数说明：密码加密			
参数	参数说明	返回值	返回值说明
str	初始密码形式	BigInteger	密码加密形式

3.5.5 upload

表 15.upload 表

upload			
函数说明：文件上传			
参数	参数说明	返回值	返回值说明
pageContext	文件形式	NoneType	无

3.5.6 send

表 16.send 表

send			
函数说明：发送邮件			
参数	参数说明	返回值	返回值说明
address	发送地址	NoneType	无
subject	发送对象	NoneType	无

text	发送内容	NoneType	无
------	------	----------	---

4.测试报告

4.1 测试过程

表 17.测试过程表

测试编号	测试结果	修正
T-001	众多界面按钮错位	调整按钮位置
T-002	百度地图定位无法访问	使用 https 形式访问
T-003	注册界面摄像头无法启用	使用域名绑定 ip 地址
T-004	信息管理界面信息错误	调整 sql 语句
T-005	服务器部署链接错误	项目重新部署
T-006	无错误	无

4.2 技术指标

表 18.技术指标表

技术指标	说明
运行速度	页面切换相应时间 $t < 1s$
安全性	https 形式网站部署，严格保障网站安全
扩展性	项目通过接口连接形式，可后期扩展
部署方便性	网站部署至云服务器，较为方便
可用性	操作简便，界面简约大方

5.安装及使用

5.1 环境要求

表 19.环境要求表

分类	名称	版本
开发工具	pycharm	5

	eclipse	4.7
编译环境	python	3.6
	Java(TM) SE Runtime Environment	1.8.0
	Tomcat	9.0
重要库	cmake	3.14.3
	boost	1.70.0
	dlib	19.7.0
	numpy	1.16.2
	opencv-python	4.1.0.25
数据库平台	mysql	5.7.21
云服务器	ubuntu	Linux 16.04

5.2 安装流程

5.2.1 Python 端环境安装

1、安装 python。Windows、Linux 系统安装方式不同，具体方法如下所示：

①在 Windows 上安装 Python。首先，根据 Windows 版本（64 位还是 32 位）从 Python 的官方网站下载 Python 3.7 对应的 64 位安装程序或 32 位安装程序，然后，运行下载的 EXE 安装包。

②安装依赖包。首先安装 gcc 编译器，然后安装其他依赖包；下载 python3.6.0 源码，根据需求下载。

解压 Python-3.6.0.tgz：tar -zxvf Python-3.6.0.tgz

建立一个空文件夹，用于存放 python3 程序：mkdir /usr/local/python3

执行配置文件，编译，编译安装：

cd Python-3.7.0

./configure --prefix=/usr/local/python3

make && make install

安装完成没有提示错误便安装成功。

2、安装 python 重要依赖库。

①安装 opencv-python：pip install opencv-contrib-python

②安装 numpy：pip install numpy

③安装 dlib：pip install dlib

④下载 cmake，下载地址：<https://cmake.org/>

⑤下载 boost，下载地址：<https://www.boost.org/>

3、Window 环境下载 pycharm 软件，下载地址：<http://www.jetbrains.com/pycharm/>。Linux 环境则无需安装，可直接编译相应文件。

5.2.2 Java 端环境安装

1、安装 JDK 开发环境。下载网站：<http://www.oracle.com/>，根据电脑系统选择相应的安装版本。配置 JAVA 环境变量。

2、Eclipse 下载安装。下载网站：<https://www.eclipse.org/>，选择电脑系统匹配的 32 位或 64 位版本进行下载。

3、tomcat 下载安装与配置。下载网站：<https://tomcat.apache.org/>，下载相应 zip 文件，安装解压与环境变量配置。

5.2.3 数据库导入

1、首先建空数据库

```
mysql>create database dbname ;
```

2、导入数据库

方法一：

(1) 选择数据库

```
mysql>use dbname ;
```

(2) 设置数据库编码

```
mysql>set names utf8;
```

(3) 导入数据（注意 sql 文件的路径）

```
mysql>source /home/xxxx/dbname .sql;
```

方法二：

```
mysql -u 用户名 -p 密码 数据库名 < 数据库名.sql
```

6.项目总结

6.1 项目优化

6.1.1 后续升级

本项目具有较大的潜在开发空间，其中可本项目已包含的部分进行二次开发与性能优化，如：

- 1、人脸识别技术性能优化，提升图像对比速度，增强数据集测试。
- 2、地图导航与路径规划，后续添加多因素，实现更优路径推荐。
- 3、美化网站页面，缩短页面链接响应时间。

6.1.2 功能扩展

根据不同需求和应用环境及对象的变化,可进行功能扩展,实现功能适应性。若网站的业务范围扩大,实现多区域医院科室挂号,并且允许采集医院相关监控数据,则可进行下列功能添加:

- 1、基于树莓派的视频端采集数据,通过网站后台行为分析算法计算研究,判断当前用户行为是否违规(如医闹行为,斗殴行为等),并进行相应警报提示。
- 2、网站端添加基于自然语言处理的智能问答模块,将用户提问与标准问题匹配。
- 3、智能人脸识别追踪数据库中的黑名单人员在医院中的出现情况,作出相应机制处理。

6.1.3 商业推广

通过线上线下结合模式,对本项目进行商业推广。线下推广主要采用医院宣传、团队人员管理等手段,与相关医院进行合作的前提下,依靠精炼有趣的平面广告或软文进行现场宣传,达到推广目的。线上推广主要借助网络平台进行高性价比且快速的推广方案,包括各种网络调研、网络广告、网络分销、网络服务等网络推广活动。

6.2 小组感悟

任务分解: 本项目因存在核心创新技术点,所以任务点大致分为以下几点:项目创意设计、前端页面设计、核心算法设计与研究、前后端适配连接、信息采集与分析、网站部署与测试、文档编辑、视频制作、文档初审与复核。

项目协调: 根据任务分解,将各任务点划分相应模块并分配给相应开发人员。负责人负责项目创意与内容设计以及项目全局进度把控,将前端、后台部分分配至两位技术人员,两人进行前后端适配连接以及测试。全体开发人员抓准主要矛盾,合理解决问题,切实有效的实现项目保质保量完工。

通过本项目的开发,我们对此项目的开发从开始迷茫到现在的熟悉,加深了对项目开发流程以及相应技术的理解。其中我们遇到了不少困难,如前端开发页面逻辑设计,生物识别技术使用等,经过查阅资料并认真思考解决了这些困难。同时我们进一步懂得了如何让大家高效协作,来一起完成我们艰巨的任务,如何协调组员之间观点,团队的力量是无量的,一个好的团队可以攻破不可思议的难题。