《基于Android的云相册设计》项目文档

## 云相册设计流程图



**图1 流程图**

## 数据库设计

数据库名：mytestimg

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Imgs | | | | | | |
| 属性名 | filename | upload\_time | face\_token | userid | userinfo | groupid |
| 描述 | 文件名 | 上传时间 | 人脸唯一标识 | 人脸id | 人脸信息 | 人脸所属用户组 |

**表1 云相册数据库**

## 单元测试设计

### 3.1编写目的

1. 对单元测试结果进行整理和汇总，形成正确的测试文档。
2. 为软件单元的评审验收提供依据
3. 纳入软件产品配置管理库

### 3.2软件单元描述

|  |  |  |  |
| --- | --- | --- | --- |
| 被测试单元 | 与之相关的单元 | 单元需求 | 单元设计 |
| testDetectFaceWithUrl | BaiduApi | 能根据图片url来进行人脸识别 | 传入图片的URL，向百度发出请求进行人脸识别 |
| testDetectFaceWithBase64 | BaiduApi | 能根据图片base64编码来进行人脸识别 | 传入图片的base64，向百度发出请求进行人脸识别 |
| testAddFaceWithBase64 | BaiduApi | 将图片传入到百度人脸库 | 传入图片的base64编码，以及该相片的userinfo，加入到百度人脸库，以便后续的识别和搜索。 |
| testSearchWithBase64 | BaiduApi | 根据图片的base64编码在人脸库搜索该人脸信息 | 传入图片的base64能在百度人脸库返回结果 |
| MyAlbum | AlbumAdapter | 首页能根据不同人脸建立每个人的相册 | 更加人脸库的userlist建立每个人的相册，每个相册中有对应的照片。 |

### 3.3测试过程

|  |  |
| --- | --- |
| 被测试单元 | 测试过程 |
| testDetectFaceWithUrl | 传入一张有人脸的图片的URL，以及传入不含人脸的图片的URL，查看返回结果是否正确。 |
| testDetectFaceWithBase64 | 传入一张有人脸的图片的base64，以及传入不含人脸的图片的base64，查看返回结果是否正确。 |
| testAddFaceWithBase64 | 传入图片的base64编码，查看能否添加成功，以及groupid和userid和userinfo是否正确。 |
| testSearchWithBase64 | 传入图片的base64编码，查看能否搜索到匹配的人脸并返回相应信息，如果搜索不成功，返回提示信息。 |
| MyAlbum | 检测是否能得到人脸库所有用户的集合，测试能否从阿里云服务器中得到每个人的照片。 |

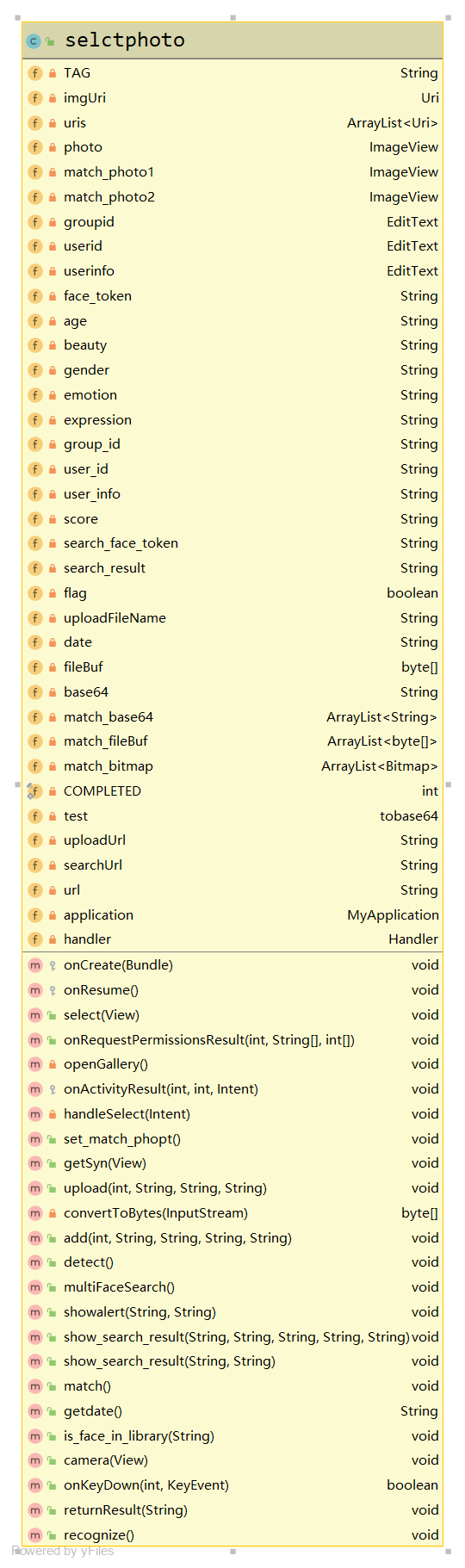
### 3.4测试结果

|  |  |  |
| --- | --- | --- |
| Bug ID | 问题描述 | 是否解决 |
| 1 | 打开相机拍照程序报错 | 是 |
| 2 | 相机拍照后进行人脸识别时，程序出错 | 是 |

## 四、类设计

### 4.1整体结构

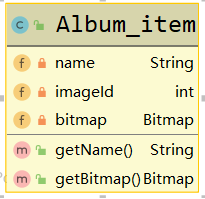
* selctphoto



包含拍照、选择照片、上传到图片库、人脸检测、人脸搜索、人脸对比、图像识别的点击调用事件。

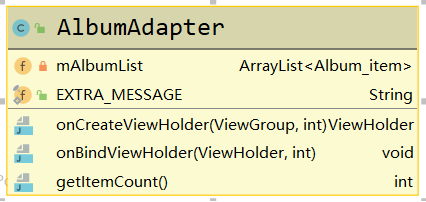
* Album\_item

recyclerView中的一项，包含bitmap和name两个属性。



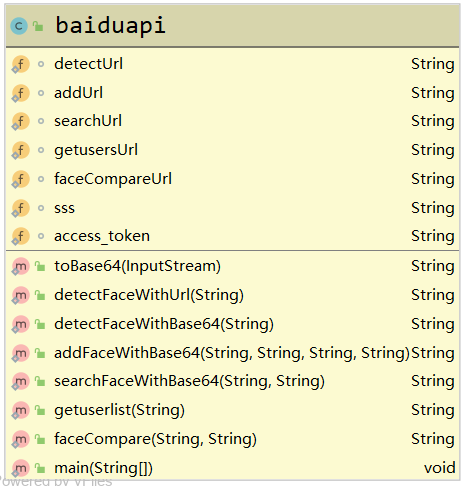
* AlbumAdapter

AlbumAdapter，用于设置recyclerview的适配器。



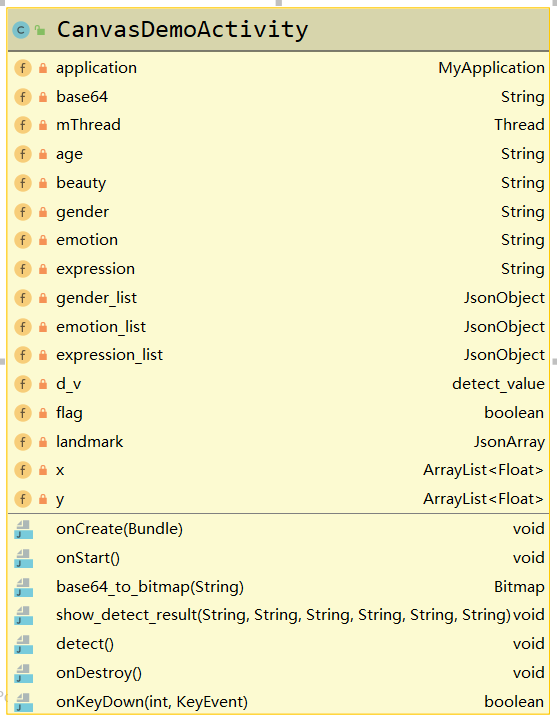
* baiduapi

访问百度API的接口，包括人脸检测，人脸对比，人脸搜索，人脸注册，获取用户列表。



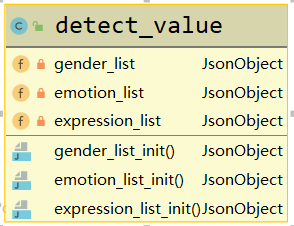
* CanvasDemoActivity

人脸检测的脸部框选功能，利用百度API返回的72个关键点，在画布上画出人脸区域。



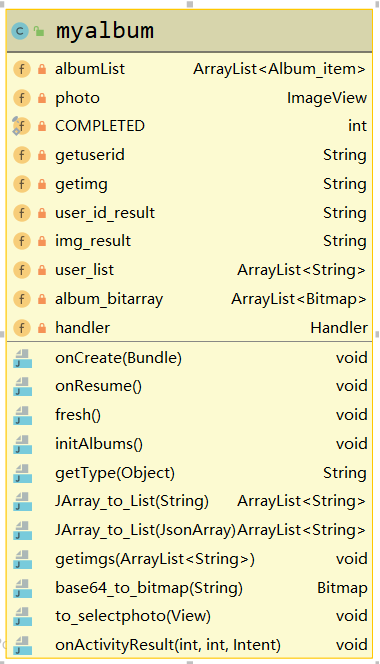
* detect\_value

包含三个json对象，把百度API的返回值转成对应的中文。



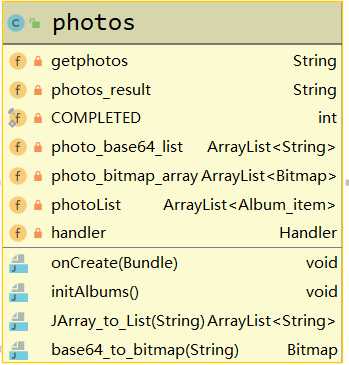
* myalbum

相册。从百度api获取人脸库用户列表，再根据获取的列表每个用户取一张图片，利用recyclerview显示到页面。



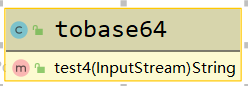
* Photos

根据user\_id，从服务器下载同一个人的所有照片。

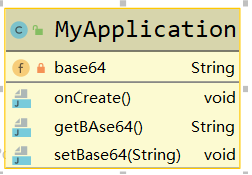


* tobase64

把数据流转化成base64



* MyApplication



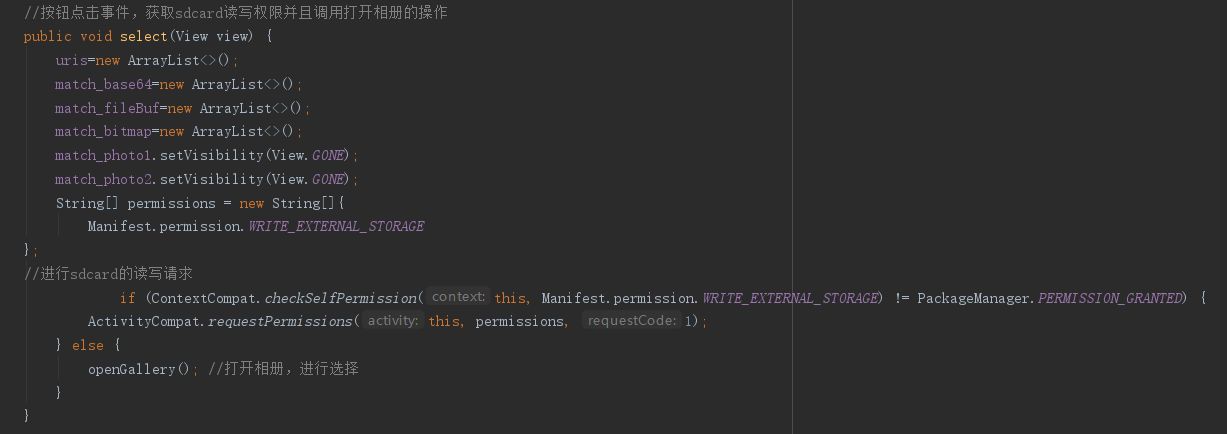
### 4.2从相册中读取照片并显示select

**类设计思路：**

1. 点击从相册选择图片时，首先要获取sdcard的权限。
2. 权限通过时，打开《相册》进行选择（系统会将选择到的图片以Uri封装，存入到intent对象中）。
3. 选择结束时，从intent中获取uri(intent.getData())， 再利用ContentResolver从uri中获取文件的绝对路径，最终显示到ImageView中。
4. 选择多张照片后，显示在不同的ImageView中。

**具体实现过程：**

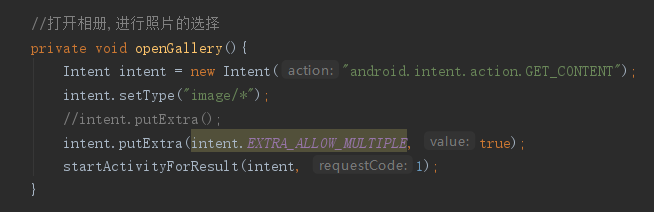
1. sdcard的读写权限申请



1. 获取权限

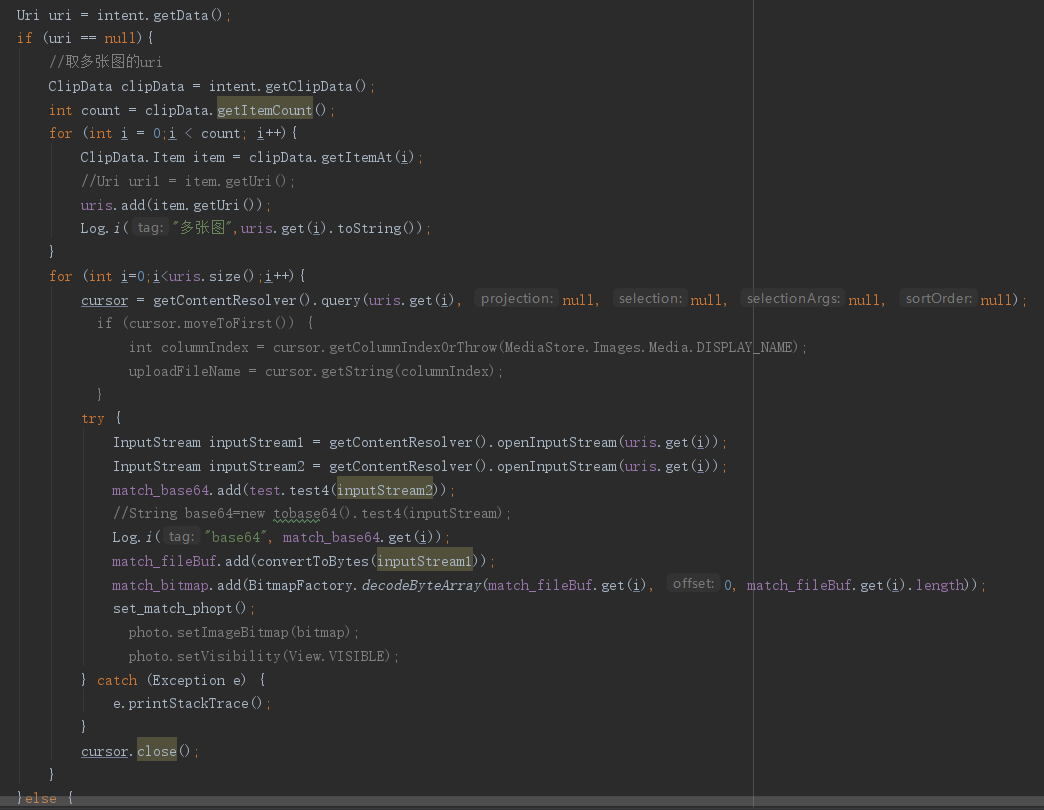


1. 打开相册



1. 进行图片的选择（一张或者多张）

系统会将选择到的图片以Uri封装，存入到intent对象中，采用getClipData()获取多张图片的uri，再将图片转成位图或base64格式，存入列表数组match\_bitmap或者字节数组match\_base64中，以备后续使用。



### 4.3调用相机拍照并显示camera

**类设计思路：**

1. 首先删除并创建临时文件File，用于保存拍照后的照片。
2. 获取uri，以及打开相机的权限
3. 利用actionName和Extra，启动相机Activity

**具体实现过程：**

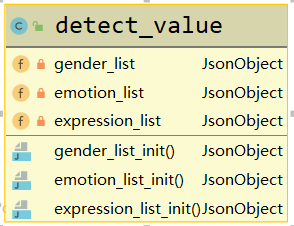


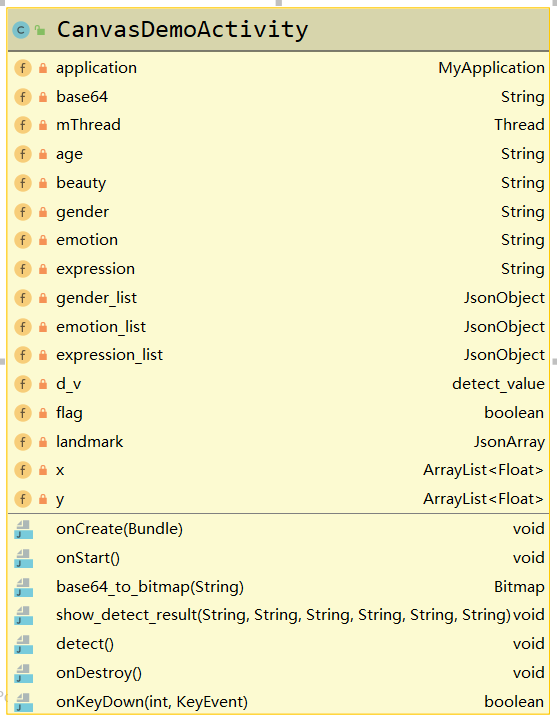
### 4.4人脸检测detect

**类设计思路**：

1. 从相册中选择一张图片，或者调用相机拍照获取图片，将其显示在ImageView中（从相册选择图片并显示和调用相机拍照并显示见上文）。
2. 点击人脸检测后，调用百度人脸检测API检测判断是否为包含人脸的图片，如果是人脸图像则根据人脸的像素点利用Canvas画布将人脸区域框出来，在自定义的view中显示，并利用AlertDialog将人脸的基本信息弹窗显示。
3. 其中在点击人脸检测之前，添加错误控制，防止用户未选择图片就进行检测，造成闪退。

**类图展示：**

****

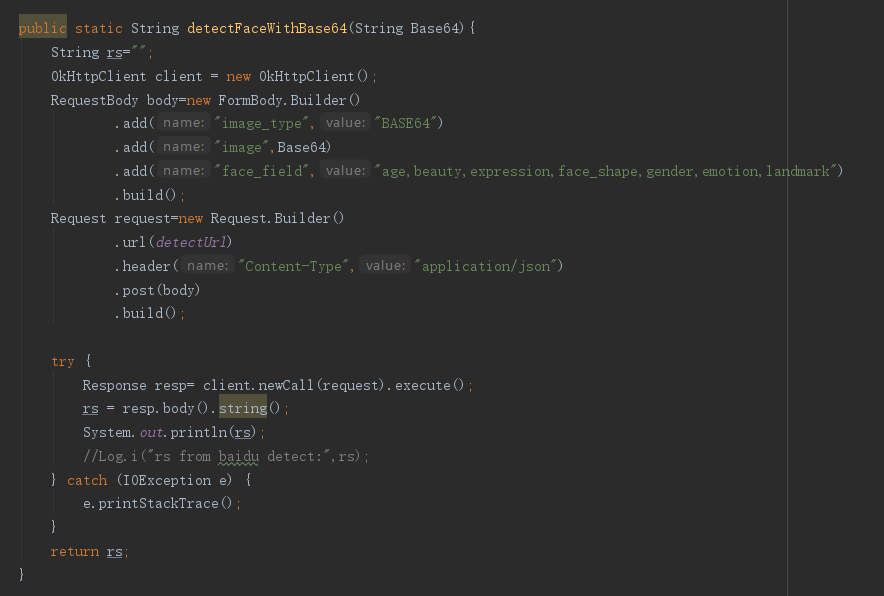
****

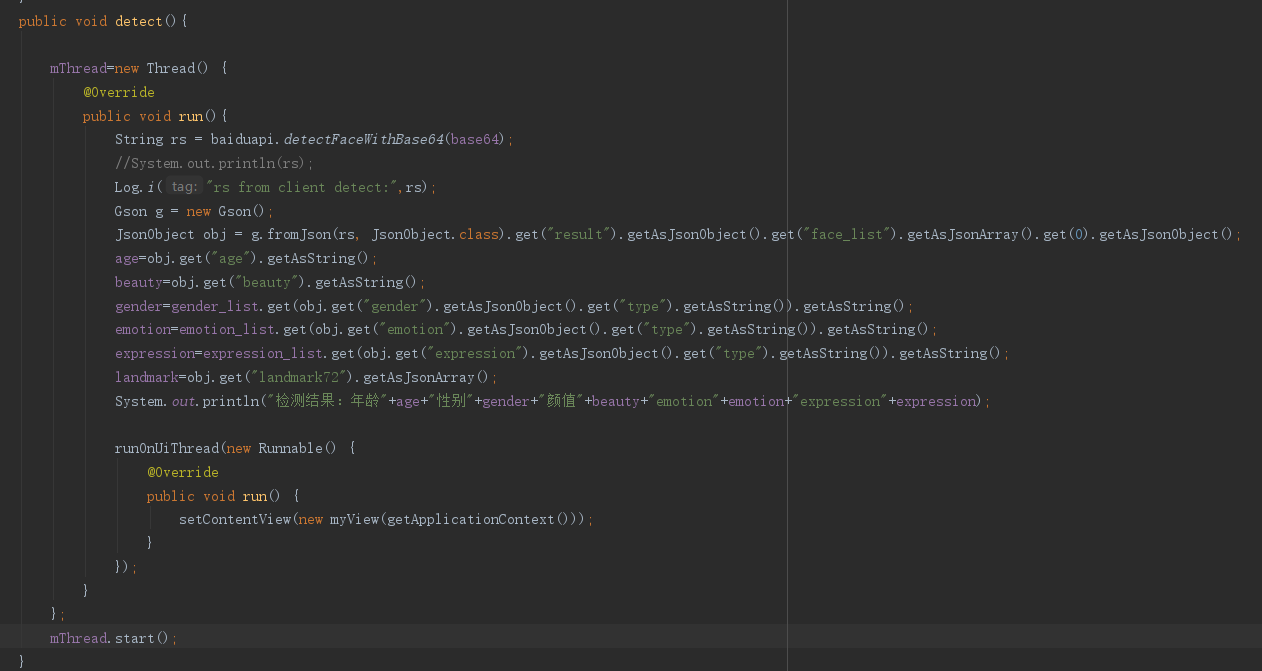
**具体实现过程：**

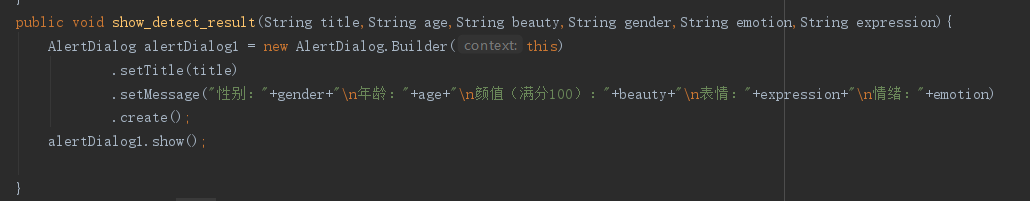
1. 错误控制



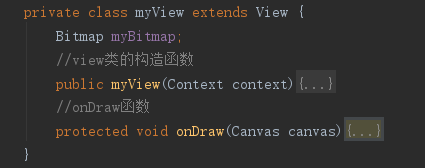
1. 人脸检测detectFaceWithBase64()，采用Okhttp方式访问百度云服务器，向API服务地址使用POST发送请求，通过返回的json对象得到人脸检测结果，并将人脸检测信息的显示show\_detect\_result()

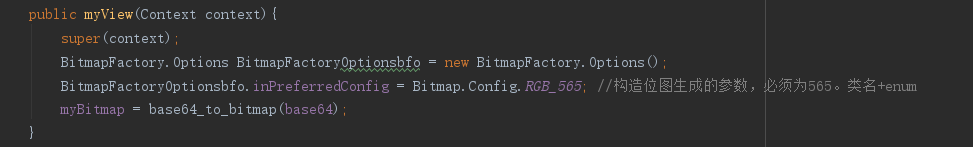






1. 人脸检测后对人脸区域框图ondraw()，并在自定义的view中显示



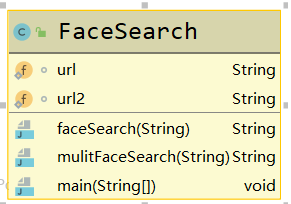


### 4.5人脸搜索facesearch、mulitFaceSearch

**类设计思路：**

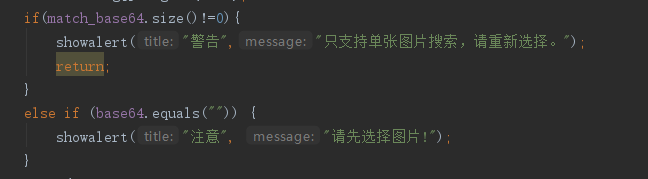
1. 从相册中选择一张图片，或者调用相机拍照获取图片，将其显示在ImageView中（从相册选择图片并显示和调用相机拍照并显示见上文）。
2. 点击人脸搜索后，调用百度人脸搜索API，在百度云人脸库中进行搜索，将搜索结果利用AlertDialog弹窗显示，如果人脸库中没有该人脸信息则返回未找到，如果有该图片则显示其在人脸库中的信息。
3. 其中包含多人脸的检测搜索，如果图片中包含多个人脸，则会根据mulitFaceSearch()，进行多人脸检测，
4. 其中在点击人脸检测之前，添加错误控制，防止用户未选择图片就进行检测，造成闪退。

**类图展示：**

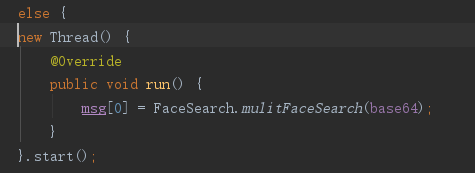
****

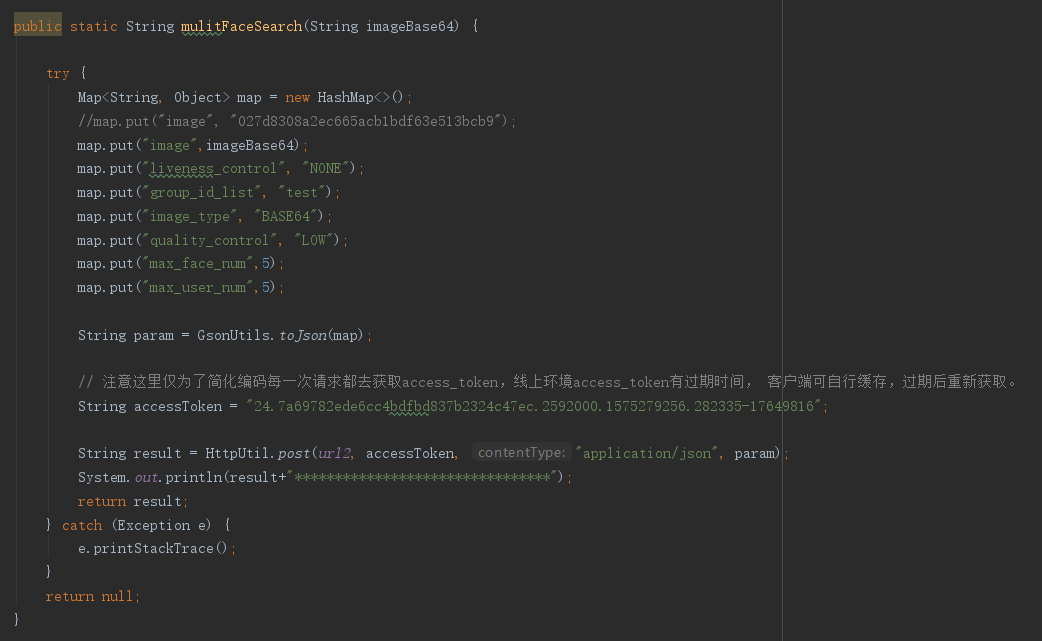
**具体实现过程：**

1. 错误控制

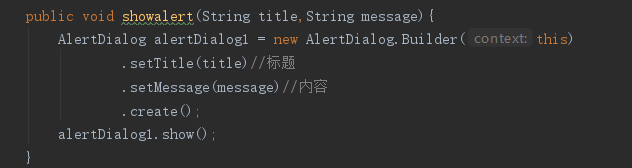


1. 多人脸搜索mulitFaceSearch()





1. 利用json传回结果，将匹配度大于80的图片显示出其在人脸库中的位置信息。若匹配度低于80的则认为该人物信息不在库中存在。
2. 人脸搜索结果弹窗展示showalert()

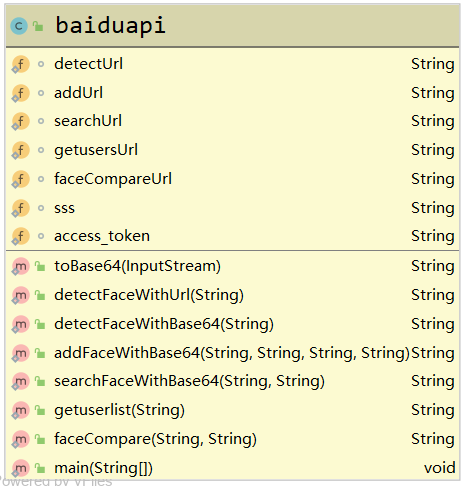


### 4.6人脸对比faceCompare

**类设计思路：**

1. 在相册中选择多张图片，显示在ImageView中。（从相册中选择多张图片并显示的方法见上文）
2. 点击时，调用百度API对选择的两张图片进行对比，然后利用AlertDialog弹窗显示对比结果，得出人脸匹配结果。
3. 在点击检测前，要有错误控制，防止未选照片就进行对比操作，造成程序闪退。

**类图展示：**

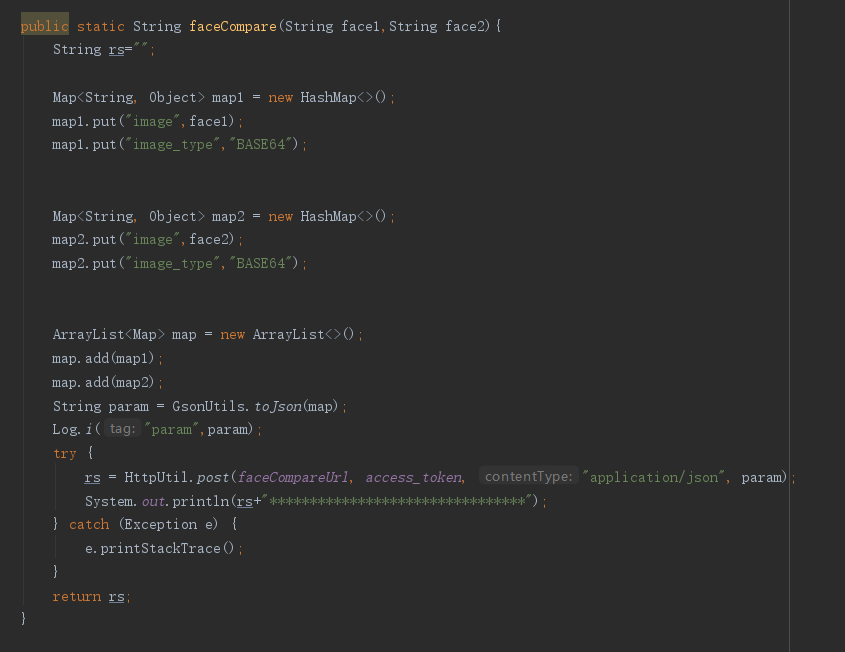


**具体实现过程：**

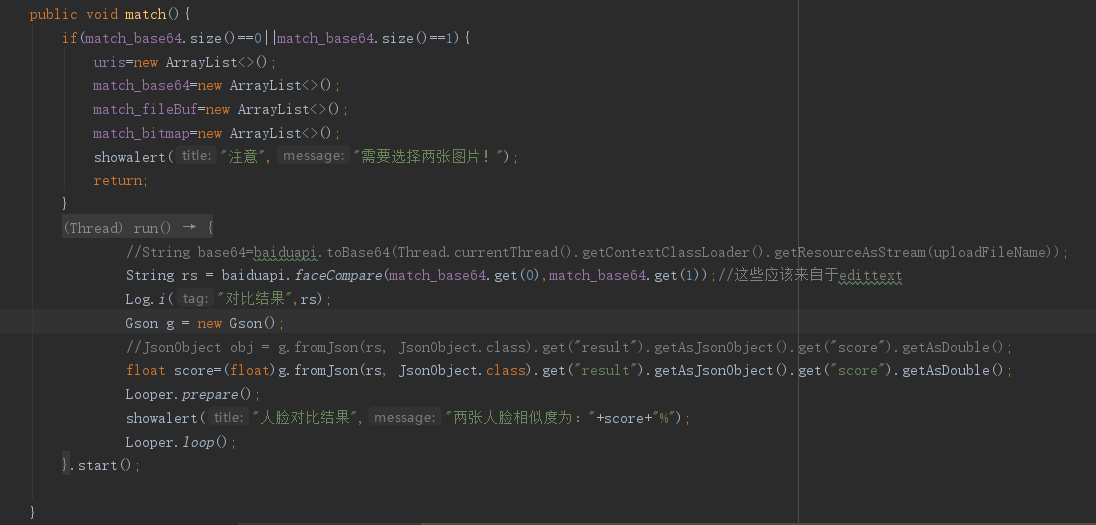
1. 错误控制



1. 多张图片的选择handleSelect()，见上文从相册中读取照片并显示
2. 人脸对比faceCompare()，利用图片的base64编码格式，需要将



1. 将json传回的结果在showalert()中显示。



### 4.7图片识别recognize

**类设计思路**：

1. 从相册中选择一张图片，或者调用相机拍照获取图片，将其显示在ImageView中（从相册选择图片并显示和调用相机拍照并显示见上文）。
2. 点击人脸检测后，调用百度人脸检测API检测判断图片中的信息，对返回的json数据解析，并将信息弹窗显示
3. 其中在点击图片识别之前，添加错误控制，防止用户未选择图片就进行检测，造成闪退。

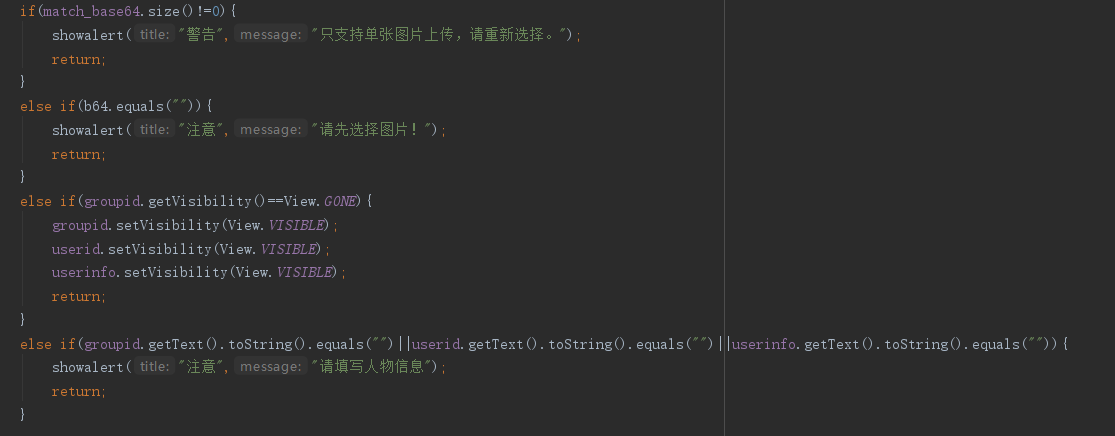
### 4.8人脸注册add

**类设计思路：**

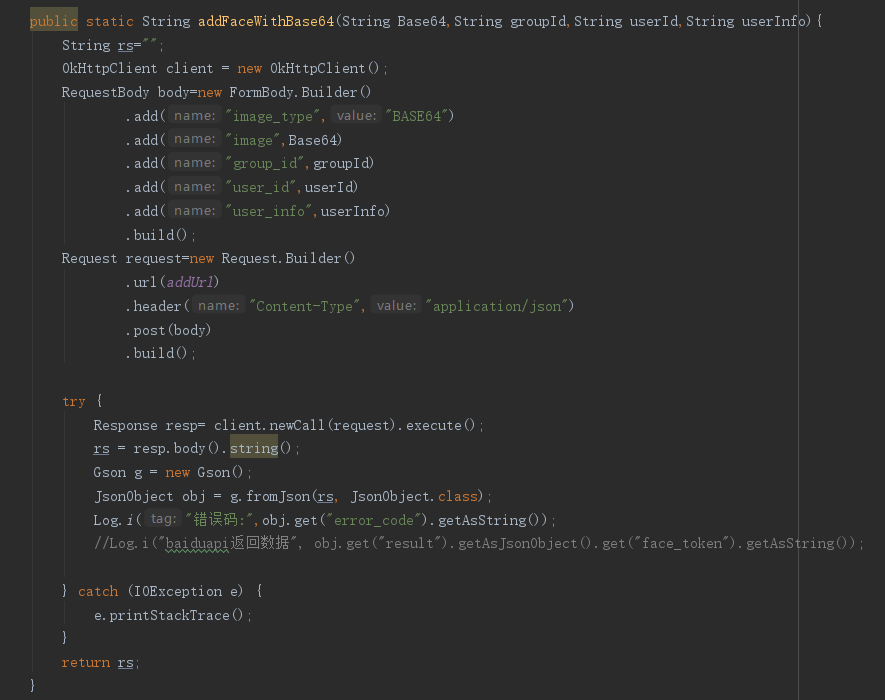
1. 点击图片添加到图片库时，首先要设置错误控制，要求用户必须先选择照片并且填写groupid，userid，userinfo信息。
2. 然后利用图片的base64格式以及填写的信息请求百度人脸注册接口，将图片添加到人脸库对应的位置。
3. 在添加时，若库中存在该图片，则提醒。若不存在，则添加成功且同时添加在数据库中。

**具体实现过程：**

1. 错误控制



1. 添加到人脸库





## 五、网络接口设计

|  |  |
| --- | --- |
| 请求URL | 112.124.46.47/routes/getimg |
| 功能描述 | 根据得到的请求参数userid，获取人脸库的用户的照片(一张) |
| URL参数 | userid，通过请求百度API获取 |
| HTTP方法 | GET |
| 请求参数 | 无 |
| 返回说明 | 返回一张图片的base64编码  数据示例：{“sfasq35sadvsvqwr5q...”} |

|  |  |
| --- | --- |
| 请求URL | 112.124.46.47/routes/search |
| 功能描述 | 判断图片是否已经存入数据库 |
| URL参数 | face\_token |
| HTTP方法 | GET |
| 请求参数 | 无 |
| 返回说明 | 如果存在，返回文件名；不存在则返回空JsonArray |

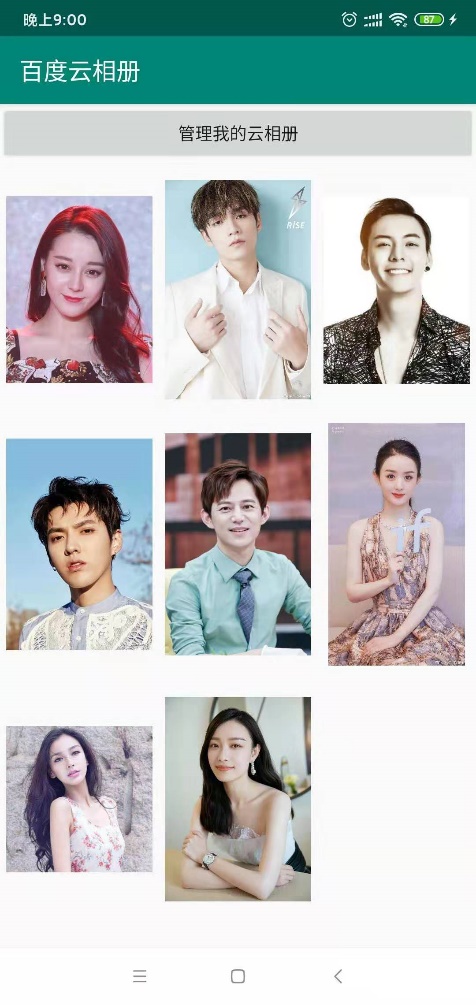
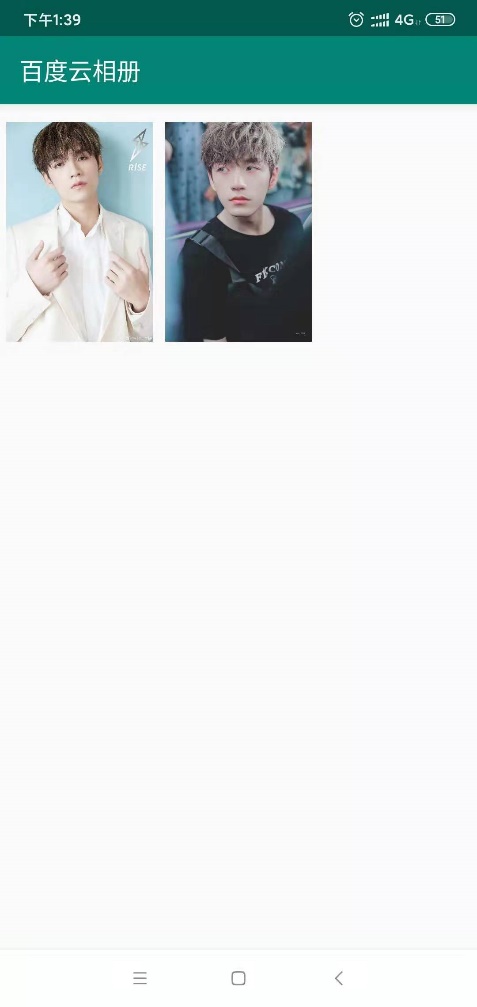
|  |  |
| --- | --- |
| 请求URL | 112.124.46.47/routes/getphotos |
| 功能描述 | 返回某用户的所有照片的base64编码 |
| URL参数 | userid |
| HTTP方法 | GET |
| 请求参数 | 无 |
| 返回说明 | 返回JasonArray  示例：[“sfasq35sadvsvqwr5q...”,” sfasq35sadvsvqwr5q...”,……] |

|  |  |
| --- | --- |
| 请求URL | 112.124.46.47/routes/img |
| 功能描述 | 接收客户端上传的图片文件并保存，且将相应信息保存在数据库中 |
| URL参数 | 无 |
| HTTP方法 | POST |
| 请求参数 | 普通表单+文件上传域  示例：{“title”:”photo1”,“groupid”:”test”,”userid”:”yy”,”userinfo”:”杨颖”,”face\_token”:” 35sadvsvqwr5q...”,”upload\_time”:”2019…”} |
| 返回说明 | 返回上传结果信息 |

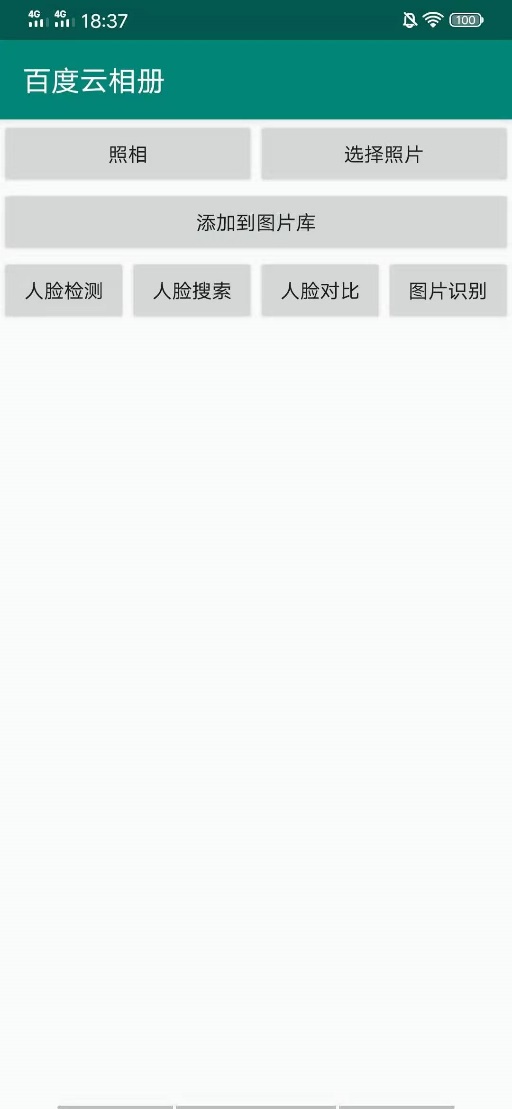
## 六、软件安装以及使用说明

本软件是一个基于百度人脸识别的云相册，相册来自于用户自主建立的人脸库。具有相册展示、手动注册人脸、自动注册人脸、人脸检测、人脸识别、人脸搜索、人脸对比、图片识别的功能。

**主页**：按不同用户分类相册，点击任意图片进入二级相册，可以查看该用户的所有照片

**管理界面**：点击“管理我的云相册”按钮进入本界面。功能有：人脸检测，人脸对比，人脸搜索，图片识别。图片来源可以是拍照，也可以是从相册选择。

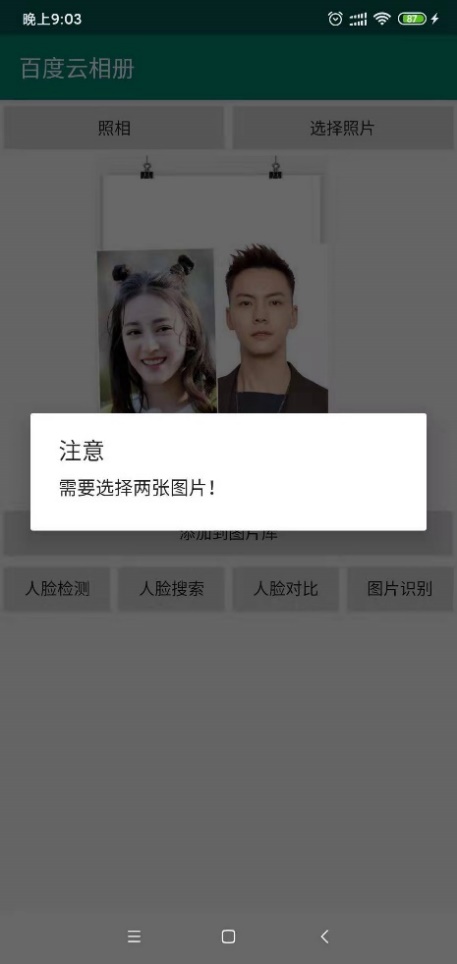
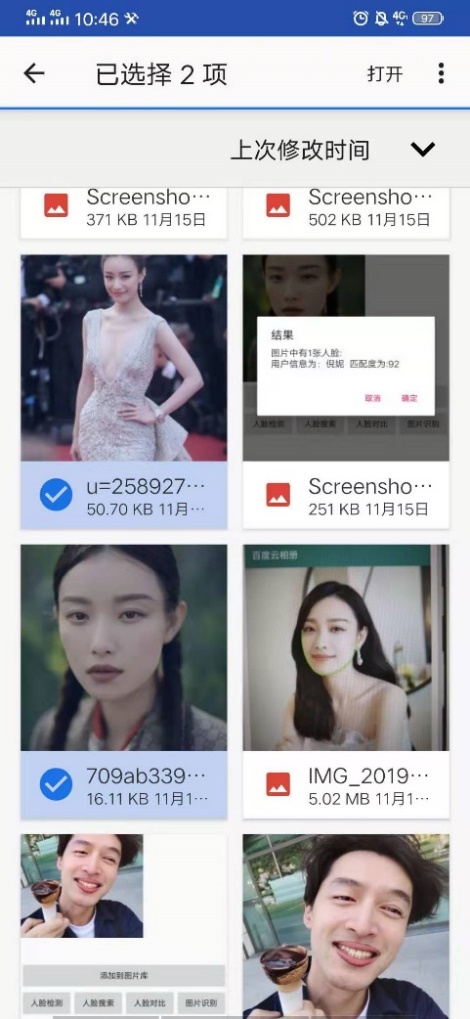
所有功能使用前必须照相或选择图片，否则出现以下错误：

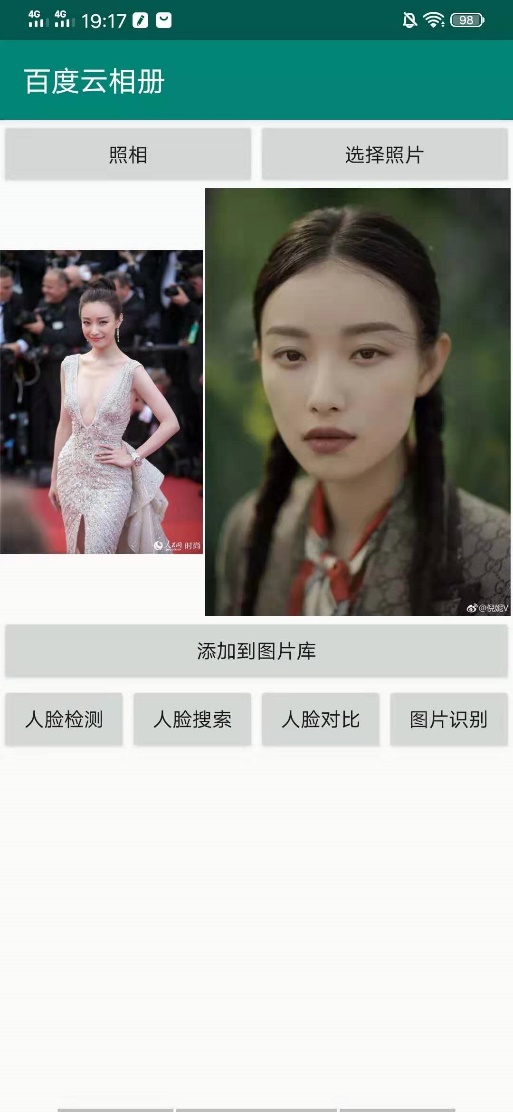
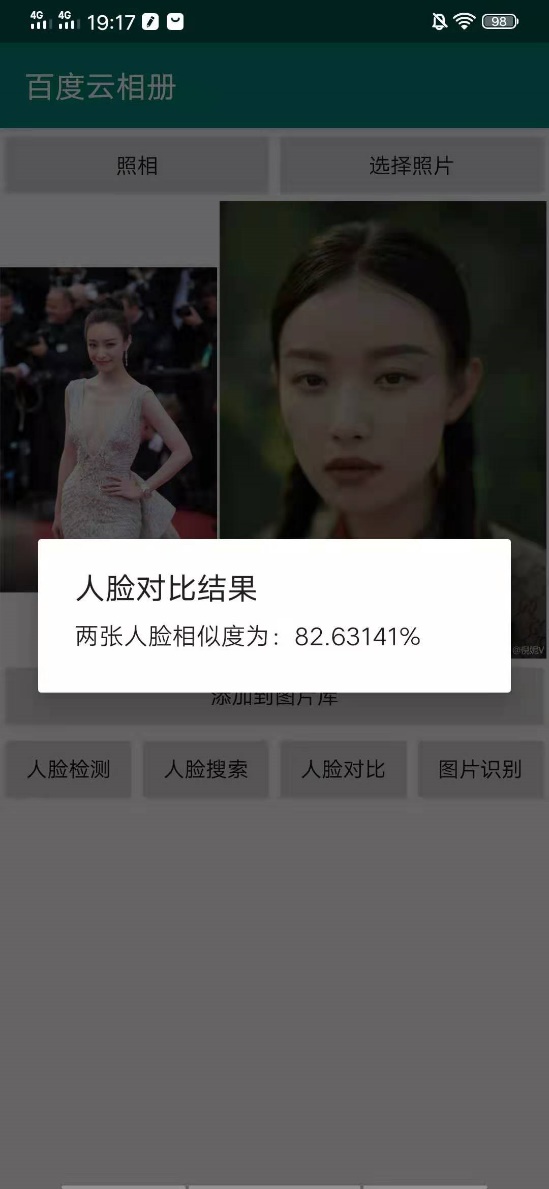


人脸检测：可框选人脸和人物相关信息 人脸搜索：可显示人物的身份和匹配度

人脸对比（需要选择两张图片）：

图片识别：