﷽﷽﷽﷽﷽﷽﷽﷽﷽﷽﷽﷽﷽﷽﷽﷽k﷽百度人脸SDK Android开发手册

v2.1

1. 简介

百度FaceSdk Android 版 SDK是一种面向 Android 移动设备人脸解决方案,目前包含人脸检测、活体识别等功能，以 AAR 包 + 动态链接库的形式发布。 基于该方案,开发者可以轻松的构建包含人脸检测和活体识别的APP。

在您使用SDK之前，我们首先为您介绍一下SDK本身及相关人脸能力，以便您能更方便使用人脸服务。

* **本地版活体检测**：通过让用户做出指定脸部交互动作，识别当前操作者是否为活体，此功能可离线使用，可设定指定动作是否使用及应用顺序。可有效抵御高清图片、3D建模、视频等攻击。
* **在线版活体检测**：基于本地版有交互活体检测输出的质量合格的图片，请求在线活体检测接口，可以进一步判断用户上传的图片是否为活体，进一步抵御高精度视频、高精度建模等攻击。此接口建议与本地版活体检测结合使用，且输入的照片一定为二次采集图片（即限于产品前端拍照所得，不可为相册图片）。
* **人脸质量检测**：判断视频流中的图片帧中，哪些图片质量较佳，即人脸图像特征清晰（满足角度、姿态、光照、模糊度等校验）。
* **人脸图像采集**：通过本地SDK能力，采集人脸图像，同时经过人脸质量检测，确保采集到的人脸图像符合各条件校验（角度、姿态、光照、模糊度等），为设备前端获取有效可分析人脸的主要功能。
* **在线鉴权**：API接口的权限判断，在后台获取API Key和Secret Key后生成的token，用于每次请求时供服务器判断是否可通过此次请求。
* **离线授权**：SDK的授权判断，授权介质也称为license，在SDK使用中，需要通过license向授权服务器发起请求，判断SDK的使用合法性及使用有效期。
* **人脸识别**：人脸识别技术的泛称，通常指识别图片或视频中的人脸，提取人脸特征值，用于进一步的人脸对比、搜索等业务操作。
* **人脸检测**：分析并定位图片中的人脸及位置，是人脸识别技术的第一步分析内容。
* **人脸1：1**：目标分析人脸与已知身份人脸进行比对，对比两个图片中的人脸的相似度，根据分值判断两张人脸是否为同一人。
* **人脸1：N**：在一个已知身份的人脸集合中找到相似的人脸，返回内容为人脸集合中每一张脸与目标人脸的相似度分值，找出目标人脸是否属于哪一个已知身份的用户。
* **人脸属性**：通过人脸特征抽取，分析人脸的性别、年龄、种族、是否戴眼镜等属性值。

2.1 功能介绍

FaceSDK支持下列功能（离线使用）：

有动作活体检测： 通过用户做出指定交互动作，判断用户是否为真人，动作包含眨眼、张嘴、左摇头，右摇头，摇摇头、向上抬头，向下低头七个。

* 人脸检测：检测当前镜头前是否存在人脸，并标记出人脸轮廓位置和关键点。
* 人脸质量检测：对采集到的人脸进行光照、姿态角度、模糊度等条件的判断，确保采集到人脸符合可分析的质量。
* 人脸采集：捕获已检测到的符合质量检测的人脸图像。
* 人脸跟踪：对当前检测到的人脸持续检测跟踪。

2.2兼容性

系统:支持 Android 4.0.3(API Level 15)及以上系统。需要开发者通过 minSdkVersion 来保证支持系统的检测。

机型：手机和平板皆可

构架：支持 CPU架构平台【arm-v7,arm-64，x86】

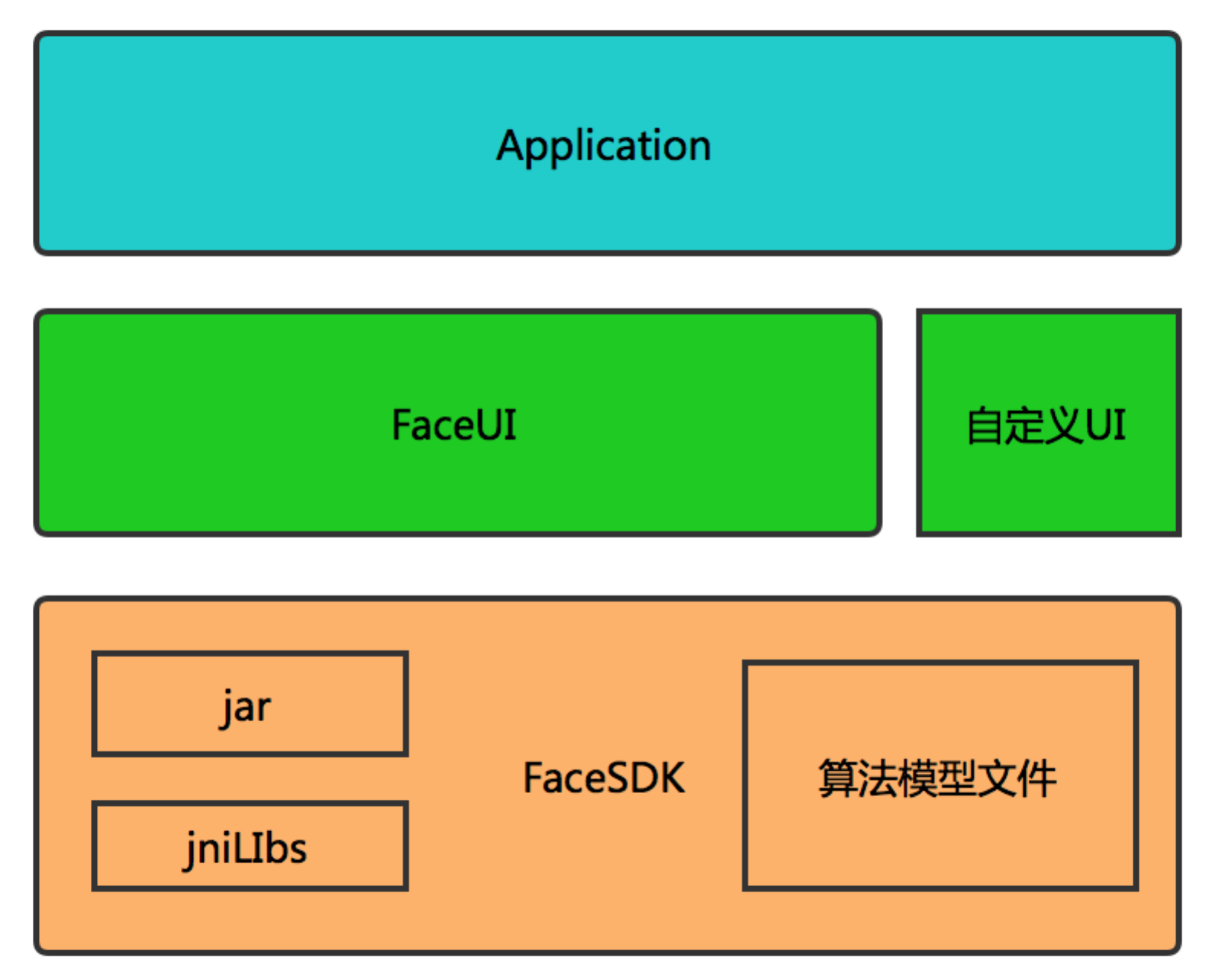
网络：支持 WIFI 及移动网络,移动网络支持使用 NET 网关及 WAP 网关(CMWAP、CTWAP、UNIWAP、3GWAP)。

2.3开发包说明

|  |  |
| --- | --- |
| 文件/文件夹名 | 说明 |
| /faceplatform-release | SDK lib 库相关代码的 aar。 |
| /faceplatform-ui | SDK的UI库，封装拍照裁剪等功能,以及各平台的so库。so包含以下几个平台如果关注包大小，请自行删减。armeabi/armeabi-v7a/arm64-v8a/x86， |
| 人脸SDK Android 版开发手册 | 本手册 |
| /FacePlatform/ | DEMO工程 |

3集成指南

本章将进行 Step-By-Step 的讲解,如何快速的集成 人脸Sdk到现有应用中。一个完整的 Demo 请参考开发包中的示例程序 FacePlatform。

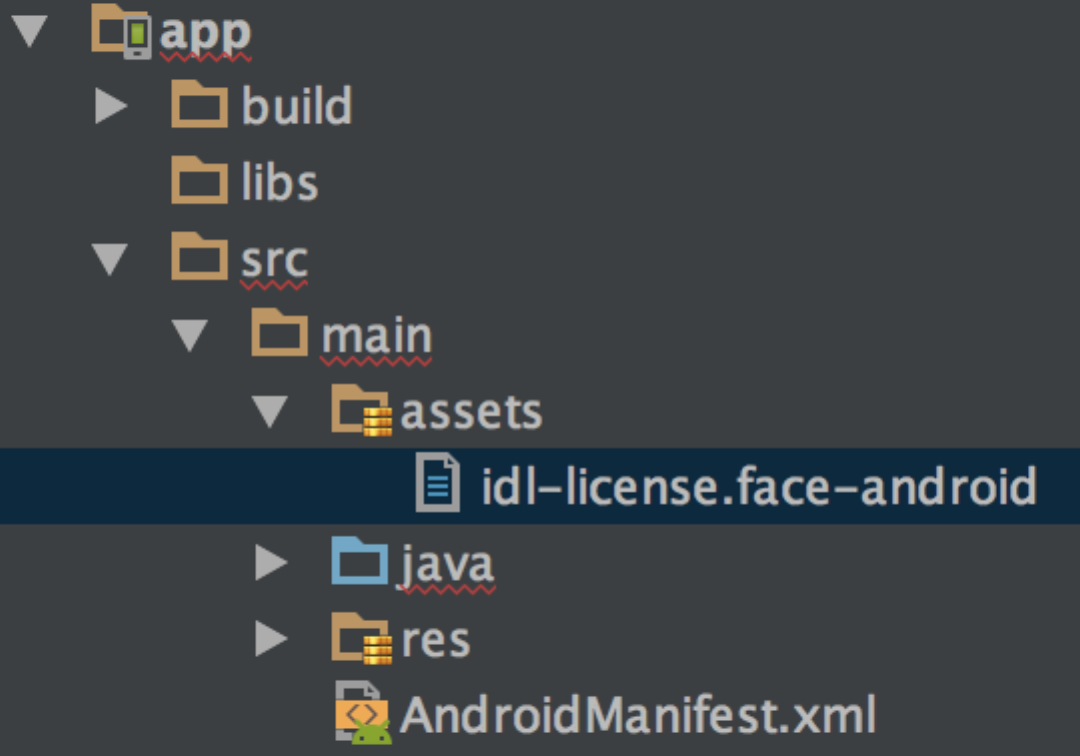


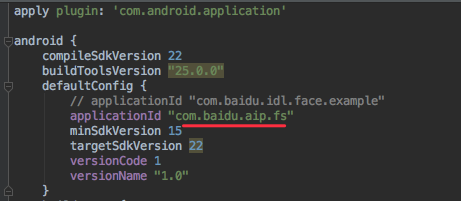
3.1 Sample示例

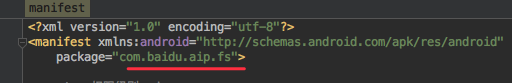


换新界面的图片

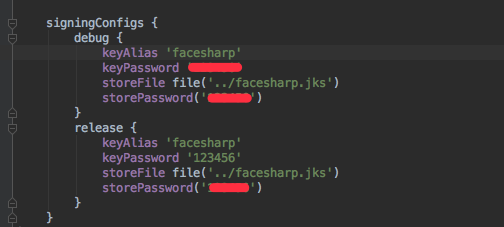
3.2体验Demo

1. 将demo通过Android studio打开；
2. 将申请SDK时提供的license添加到app->src->main->assets下。
3. 将app.gradle和AndroidManifest.xml里的包名改成您申请license的填入的包名;





1. 申请license时需要打包签名key的md5，SDK运行需要使用该签名，为了便于debug调试查看日志。请把打包签名key放到项目根目录，然后在app.gradle里面修改。



1. 修改SDK初始化参数，第二个参数APPID为：appname-face-android，其中LicenseID需要改成您申请SDK时填入的鉴权ID。同时请确保第三个参数“idl-license.face-android“与assets下的license文件名一致。license具体参见3.3



1. 运行Demo开始体验

3.3 license授权配置

1本SDK采有license离线授权方式，定期联网校验。拷贝在百度AI官网生成的license放到assets下。

注意：

1、AndroidManifest.xml 和 gradle里面的包名改成申请license时填入的包名；

2、SDK初始化传入第二个参数APPID = **LicenseID** + “-face-android”的拼接字符串， LicenseID为申请sdk时填写的应用名。 **LicenseID是应用名不是包名。**

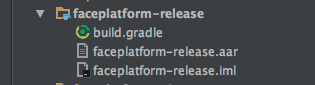
3、SDK初始化传入第三个参数"idl-license.face-android"为app->src->main->assets下的license文件名

4、打包用的key需要使用申请license时填入md5

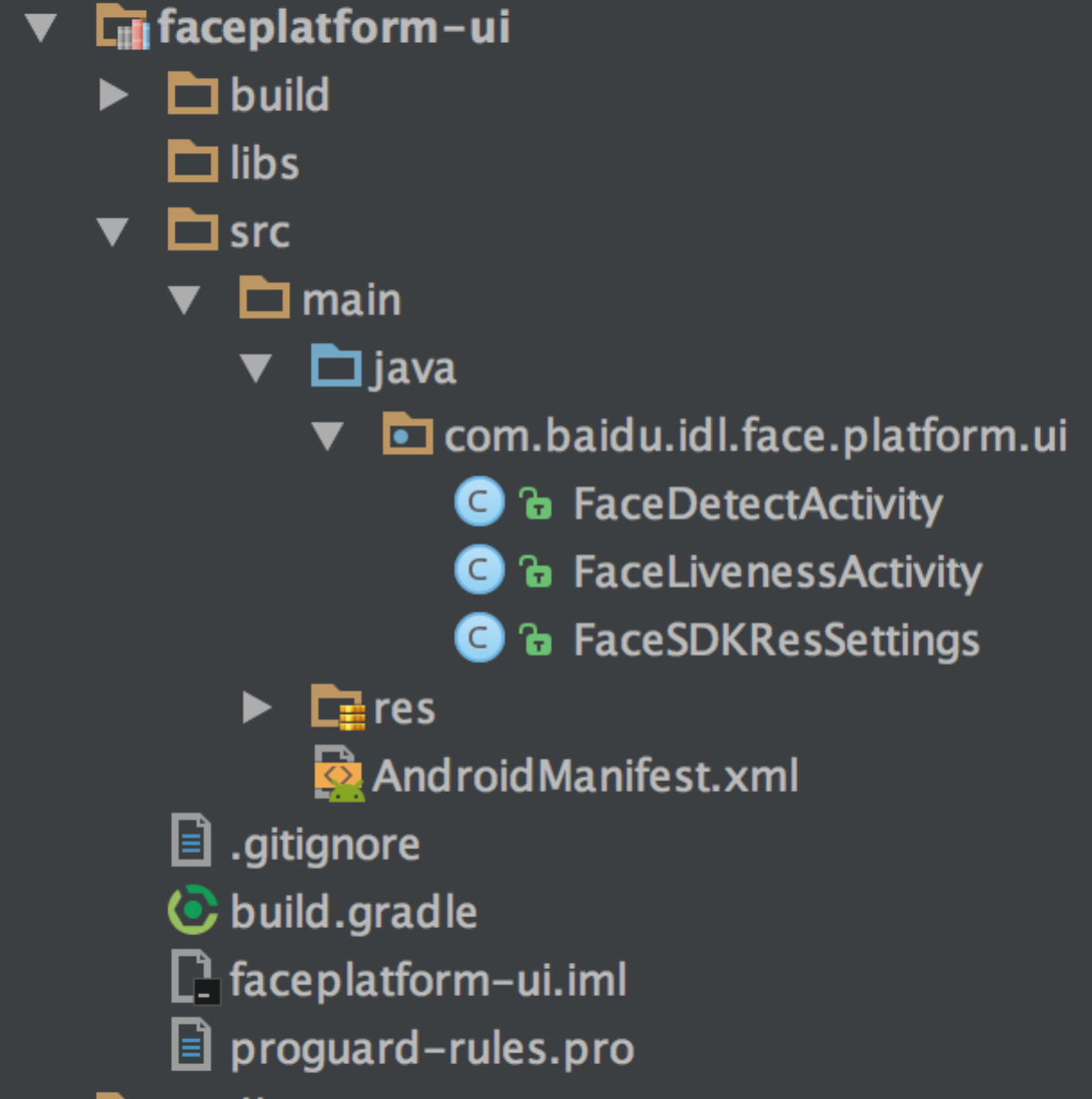
3.4添加FaceSdk到工程

FaceSdk以android studio开发方式提供，以下介绍在android studio开发工具导入FaceSdk

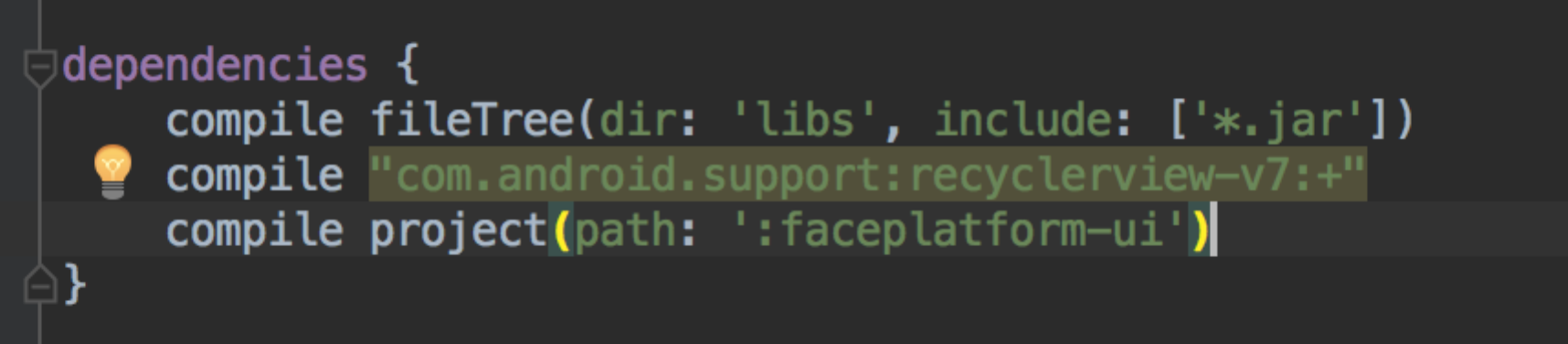
1. 将开发包中的faceplatform-release库 Copy 到工程根目录。



1. 将开发包中的faceplatform-ui库 Copy 到工程根目录。
2. SDK提供的了开源的faceplatform-ui库，把活体检测和人脸图像采集功能等功能进行了封装，适配了主流机型机型。如果需要使用，请添加faceplatform-ui模块到的工程中。faceplatform-ui目录结构如下图



1. 在build.gradle使用compile project引人faceplatform-ui库工程。



1. Setting.gradle中include faceplatfrom-ui和facepaltfrom-release



1. 从官网下载授权文件license，复制到 app/src/main/assets/license目录下。
2. 申请的license已经和打包签名key进行了绑定（申请时用到了签名的md5，为了便于debug模式也能调用SDK的功能，需要把debug的key改成申请license的key。
3. 把key拷贝到项目根目录下
4. 主app build.gradle android 下面添加(修改) signingConfigs相关的配置。如下图。



3.5权限声明

|  |  |
| --- | --- |
| 名称 | 用途 |
| android.permission.INTERNET | 允许应用联网,SDK联网授权。 |
| android.permission.READ\_PHONE\_STATE | 获取用户手机的 IMEI,用来唯一的标识用户 |
| android.permission.CAMERA | 允许调用相机进行拍照 |
| android.hardware.camera.autofocus | 允许相机对焦 |
| android.permission.WRITE\_EXTERNAL\_STORAGE | 图片裁剪临时存储 |

3.6 功能使用

3.6.1 活体识别

1. 调用FaceSDKManager.getIntance().initialize(context, LicenseID-face-android，“idl-license.face-android”);初始化SDK。Demo中此段代码在MainActivity中。
2. FaceEnvironment 相应的方法设置，提示音&提示语资源。(非必须)
3. 初始化相机。注册预览回调，开启预览。
4. 调用FaceSDKManager.getInstance().getLivenessStrategyModule() 获得ILivenessStrategy对象。(该方法每次调用都会返回一个新对象).
5. 调用 ILivenessStrategy.setPreviewDegree();设置预览图片的旋转角度。调用setDetectStrategySoundEnable设置是否开启语音。调用setDetectStrategyConfig设置，预览图的大小，人脸检测框的坐标和回调。
6. 多次调用livenessStrategy进行人脸图片采集，人脸跟踪。
7. 实现ILivenessStrategyCallback的onLivenessCompletion并处理结果。

3.6.2人脸采集

1. 调用FaceSDKManager.getIntance().initialize(context, appname-face-android, l idl-license.face-android”);初始化SDK.。
2. FaceEnvironment 相应的方法设置，提示音&提示语资源。(非必须)
3. 初始化相机。注册预览回调，开启预览。
4. 调用FaceSDKManager.getInstance().getDetectStrategyModule() 获得IDetectStrategy对象。(该方法每次调用都会返回一个新对象).
5. 调用 IDetectStrategy.setPreviewDegree();设置预览图片的旋转角度。调用setDetectStrategySoundEnable设置是否开启语音。调用setDetectStrategyConfig设置，预览图的大小，人脸检测框的坐标和回调。
6. 多次调用detectStrategy进行人脸图片采集。
7. 实现IDetectStrategyCallback的onDetectCompletion并处理结果。

3.6.3其他设置

com.baidu.idl.face.platform.FaceConfig 类用于人脸检测参数设置。

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 名称 | 默认值 | 取值范围 |
| brightnessValue | 图片爆光度 | 40f |  |
| blurnessValue | 图像模糊度 | 0.5f | 0~1.0f |
| occlusionValue | 人脸遮挡阀值 | 0.5f | 0~1.0f |
| headPitchValue | 低头抬头角度 | 10 | 0~45 |
| headYawValue | 左右角度 | 10 | 0~45 |
| headRollValue | 偏头角度 | 10 | 0~45 |
| cropFaceValue | 裁剪图片大小 | 400 |  |
| minFaceSize | 最小人脸检测值  小于此值的人脸将检测不出来。最小值为80. | 200 |  |
| notFaceValue |  | 0.6f | 0~1.0f |
| isCheckFaceQuality | 是否检测人脸质量 | true | true/flase |

3.7 界面定制说明

3.7.1 修改faceplatform\_ui界面

1、修改界面方式，替换布局文件或图片资源文件。

2、修改提示语音音频文件，有两种方式。

a、直接替换FaceUI工程raw下的mp3文件和string.xml。

b、FaceEnvironment 提供了setSoundId(FaceStatusEnum status, int soundId); 设置提示音资源。FaceStatusEnum为不同的状态。soundId为资源文件所对应的resource id. //TODO 都支持哪些音频格式？ 和setTipsId(FaceStatusEnum status, int tipsId); 设置提示语。

4.接口设计说明

4.1人脸功能管理器

人脸功能管理器FaceSDKManager，使用谷歌系统框架，函数与使用android自有框架基本相同。

4.1.1 创建实例

* 方法

FaceSDKManager getInstance()

* 参数

无

* 返回

人脸功能管理器

* 说明

创建人脸功能管理器。

* + 1. 人脸功能管理器初始化
* 方法

**public void** initialize(**final** Context context, String licenseID, String licenseFileName)

* 参数

context 上下文环境

licenseID 传入申请License时获取的应用名称+\_face\_android后缀

* 返回

无

* 说明

初始化人脸检测功能。进行人脸检测功能License鉴权验证。

* + 1. 设置人脸功能控制参数
* 方法

void setFaceConfig(FaceConfig config)

* 参数

config 人脸功能控制参数对象

* 返回

无

* 说明

设置人脸功能控制参数对象。

FaceConfig对象参数：

光照阀值

图像模糊阀值

人脸遮挡阀值

头部姿态角度

最小人脸检测阀值

人脸检测精度阀值

截取人脸图片大小

进行活体检测的动作类型列表

是否进行人脸图片质量检测

* + 1. 取得人脸图像采集功能接口
* 方法

IDetectStrategy getDetectStrategyModule()

* 参数

无

* 返回

人脸图像采集功能接口

* 说明

取得人脸图像采集功能接口。人脸图像采集接口完成，解析图片人脸信息，返回检测结果。

* + 1. 取得活体检测功能接口
* 方法

ILivenessStrategy getLivenessStrategyModule()

* 参数

无

* 返回

活体检测功能接口

* 说明

取得活体检测功能接口。活体检测功能接口完成，解析图片人脸信息，返回活体检测结果。

* 1. 人脸图像采集器

人脸图像采集器IDetectStrategy，检测图片中人脸信息，返回人脸检测状态，完成人脸图像采集。

4.2.1设置人脸图像采集功能参数

* 方法

void setDetectStrategyConfig(Rect previewRect, Rect detectRect, IDetectStrategyCallback callback);

* 参数

previewRect 人脸图片大小，类型：Rect

detectRect 人脸检测区域大小，类型：Rect

callback 人脸图像采集功能状态监听器

* 返回

无

* 说明

设置人脸功能控制参数对象。

4.2.2人脸图像采集

* 方法

void detectStrategy(byte[] imageData);

* 参数

imageData 图片信息

* 返回

无

* 说明

检测图片中的人脸信息，完成人脸图像采集，返回检测状态和结果。

* 1. 活体检测器

活体检测器ILivenessStrategy，检测图片人脸信息，活体检测结果状态。

* + 1. 设置人脸功能控制参数
* 方法

void setLivenessStrategyConfig(

List<LivenessTypeEnum> livenessList,

Rect previewRect,

Rect detectRect,

ILivenessStrategyCallback callback);

* 参数

livenessList 活体动作列表

previewRect 人脸图片大小，类型：Rect

detectRect 人脸检测区域大小，类型：Rect

callback 人脸图像采集功能状态监听器

* 返回

无

* 说明

设置活体检测功能参数对象。

* + 1. 活体检测
* 方法

void livenessStrategy(byte[] imageData);

* 参数

imageData 图片信息

* 返回

无

* 说明

检测图片中的人脸信息，完成人脸图像采集和人脸活体检测，返回检测状态和结果。

* 1. 人脸图像采集界面

人脸图像采集器界面FaceDetectActivity，包括UI界面，系统相机控制，使用人脸图像采集器IDetectStrategy处理相机采集到的图像。

* 1. 活体检测界面

活体检测界面FaceLivenessActivity，包括UI界面，系统相机控制，使用活体检测器ILivenessStrategy处理相机采集到的图像，完成活体检测功能。

1. 常见问题
2. **license文件有什么用，该放在什么地方？**

license文件需要申请，目的是作为sdk校验开发者的使用合法性，license文件放置位置 不对或未放置license文件会导致没法使用sdk，一般应先申请license文件，并把申请得到的license文件，放置在assets目录下面。

**2、appname没有替换成你自己的appname？**

如android版本，应该在初始化sdk时候，如： FaceSDKManager.getInstance().initialize(this, "appname-face-android”); 其中appname为你申请的应用名称.

**3、license文件失效了，不能用了怎么办？**

icense文件申请时候有期限，如过期会导致校验失效，需要上官网申请延期。

1. **活体检测常见有那些动作？是否可配置？**

常见有6个动作，眨眼，张大嘴，向上抬头，向下低头，向左摇头，向右摇头等。 sdk提供传入设置参数，如活体检测角度、光线，检测动作，检测动作数量等设置。

1. **使用sdk一般会用到活体检测拍照等功能，有什么需要注意？**

android6.0+需要注意相机拍摄权限问题。如没申请权限，可能导致没法调起相机。

非用户功能，开发可以了解下

1.  
***Detect\_DataHitOne***,  
***Detect\_DataHitLast***,

加入datahitone,datahitlast状态，sdk检测人脸方法返回这两个状态值时，可以保存当前图片帧作为bestimage未剪裁图片信息。

2. *FaceSDK.setPerfLogFlag(0);  
 FaceSDK.setValueLogFlag(0);*

底层log输出开关，perlog耗时log输出，valuelog 检测数据log输出

3.SDK不再提供arm架构的so，只提供arm-v7,arm-64,x86架构平台的so