

数据宝活体检测人像比对产品对接说明

(IOS)

2018 年 4 月 23 日
数据宝技术部

目录

1. 活体检测人像比对 SDK 介绍	4
1.1. SDK 详细内容	4
1.2. 活体检测人像比对流程介绍	4
1.3. Sample 示例	5
2. 开发集成指南	7
2.1. 引入 SDK 包，最低支持 iOS 8.0	7
2.2. 编译选项设置	8
2.3. 开发指南	9
2.3.1. 初始化和可用产品检查	9
2.3.2. OCR+活体检测+实名认证+人像比对产品	9
2.3.3. 实名认证+活体检测+人像比对产品	10
2.3.4. 结束服务	11
3. 函数库说明	11
3.1. CDPAuth	11
3.2. CDPHandler	14
3.3. CDPInfoHandler	14
3.4. CDPSTDeviceErrorHandler	15

3.5.	CDPSTCancelHandler	15
3.6.	CDPSTSuccessHandler	15
3.7.	CDPSTFailedHandler.....	15
3.8.	STLivenessEnumType	15

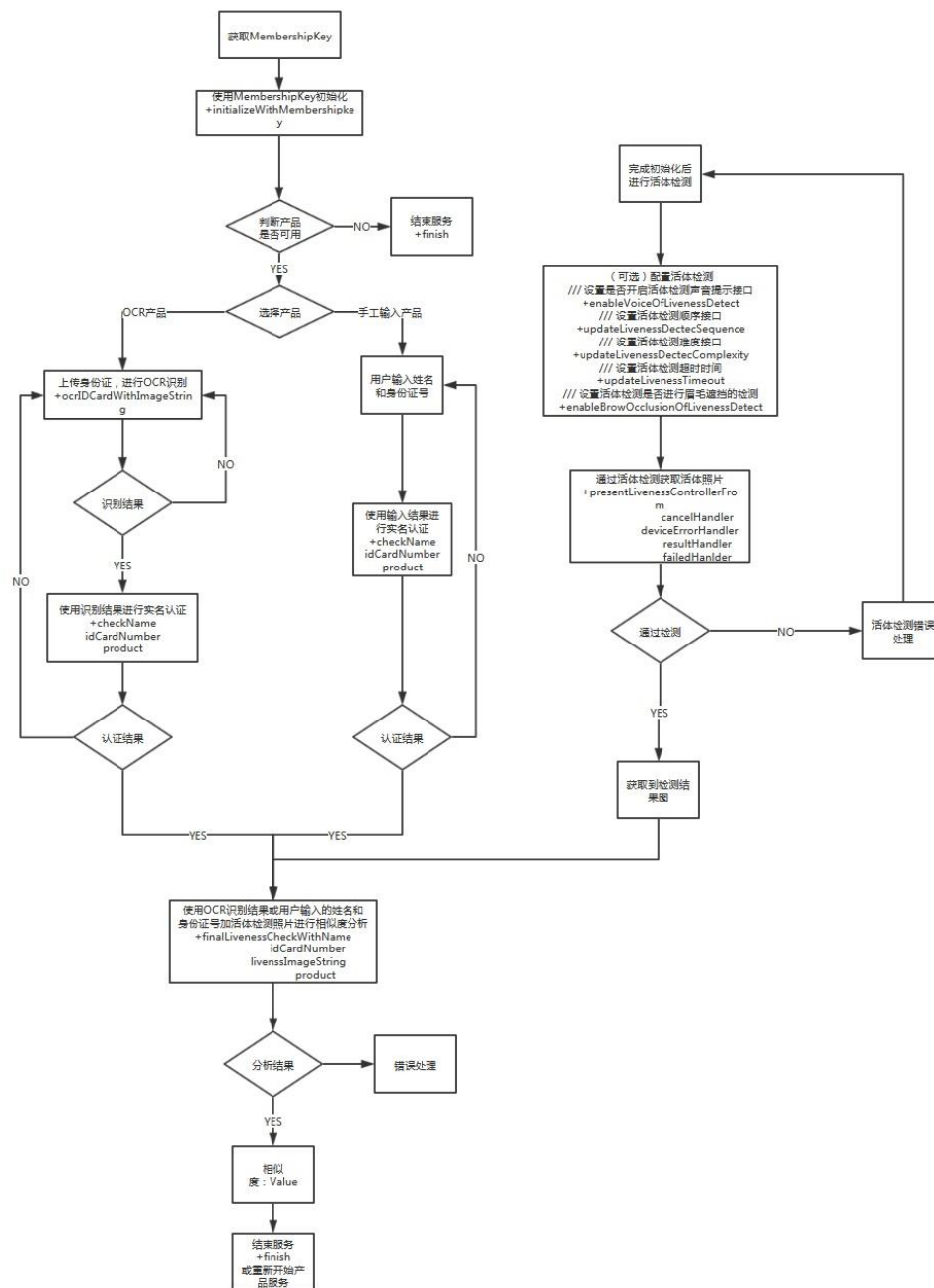
1. 活体检测人像比对 SDK 介绍

1.1.SDK 详细内容

SDK 主要包括四个方面内容，如下图所示



1.2.活体检测人像比对流程介绍



1.3.Sample 示例

- 第一步初始化鉴权和产品页面和实名认证输入

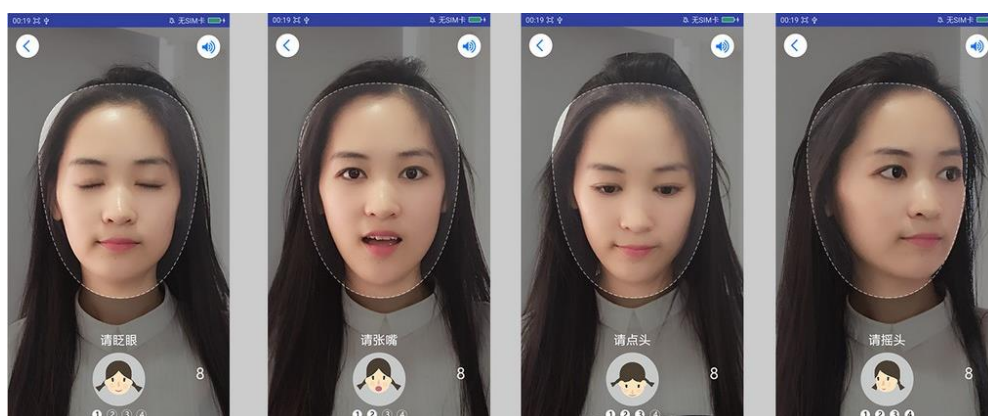


➤ 第二步活体检测

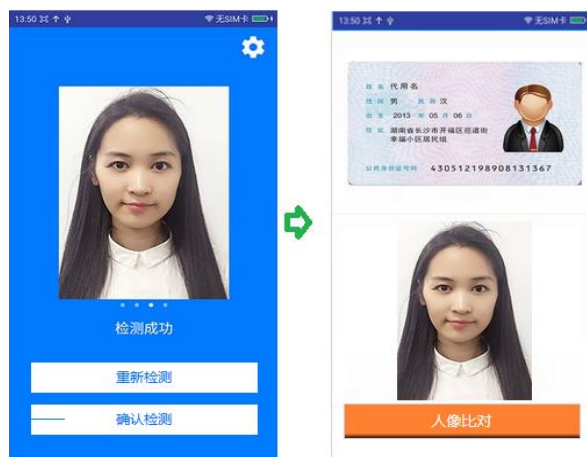
A. 人脸对正



B. 根据提示做出相应动作，动作完成后，自动进行活体检测输出结果



C. 确认检测后进入对比

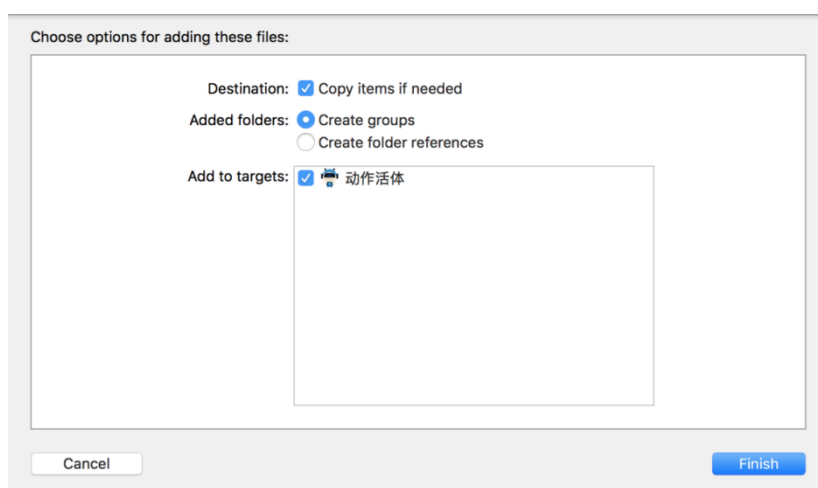


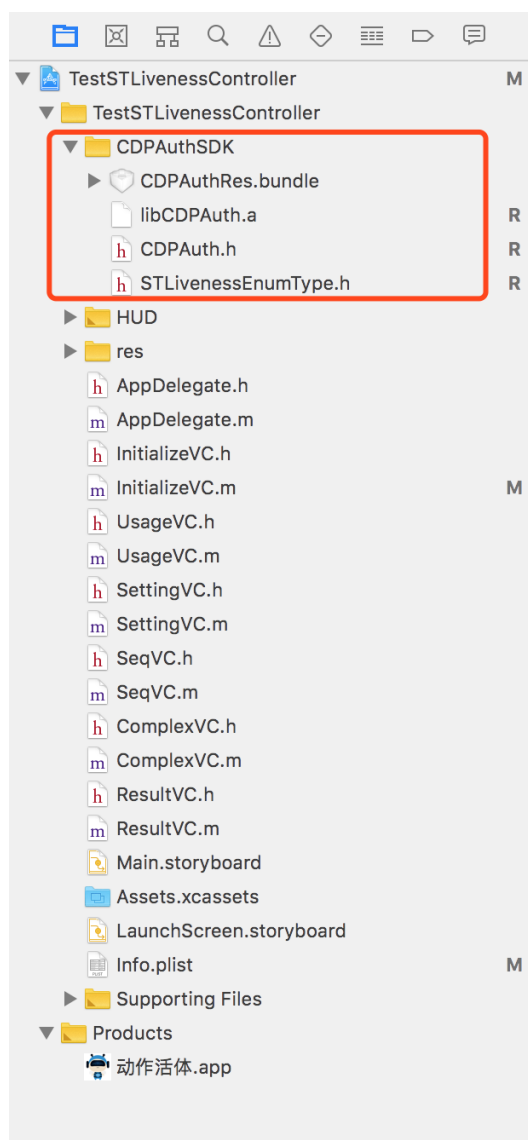
➤ 第三步人像比对



2. 开发集成指南

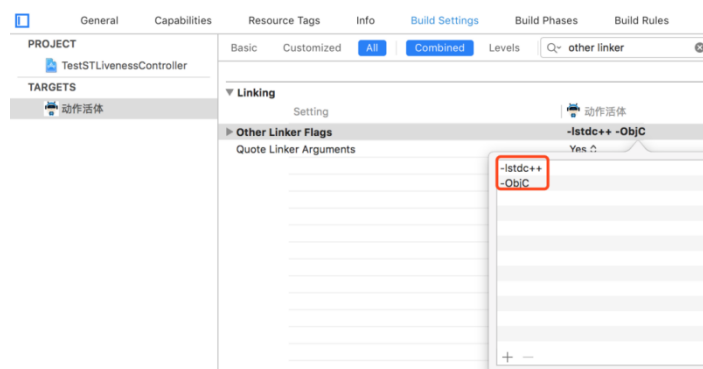
2.1. 引入 SDK 包，最低支持 iOS 8.0



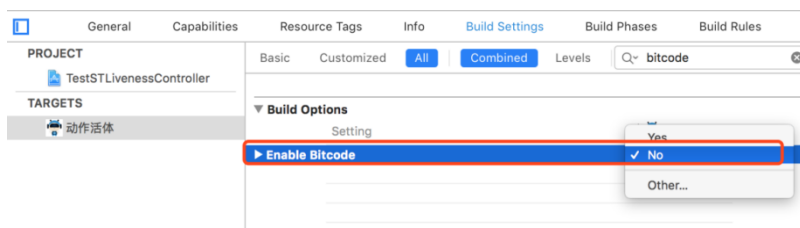


2.2.编译选项设置

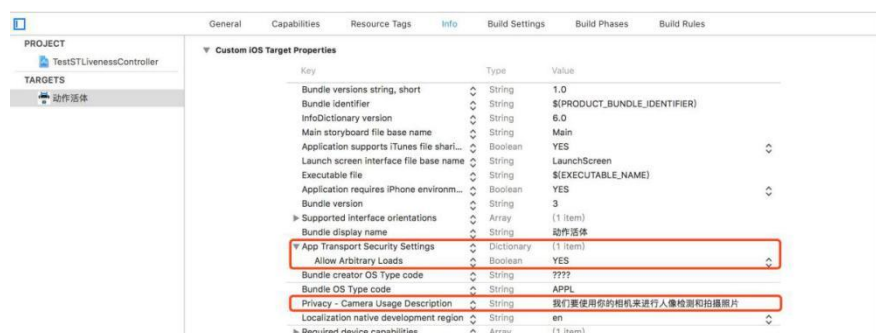
- 需要添加 Xcode 链接器参数：-ObjC 和 -lstdc++



- 需要手动关闭 Bitcode



- 在 info.plist 文件中加入 ATS 说明，相机权限请求说明



2.3.开发指南

2.3.1. 初始化和可用产品检查

```
// 在需要的页面引入头文件
#import "CDPAuth.h"
// 在服务开始前执行初始化
[CDPAuth initializeWithMembershipkey:#KEY handler:^(BOOL success, NSError * _Nullable error) {
    if (success) {
        // 处理成功结果
        if ([CDPAuth isProduct1Available]) {
            // OCR 产品可用
        }
        if ([CDPAuth isProduct2Available]) {
            // 输入姓名和身份证号产品可用
        }
    }
    else {
        // 处理失败结果，错误信息在 error.userInfo[NSLocalizedStringDescriptionKey]中
        // 两个产品都不可用时也会报错处理
    }
}];
```

2.3.2. OCR+活体检测+实名认证+人像比对产品

2.3.2.1. 将身份证照片上传到服务器获取身份证号和姓名

```
NSData *data = UIImageJPEGRepresentation(#IMAGE, 0.5);
NSString *imageString = [data.base64Encoding stringByAddingPercentEscapesUsingEncoding:NSUTF8StringEncoding];
[CDPAuth ocrIDCardWithImageString:imageString handler:^(BOOL success, NSError *error, NSDictionary *info) {
    if (success) {
        // 识别成功可以获取到未经认证的姓名和身份证号
        NSString *name = info[@"name"];
        NSString *number = info[@"idcard"];
    }
    else {
        // 识别失败获取错误原因，错误信息在 error.userInfo[NSLocalizedStringDescriptionKey]中
    }
}];
```

对于摄像头的定制详见 demo 源码，图片的处理在 TestSTLivenessController 中 UsageVC.m

2.3.2.2. 将姓名和身份证号上传到服务器进行实名认证

```
//参数 product:根据产品线区分, 传 1 或 2, 此处传 1
[CDPAuth checkName:#NAME idCardNumber:#NUMBER product:1 handler:^(BOOL success, NSError *error) {
    if (success) {
        // 实名认证通过
    } else {
        // 实名认证未通过, 错误信息在 error.userInfo[NSLocalizedStringKey]中
    }
}];
```

2.3.2.3. 使用活体检测功能获取活体照片

```
[CDPAuth presentLivenessControllerFrom:#VIEWCONTROLLER cancelHandler:^(
    //用户取消
) deviceErrorHandler:^(STIDDeviceError errorCode) {
    // 设备出错
    switch (errorCode) {
        case STID_E_CAMERA:
            //相机权限获取失败:请在设置-隐私-相机中开启后重试
            break;
        case STID_WILL_RESIGN_ACTIVE:
            //活体检测已经取消"
            break;
    }
} resultHandler:^(NSArray<UIImage *> *images) {
    // 识别成功, 获取到识别期间的几张图像
} failedHandler:^(LivefaceErrorType errorType) {
    // 识别失败
    switch (errorType) {
        case STID_E_LICENSE_INVALID:
            //未通过授权验证
            break;
        .....
    }
}];
```

2.3.2.4. 将姓名, 身份证号和活体检测图片上传获取相似度结果

```
NSData *data = UIImageJPEGRepresentation(#LIVENESSIMAGE, 1.0);
NSString *imageString = [data.base64Encoding stringByAddingPercentEscapesUsingEncoding:NSUTF8StringEncoding];
//参数 product:根据产品线区分, 传 1 或 2, 此处传 1
[CDPAuth finalLivenessCheckWithName:#NAME idCardNumber:#NUMBER livenessImageString:imageString product:1 handler:^(BOOL success,
NSError *error, NSDictionary *info) {
    if (success) {
        // 识别成功, 获取到相似度评分
        float value = [info[@"score"] floatValue];
    } else {
        // 识别失败, 错误信息在 error.userInfo[NSLocalizedStringKey]中
    }
}];
```

2.3.3. 实名认证+活体检测+人像比对产品

2.3.3.1. 输入姓名和身份证号进行实名认证

```
//参数 product:根据产品线区分, 传 1 或 2, 此处传 2, 其他同 2.3.2.2
```

2.3.3.2. 使用活体检测功能获取活体照片

```
//同 2.3.2.3
```

2.3.3.3. 将姓名, 身份证号和活体检测图片上传获取相似度结果

//参数 product:根据产品线路区分, 传 1 或 2, 此处传 2, 其他同 2.3.2.4

2.3.4. 结束服务

```
// 调用该方法结束服务
[CDPAuth finish];
```

3. 函数库说明

3.1.CDPAuth

回调处理说明

统一错误说明: NSError *error

错误分类	说明
系统错误	iOS系统返回的错误, 例如网络请求出错, 错误信息: error.userInfo[NSLocalizedStringDescriptionKey]由系统返回, 错误码: error.code由系统返回
方法调用错误	例如仅获取到ProductOne, 但在调用函数时, product参数传入了2。错误信息: error.userInfo[NSLocalizedStringDescriptionKey]为SDK错误提示, 错误码: error.userInfo[@"code"] 统一为@"0"
服务器返回错误	服务器方返回的错误。错误信息: error.userInfo[NSLocalizedStringDescriptionKey]由服务器返回, 错误码: error.userInfo[@"code"]由服务器返回

初始化

```
+ (void)initializeWithMembershipKey:(NSString *)membershipKey
    handler:(nullable CDPHandler)handler;
```

参数	说明
membershipKey	认证序列号, 请联系客服获取
handler	结果处理结果Block
返回结果	无

产品可用性

```
+ (BOOL)isProduct1Available;
```

参数	说明
返回结果	是否可用OCR产品

```
+ (BOOL)isProduct2Available;
```

参数	说明
返回结果	是否可用输入文本经行实名认证产品

实名认证接口

```
+ (void)checkName:(nonnull NSString *)name
    idCardNumber:(nonnull NSString *)cardNumber
    product:(int)product
    handler:(nullable CDPHandler)handler;
```

参数	说明
name	姓名
cardNumber	身份证号
product	产品线，1：OCR方案或2：文字方案，不填直接返回失败处理
handler	结果处理结果Block
返回结果	无

OCR识别接口

```
+ (void)ocrIDCardWithImageString:(nonnull NSString *)imageString
    handler:(nullable CDPInfoHandler)handler;
```

参数	说明
imageString	身份证转码字符串，参考 NSString *imageString = [UIImageJPEGRepresentation(image, 1.0).base64Encoding stringByAddingPercentEscapesUsingEncoding:NSUTF8StringEncoding]
handler	结果处理结果Block，识别结果在回调的字典中
返回结果	无

比对服务

```
+ (void)finalLivenessCheckWithName:(nonnull NSString *)name
    idCardNumber:(nonnull NSString *)cardNumber
    livenssImageString:(nonnull NSString *)livenssImageString
    product:(int)product
    handler:(nullable CDPInfoHandler)handler;
```

参数	说明
name	姓名
cardNumber	身份号
livenssImageString	活体识别转码字符串，参考 NSString *livenssImageString = [UIImageJPEGRepresentation(image, 1.0).base64Encoding stringByAddingPercentEscapesUsingEncoding:NSUTF8StringEncoding]
product	产品线，1：OCR方案或2：文字方案，不填直接返回失败处理
handler	结果处理结果Block，识别结果在回调的字典中
返回结果	无

活体识别配置项

+ (BOOL)isLivenessVoiceEnabled;

参数	说明
返回结果	是否开启声音提示

+ (BOOL)enableVoiceOfLivenessDetect:(BOOL)isVoiceEnabled;

参数	说明
isVoiceEnabled	配置是否开启声音提示，默认开启
返回结果	是否配置成功

+ (nonnull NSArray *)livenessDetectSeq;

参数	说明
返回结果	当前活体检测顺序接口

+ (BOOL)updateLivenessDetectSequence:(nonnull NSArray *)sequence;

参数	说明
sequence	配置顺序，默认为以下参考 NSArray *sequence = @[@(LIVEBLINK), @(LIVEMOUTH), @(LIVENOD), @(LIVEYAW)]
返回结果	是否配置成功

+ (LivefaceComplexity)livenessDetectComplexity;

参数	说明
返回结果	当前活体检测难度

+ (BOOL)updateLivenessDetectComplexity:(LivefaceComplexity)Complexity;

参数	说明
Complexity	活体检测难度，默认为LIVECOMPLEXITYNORMAL
返回结果	是否配置成功

+ (float)livenessTimeout;

参数	说明
返回结果	当前活体检测时间限制

+ (BOOL)updateLivenessTimeout:(float)timeout;

参数	说明
timeout	活体检测超时时间，默认为10秒
返回结果	是否配置成功

```
+ (BOOL)isLivenessBrowOcclusionEnabled;
```

参数	说明
返回结果	是否进行眉毛遮挡的检测

```
+ (BOOL)enableBrowOcclusionOfLivenessDetect:(BOOL)isBrowOcclusionEnabled;
```

参数	说明
isBrowOcclusionEnabled	是否进行眉毛遮挡的检测，默认为不检测
返回结果	是否配置成功

活体识别接口

```
+ (void)presentLivenessControllerFrom:(nonnull UIViewController *)viewController
cancelHandler:(nullable CDPSTCancelHandler)cancelHandler
deviceErrorHandler:(nullable CDPSTDeviceErrorHandler)deviceErrorHandler
resultHandler:(nullable CDPSTSuccessHandler)successHandler
failedHandler:(nullable CDPSTFailedHandler)failedHandler;
```

参数	说明
viewController	将从这个viewController唤起活体识别界面
cancelHandler	用户取消处理
deviceErrorHandler	设备错误处理，返回一个STIdDeviceError错误码，无相机权限或设备不再活跃
successHandler	检测成功结果处理，返回一组UIImage照片
failedHandler	检测失败结果处理，返回一个LivefaceErrorType错误码，错误原因参照错误代码与对照表
返回结果	无

终结服务

```
+ (void)finish;
```

参数	说明
返回结果	无

3.2.CDPHandler

通用简单回调 `typedef void (^CDPHandler)(BOOL success, NSError * _Nullable error);`

参数	说明
success	是否成功
error	如果不成功，产生错误，详见统一错误说明
返回结果	无

3.3.CDPInfoHandler

通用信息回调 `typedef void (^CDPInfoHandler)(BOOL success, NSError * _Nullable error, NSDictionary * _Nullable info);`

参数	说明
success	是否成功
info	返回的信息，根据实际情况决定是否使用
error	如果不成功，产生错误，详见统一错误说明
返回结果	无

3.4.CDPSTDeviceErrorHandler

活体检测设备回调 `typedef void (^CDPSTDeviceErrorHandler)(STIdDeveiceError errorCode);`

参数	说明
errorCode	错误码，详见STIdDeveiceError
返回结果	无

3.5.CDPSTCancelHandler

活体检测取消回调 `typedef void (^CDPSTCancelHandler)(void);`

参数	说明
返回结果	无

3.6.CDPSTSuccessHandler

活体检测成功回调 `typedef void (^CDPSTSuccessHandler)(NSArray<UIImage *> * _Nonnull images);`

参数	说明
images	图像合集，可任意使用
返回结果	无

3.7.CDPSTFailedHandler

CDPSTFailedHandler

活体检测失败回调 `typedef void (^CDPSTFailedHandler)(LivefaceErrorType errorType);`

参数	说明
errorType	失败错误码，详见LivefaceErrorType
返回结果	无

3.8.STLivenessEnumType

STLivenessEnumType类说明了活体检测所用到的枚举类型。

LivefaceErrorType

活体检测失败类型。

类型	说明
STID_E_LICENSE_INVALID	License文件不合法(SenseID_Liveness_Interactive.lic)
STID_E_LICENSE_FILE_NOT_FOUND	License文件未找到(SenseID_Liveness_Interactive.lic)
STID_E_LICENSE_BUNDLE_ID_INVALID	License绑定包名错误
STID_E_LICENSE_EXPIRE	License文件过期
STID_E_LICENSE_VERSION_MISMATCH	License与SDK版本不匹配
STID_E_LICENSE_PLATFORM_NOT_SUPPORTED	License不支持当前平台
STID_E_MODEL_INVALID	Model文件不合法
STID_E_MODEL_FILE_NOT_FOUND	Model文件未找到
STID_E_MODEL_EXPIRE	模型文件过期
STID_E_NOFACE_DETECTED	没有人脸
STID_FACE_OCCLUSION	人脸遮挡
STID_E_TIMEOUT	检测超时
STID_E_INVALID_ARGUMENTS	参数设置不合法
STID_E_CALL_API_IN_WRONG_STATE	调用API状态错误
STID_E_API_KEY_INVALID	API_KEY或API_SECRET错误
STID_E_SERVER_ACCESS	服务器访问错误
STID_E_SERVER_TIMEOUT	服务器访问超时
STID_E_HACK	活体检测失败

STidDeveiceError

设备错误的类型。

类型	说明
STID_E_CAMERA	相机权限获取失败
STID_WILL_RESIGN_ACTIVE	应用即将被挂起

LivenesssTrackerFaceBoundStatus

活体对准中人脸的位置。

类型	说明
STID_BOUND_NO_FACE	没有人脸
STID_FACE_IN_BOUNDE	人脸在框内
STID_BOUND_FACE_OUT_BOUND	人脸出框

LivenesssTrackerFaceDistanceStatus

活体对准中人脸远近。

类型	说明
STID_FACE_TOO_FAR	人脸距离手机过远
STID_FACE_TOO_CLOSE	人脸距离手机过近
STID_DISTANCE_FACE_NORMAL	人脸距离正常
STID_DISTANCE_UNKNOWN	人脸距离未知

LivefaceDetectionType

检测模块类型。枚举类型定义了眨眼、点头、张嘴、摇头四种。

类型	说明
LIVE_BLINK	眨眼检测
LIVE_NOD	上下点头检测
LIVE_MOUTH	张嘴检测
LIVE_YAW	左右转头检测

LivefaceOrientaion

人脸方向。枚举类型定义了上、下、左、右四个方向。

类型	说明
LIVE_FACE_UP	人脸向上，即人脸朝向正常
LIVE_FACE_LEFT	人脸向左，即人脸被逆时针旋转了90度
LIVE_FACE_DOWN	人脸向下，即人脸被逆时针旋转了180度
LIVE_FACE_RIGHT	人脸向右，即人脸被逆时针旋转了270度

LivefaceComplexity

活体检测复杂度。

类型	说明
LIVE_COMPLEXITY_EASY	简单, 人脸变更时不会回调 LIVENESS_FACE_CHANGED 错误, 活体阈值低
LIVE_COMPLEXITY_NORMAL	正常, 人脸变更时会回调 LIVENESS_FACE_CHANGED 错误, 活体阈值较低
LIVE_COMPLEXITY_HARD	困难, 人脸变更时会回调 LIVENESS_FACE_CHANGED 错误, 活体阈值较高
LIVE_COMPLEXITY_HELL	地狱, 人脸变更时会回调 LIVENESS_FACE_CHANGED 错误, 活体阈值最高