



## 小程序接口文档

`wx.startFacialRecognitionVerifyAndUploadVideo(OBJECT)`

请求进行基于生物识别的人脸核身，  
同时上传认证过程视频

微信开放平台

(-内部文档 禁止外传-)

# 目录

一、小程序接口使用说明.....	3
1.1 背景简介.....	3
1.2 接口说明.....	3
二、小程序接口描述.....	4
2.1 接口描述.....	4
2.2 参数说明.....	4
三、上传用户姓名身份证的后台 API.....	8
3.1 API 说明.....	8
3.2 请求数据格式.....	8
3.3 返回数据.....	9
四、再次获取人脸核身结果后台 API.....	9
4.1 API 说明.....	9
4.2 请求数据说明.....	10
4.3 返回参数说明.....	11
五、小程序辅助接口：检查设备是否支持人脸检测.....	12
5.1 接口名称.....	12
5.2 接口说明和使用.....	12
5.3 接口说明和使用.....	13
六、接口申请.....	14
6.1 联系腾讯开通权限.....	14
6.2 申请方式.....	15

# 一、小程序接口使用说明

## 1.1 背景简介

互联网+国家战略发布以来，各行业紧锣密鼓推进改革，尤其是政务民生类业务。2015年至2016年，国家公安部拟定了“居民身份证网络应用国家标准”，从政策层面推进标准执行。这为需要“身份证件”参与办理的线下业务，提供了延展至线上办理的政策法律层面、技术标准层面、公安认可层面等社会层面的支持。

2016年先后，国家在医疗挂号、App注册、快递收寄、票务等多领域规定，需要用户实名才可办理业务，预计国家后续也会有越来越多的此类法规。基于此，微信参照公安部“互联网+”可信身份认证服务平台标准，依托腾讯公司及微信的生物识别技术，建立微信“实名实人信息校验能力”，即通过人脸识别+公安比对，校验用户实名信息和本人操作。拟提供给资质符合要求的业务方，在合适的业务场景内使用。

## 1.2 接口说明

1、目前此人脸核身的接口接入处于邀请内测阶段，需邮件申请开通权限，开通流程请见本文档最后部分

2、本人脸核验接口中，用户交互方式分为两种，一种是正对屏幕框住位置，朗读手机屏幕展示的数字（简称“读数字”），另一种是正对屏幕限定的光圈，静止并等待屏幕闪烁（简称“屏幕闪烁”）。这两种交互方式可二选一，也可由系统自动根据当前情况，选择适宜的交互。

3、接口中读数字与屏幕闪烁的交互方式，支持的微信版本不同，请留意。

4、本人脸核验接口，会上传认证过程的视频，仅提供给需要使用的场景，请留意。

## 二、小程序接口描述

### 2.1 接口描述

名称: wx.startFacialRecognitionVerifyAndUploadVideo(OBJECT)

功能: 请求进行基于生物识别的人脸核身, 同时上传认证过程视频

验证方式: 在线验证

兼容版本:

读数字: android 微信 6.5.4 及以上版本, iOS 微信 6.5.6 及以上版本

屏幕闪烁: android 微信 6.7.2 及以上版本, iOS 微信 6.7.2 及以上版本

开放状态: 未开放

开放范围: 白名单

### 2.2 参数说明

#### 1、OBJECT 参数说明:

参数	类型	必填	说明
name	String	是	姓名
idCardNumber	String	是	身份证号码
success	Function	否	调用成功回调
fail	Function	否	调用失败回调
complete	Function	是	调用完成回调 (成功或失败都会回调)
checkAliveType	Number	否	人脸核验的交互方式, 默认读数字 (见表 1)

表 1: checkAliveType 的值和对应的解释:

参数	解释
0	读数字（默认）
1	屏幕闪烁
2	先检查是否可以屏幕闪烁，不可以则自动为读数字

## 2、CALLBACK 返回参数

参数	类型	说明
errMsg	String	错误信息
errCode	Number	错误码
verifyResult	String	本次认证结果凭据，第三方可以选择根据这个凭据获取相关信息

### 注 1：传递用户姓名和身份证有两种方式

1. 业务方没有用户实名信息，用户需要在前端填写身份证和姓名，那么前端直接通过 jsapi 调用传递 name 和 idCardNumber。
2. 业务方已经有用户实名信息，后台通过微信提供的 api（详情见文档后面“上传姓名身份证后台 api”）上传用户身份证姓名和身份证，api 返回 user\_id\_key 作为凭证传给前端，前端再调用 jsapi，用户姓名、身份证信息不需要经过前端，参数只需要传递 userIdKey。 **Tips：使用该功能需要小程序基础库版本号 >= 1.9.3。**

## 3、回调结果说明

错误码	解释
0	识别成功
10001	参数错误
10002	人脸特征检测失败
10003	身份证号不匹配

10004	公安比对人脸信息不匹配
10005	正在检测中
10006	appid 没有权限（后台验证部分）
10007	后台获取图片失败
10008	公安系统失败
10009	公安未查到身份证照片比对源
10010	照片质量不满足公安比对要求
10011	身份证信息未开通公安比对权限
10012	征信验证失败
10013	征信系统错误
10014	公安系统失败，征信系统成功
10015	公安服务暂时不可用
10016	存储用户图片失败
10017	非法 identify_id
10018	用户信息不存在
10020	认证超时
10021	重复的请求，返回上一次的结果
10022	用户信息错误，请检测 json 格式
10026	用户身份证数据不在公安比对数据库中
10027	语音识别失败
10028	唇动检测失败
10029	微警超时
10030	绑定身份证失败
10031	没有申请征信商户号或公众号/小程序没有设置昵称
10032	用户身份证数据不再征信数据库中
10040	请求数据编码不对，必须是 UTF8 编码
10041	非法 user_id_key
10042	请求过于频繁，稍后再重试
10045	系统失败
10052	请求数超时征信的限制
10057	检测过程中人脸主体不一致
10066	人脸流水号 bioid 重复
10069	活体检测暂时不可用
90100	用户取消
90101	用户未授权
90102	底层库出错

90103	CDN 上传出错
90104	获取配置信息出错
90105	获取确认页信息失败
90106	相机初始化失败
90107	用户采集人脸超时
90108	用户采集过程中都动态剧烈
90109	设备不支持人脸采集
90199	未知错误

#### 4、示例代码

```
wx.startFacialRecognitionVerify({
  name: '张三',
  idCardNumber: '320311123456789000',
  success(){
    //识别成功
  },
  fail(res) {
    //识别失败
    const verifyResult = res.verifyResult
  }
})
```

```
wx.startFacialRecognitionVerify({
  userIdKey: '52bdcf0b05439b1731e1e8174f42a9b3',
  success(){
    //识别成功
  },
  fail(res) {
    //识别失败
    const verifyResult = res.verifyResult
  }
})
```

请根据调用接口的情况，代码书写对应的 JSAPI 接口。

## 三、上传用户姓名身份证的后台 api

### 3.1 API 说明

#### 1、说明

业务方上传用户姓名和身份证，获取用户凭证，把凭证给到前端通过 jsapi 调用。

**Tips：使用该功能需要小程序基础库版本号 >= 1.9.3。**

#### 2、请求 URL

https://api.weixin.qq.com/cityservice/face/identify/getuseridkey?access\_token={access\_token}

#### 3、请求方式

POST

### 3.2 请求数据格式

Json

```
{
  "name": "张三",
  "id_card_number": "452122xxxxxx43215"
}
```

#### 请求示例：

```
#!/bin/bash
```

```
TOKEN='xxxxxxxxxxxx'
```



URL='https://api.weixin.qq.com/cityservice/face/identify/getuseridkey'

JSON='{ "name": "张三", "id\_card\_number": "452344xxxxxxxxxxxx234"}'

curl "\${URL}?access\_token=\${TOKEN}" -d "\${JSON}"

#### 参数说明:

json 字段	中文显示
name	姓名
id_card_number	身份证号

### 3.3 返回数据

参数	类型	说明
errcode	int	错误码
errmsg	string	错误信息
user_id_key	string	用于后台交互表示用户姓名、身份证的凭证
expires_in	uint32	user_id_key 有效期, 过期需重新获取

```
{
  "errcode": 0,
  "errmsg": "ok",
  "user_id_key": "id_key_xxxx",
  "expires_in": 3600
}
```

## 四、再次获取人脸核身结果后台 API

### 4.1 API 说明

#### 1、说明

人脸核身之后开发者可以根据jsapi 返回的 verify\_result 向后台拉取当次认证的结果信息。

## 2、请求 URL

`https://api.weixin.qq.com/cityservice/face/identify/getinfo?access_token={access_token}`

## 3、请求方式

POST

## 4、数据格式

Json

# 4.2 请求数据说明

## 1、请求

参数	类型	是否必填	描述
verify_result	string	是	jsapi 返回的加密 key

## 2、数据返回 HTTP 头如下

```
Date: Mon, 06 Feb 2017 08:12:58 GMT
Content-Type: application/json; encoding=utf-8
Content-Length: 85
Connection: close
```

### json 示例

```
{
  "errcode" : 0,
```

```
"errmsg" : "ok",  
"identify_ret" : 0,  
"identify_time" : 1486350357  
"validate_data": "8593"  
}
```

#### 请求示例

```
#!/bin/bash  
  
TOKEN='xxxxxxxxxxxx'  
URL='https://api.weixin.qq.com/cityservice/face/identify/getinfo'  
JSON='{ "verify_result": "xxxxxxxxxxxx" }'  
  
curl "${URL}?access_token=${TOKEN}" -d "${JSON}"
```

## 4.3 返回参数说明

### 1、返回参数

参数	类型	描述
errcode	int	错误码, 0 表示成功
errmsg	string	错误信息
identify_ret	int	认证结果
identify_time	uint32	认证时间
validate_data	string	用户读的数字
openid	string	用户 openid
user_id_key	string	用于后台交互表示用户姓名、身份证的凭证
finish_time	Uint32	认证结束的时间
id_card_number_md5	String	身份证号的 md5 (最后一位 X 为大写)

参数	类型	描述
name_utf8_md5	String	姓名 MD5

## 2、错误码对应信息

errcode	备注
84001	非法 identity_id
84002	用户信息过期
84003	用户信息不存在

# 五、小程序辅助接口：检查设备是否支持人脸检测

## 5.1 接口名称

接口：wx.checkIsSupportFacialRecognition(OBJECT)

功能：检查设备是否支持人脸检测

## 5.2 接口说明和使用

小程序调用该接口，可以检测当前手机设备是否具备支持人脸检测的能力，可与以上接口分开使用，为了用户体验，建议调用后对手机设备不支持的用户做对应功能处理。

## 5.3 接口说明和使用

### 1、OBJECT 参数说明：

参数	类型	必填	说明
success	Function	否	调用成功回调
fail	Function	否	调用失败回调
complete	Function	是	调用完成回调（成功或失败都会回调）
checkAliveType	Number	否	人脸核验的交互方式，默认读数字（见表 2）

表 2：checkAliveType 的值和对应的解释：

参数	解释
0	读数字（默认）
1	屏幕闪烁
2	先检查是否可以屏幕闪烁，不可以则自动为读数字

### 2、CALLBACK 返回参数

参数	类型	说明
errMsg	Boolean	错误信息
errCode	Number	错误码

### 3、回调结果说明

回调类型	ErrCode	说明
sucess	0	支持人脸采集
fail	10001	不支持人脸采集：设备没有前置摄像头
fail	10002	不支持人脸采集：没有下载到必要模型
fail	10003	不支持人脸采集：后台控制不支持

#### 4、示例代码

```

wx.checkIsSupportFacialRecognition({
  success() {
    // 支持人脸检测
  },
  fail(res) {
    // 不支持人脸检测
  }
})

```

请根据调用接口的情况，代码书写对应的 JSAPI 接口。

## 六、接口申请

### 6.1 联系腾讯开通权限

需提供以下信息，以用作权限开通：

序号	信息概述	具体信息内容
1	业务方名称	(必填，服务提供方主体(单位公司)名称)
2	使用该接口的场景和用途	(必填，用户使用该接口时的场景及服务内容)
3	使用该接口的主体性质	(必填，选择公众号或小程序)
3	使用该接口的公众号/小程序名称 (正式)	(必填，提供正式服务的名称)
	使用该接口的公众号/小程序对应 appid (正式)	(必填，提供正式服务的 appid)
4	使用该接口的公众号/小程序名称 (测试)	(选填，用于测试的名称，主体也必须符合范围要求，有调用次数限制)
	使用该接口的公众号/小程序对应 appid (测试)	(选填，用于测试的 appid，主体也必须符合范围要求，有调用次数限制)
5	日均调用量预估	(必填，调用次数、用户数预估)
6	最高并发调用量预估	(必填，分钟级并发调用次数)
7	日并发调用量均值预估	(必填，平均分钟级并发调用次数)
8	业务方产品联系人姓名、手机、邮箱	(必填，请信息提供完全)
9	业务方开发联系人姓名、手机、邮箱	(必填，请信息提供完全)
10	腾讯公司接口人姓名、邮箱	(若有，请填写。请信息提供完全，完整姓名和邮箱)



## 6.2 申请方式

申请邮件请发送至腾讯工作邮箱 [wx\\_city@tencent.com](mailto:wx_city@tencent.com)

腾讯工作人员会邮件回复资料是否准确无误，并说明权限开通时间及计划。