

# 媒体云人脸识别 REST API 帮助文档

V1.02

发布日期：2013 年 10 月 12 日

百度开发者中心

(版权所有, 翻版必究)

## 修订记录

2013.09.29	V1.00 版
2013.10.11	V1.01 版 统一对外请求地址
2013.10.12	V1.02 版 调整错误码

## 目录

一、概述.....	4
二、媒体云人脸识别 REST API 使用说明 .....	4
2.1 开启人脸识别服务 .....	4
2.2 开通媒体云人脸识别 API 权限 .....	7
2.3 获取 access_token 步骤 .....	7
2.4 API 请求方式说明 .....	9
三、媒体云人脸识别 API 列表 .....	10
3.1 人脸检测 .....	10
3.1.1 参数说明 .....	10
3.1.2 具体示例 .....	11
四、媒体云人脸识别 API 错误码总表 .....	12

## 一、概述

百度媒体云人脸识别服务依托百度世界领先的人脸识别算法，提供了（1）人脸及五官几何位置定位功能；（2）人脸图像的性别、表情等人脸属性分析功能。

人脸识别服务提供通过 REST API 的方式来使用人脸识别服务，本文档描述了使用人脸识别服务 REST API 的方法。

除了原生的 REST（Representational State Transfer，即“表述性状态转移”）API 之外，人脸识别服务还提供了 ANDROID、IOS、WP7、JAVA 及 PHP 多种平台的 SDK 来帮助开发者缩短开发周期，具体请参考“SDK”部分相关内容([开发者中心通用 SDK](#))。

## 二、媒体云人脸识别 REST API 使用说明

说明：开发者中心管理控制台可以通过开发者中心首页右上角“消息”旁边的已标记选项进入，如图 1 红框所示。



图 1 管理控制台入口

### 2.1 开启人脸识别服务

使用媒体云人脸服务之前，您须按照本节描述的步骤来完成服务的开启，后台服务的注册。通过开发者中心“管理控制台”，进入“我的应用列表”，创建应用。

（1）进入“管理控制台”



图 2 进入管理控制台

- (2) 创建应用 test, 点击“创建应用”



图 3 创建应用

- (3) 填写应用名称“test”以及应用类型

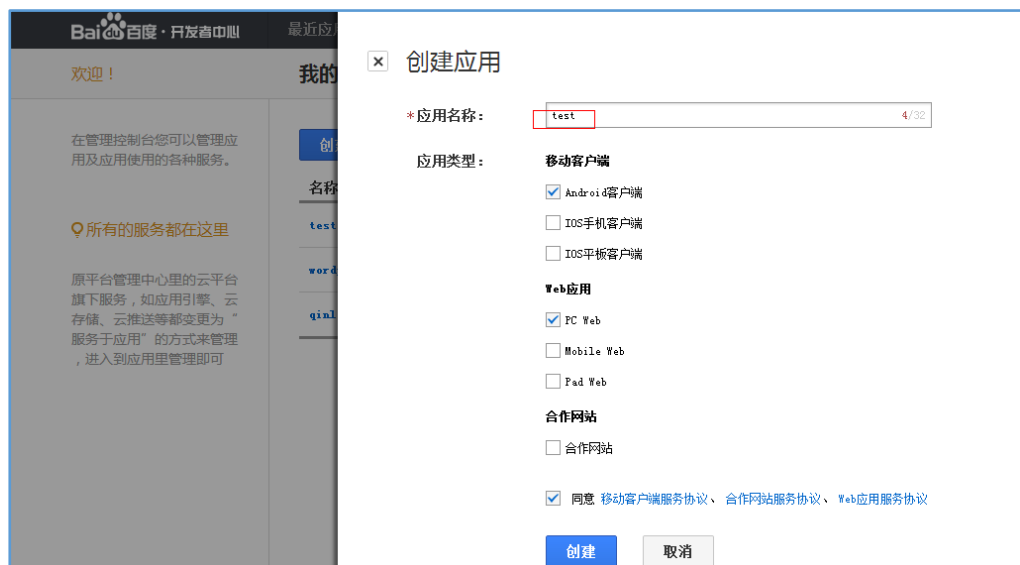


图 4 填写创建应用信息

- (4) 创建应用成功，点击左侧“媒体云”开启媒体云人脸识别服务



图 5 申请使用人脸识别服务

- (5) 填写使用场景和申请理由



图 6 人脸服务申请信息填写

- (6) 开启服务成功后，可以在控制台来使用人脸服务的统计分析功能。



图 7 人脸服务统计功能界面

## 2.2 开通媒体云人脸识别 API 权限

媒体云人脸识别 API 所有 REST API 都首先需要获取 Access Token。Access Token 是用户身份验证和授权的凭证。详细信息请参考百度开发者中心帮助文档的 [OAuth 介绍](#) 部分。媒体云人脸识别采用的是 [Client Credentials](#) 授权方式，即采用应用公钥、密钥获取 Access Token，适用于任何带 server 类型应用，通过此授权方式获取 Access Token 仅可访问平台授权类的接口，详见 [Client Credentials 授权说明](#)。

## 2.3 获取 access\_token 步骤

- (1) 如果您已是百度用户，请您访问百度开发者中心并使用百度账号直接登录。如果您还不是百度用户，请您先访问[百度 Passport](#)，填写注册邮箱和密码并激活成为百度用户，然后访问[百度开发者中心](#)使用百度账号登录。
- (2) 您需要创建一个应用以获取 API Key (client\_id) 和 Secret Key (client\_secret)，申请方式见[创建应用](#)。

说明：目前创建应用入口如下截图红框所示：



图 8 进入控制台

- (3) 创建应用之后，您可以从管理中心进入相应应用得到 API Key (client\_id) 和 Secret Key (client\_secret)。

### ● 基本信息页

从管理控制台进入创建的应用，如下图所示，API Key (client\_id) 和 Secret Key (client\_secret) 即显示于其中，ID 即为应用 id (appid)。



名称:

Icon:

ID:

1

API Key:

Secret Key:

重置

创建时间:

2013-08-23 19:30:20

更新时间:

2013-08-23 19:34:45

应用类型:

移动客户端

☐ Android客户端

☐ iOS手机客户端

☐ iOS平板客户端

图 9 应用基本信息

- (4) 根据百度 OAuth 的 [Client Credentials](#) 授权方式获取 access\_token。百度 OAuth 的 Client Credentials 和其他授权方式的说明见[百度 OAuth 介绍](#)的“百度支持的 OAuth 授权”部分。

```
https://openapi.baidu.com/oauth/2.0/token?
grant_type=client_credentials&
client_id=xxxxx
client_secret=yyyyyy
```

其中，grant\_type 的值固定为 client\_credentials，client\_id 即为（3）中截图所示的 API Key 的值，client\_secret 即为（3）中截图所示的 Secret Key 的值。更为详细的介绍请参见 [Client Credentials](#) 授权说明。

响应如下所示，其中 access\_token 字段即为请求 REST API 所需的令牌，access\_token 有效期一般为一个月，开发者需要对 access\_token 进行有效性判断，过期需重新申请 access\_token。

```
HTTP/1.1 200 OK
Content-Type: application/json
Cache-Control: no-store
{
  "access_token":
    "1.a6b7dbd428f731035f771b8d15063f61.86400.1292922000-2346678-124328",
  "expires_in": 86400,
  "session_key": "ANXxSNjwQDugf8615OnqeikRMu2bKaXCdILxn",
  "session_secret": "248APxvxjCZ0VEC43EYrvxqaK4oZExMB",
}
```



## 2.4 API 请求方式说明

- ◆ 目前人脸检测接口支持 **POST** 和 **GET** 方式。
- ◆ 媒体云人脸识别 **REST API** 的所有参数在传入时应当使用：**UTF-8** 编码。
- ◆ **URL 格式**

```
https://openapi.baidu.com/rest/2.0/media/v1/face/{request_type}?{query_string}
```

- ◆ **参数说明**

参数名称	描述
request_type	MEDIA REST API 操作方法名称，对应操作说明：detect 为人脸检测。
query_string	放在 HTTP 头部传入的参数，必须经过 UriEncode 编码。

- ◆ **HTTP GET 和 POST 方式使用说明**

URL	https://openapi.baidu.com/rest/2.0/media/v1/face/{request_type}?{query_string}	
请求方式	GET	POST
请求参数	全部携带在 HTTP 请求头部的 query_string 中。	既可携带在 query_string 中，也可携带在 HTTP Body 中 <ul style="list-style-type: none"> <li>◆ 携带在 query_string 中的参数的值，必须进行 UriEncode 编码；</li> <li>◆ 携带在 HTTPBody 中的参数，需要进行 UriEncode 编码。</li> </ul> 注:HTTP URL 长度有限，若参数值长度过长，建议将参数放在 HTTP Body 中进行传输。
HTTP BODY	不携带 HTTP BODY	application/x-www-form-urlencoded
注意	如果 HTTP BODY 和 query_string 存在相同的参数，则以 query_string 中的参数为准。	

- ◆ **使用示例**

### 1. GET 请求:

说明：用 HTTP GET 请求方式发送两个参数：access\_token =xxxx 和 url =xxxx。

```
https://openapi.baidu.com/rest/2.0/media/v1/face/detect?access_token=urlencode(xxxxx)&url=urlencode(xxxxxx)
```

### 2. POST 请求:

说明：用 HTTP POST 方式发送两个参数：access\_token =xxxx 和 url= xxxx。

```
POST /rest/2.0/media/v1/face
User-Agent: curl/7.12.1 (x86_64-redhat-linux-gnu) libcurl/7.12.1 OpenSSL/0.9.7a zlib/1.2.1.2
libidn/0.5.6
Pragma: no-cache
Accept: */*
Host:openapi.baidu.com
Content-Type:application/x-www-form-urlencoded
Content-Length:123

access_token =urlencode(xxxx)&url =urlencode(xxxx)
```

#### ◆ API 响应格式说明

返回	正常请求	异常请求
HTTP 状态码	200	200
HTTP BODY	API 响应内容(json)	异常请求的返回值为 json 字符串。 例如： <pre>{   "error_code":80001,   "error_msg":" Invalid Parameter!" }</pre> 说明： -error_code:错误码； -error_msg:错误描述信息；

## 三、媒体云人脸识别 API 列表

### 3.1 人脸检测

#### 3.1.1 参数说明

功能	对用户提交的图片数据进行人脸检测			
Http 请求方式	Get/Post			
Url	https://openapi.baidu.com/rest/2.0/media/api/v1/face/detect http/1.1			
输入参数	参数名称	参数类型	是否必须	描述
	access_token	string	是	访问令牌，开发者准入标识，HTTPS 调用时必须使用（见 2.1 节说明）
	url	string	是	要处理的图片地址(需要经过 urlencode)
	callback_url	string	否	离线请求回调地址，人脸服务收到离线请求后，将立即返回一个 session_id,处理完成后将检测

					结果通知 callback_url
返回参数	失败	参数名称	参数类型	是否必须	描述
		error_code	int	否	错误码, 如果不出错, 则返回值没有该字段。
		error_msg	string	否	错误提示, 如果不出错, 则返回值没有该字段。
	成功	Face	Json 数组	是	在原图片中检测到的人脸数组
		attribute	json	否	检测到人脸属性: 性别和笑脸
		face_id	string	否	检测到的人脸 id
		position	json	否	人脸位置: 中心点坐标, 宽和高
		img_id	string	是	当前图片 id
		url	string	是	图片 url
		session_id	string	是	人脸服务生成的请求 id, 可通过该 id 查询处理结果
		img_width	string	是	图片宽度
		img_height	string	是	图片高度

### 3.1.2 具体示例

请求示例		<p>注意: query string 需要先进行 urlencode。</p> <p>curl-v"<a href="https://openapi.baidu.com/rest/2.0/media/v1/face/detect?access_token=3.ad2ee1e9b351cd800fd62bddc8bef016.2592000.1377163536.4212382003-260146&amp;url=http://c.hiphotos.baidu.com/album/s%3D800/sign=6bd70c1da044ad342abf8a87e0a00c08/8d5494eef01f3a29d7017b3e9825bc315d607c71.jp">https://openapi.baidu.com/rest/2.0/media/v1/face/detect?access_token=3.ad2ee1e9b351cd800fd62bddc8bef016.2592000.1377163536.4212382003-260146&amp;url=http://c.hiphotos.baidu.com/album/s%3D800/sign=6bd70c1da044ad342abf8a87e0a00c08/8d5494eef01f3a29d7017b3e9825bc315d607c71.jp</a>"</p>
返回示例	成功	<pre>{   "face": //在原图片中检测到的人脸数组   [{     "attribute":{       "gender": { //检测到的人脸的性别         "value": "male" //人脸性别结果         "confidence": "1.0" , //性别的置信度       },       "smiling": { //检测到的笑脸         "confidence": "1.0" //笑脸的置信度       },     },     "face_id": "ffffff32c1f3813f0944f52427ed060b" , //检测到的人脸的 id   } }</pre>

		<pre> "position": { //检测到的人脸的位置，坐标均进行了归一化，值在 0~1 之间   "center": {     "x": "0.591000", //人脸区域中心 x 坐标,     "y": "0.267000", //人脸区域中心 y 坐标     "width": "0.181290", //人脸区域宽度     "height": "0.120600", //人脸区域高度     "eye_left": {       "x": "0.556230", //左眼 x 坐标       "y": "0.230250", //左眼 y 坐标     },     "eye_right": {       "x": "0.638620", //右眼 x 坐标       "y": "0.244190", //右眼 y 坐标     },     "mouth": {       "x": "0.574910", //嘴巴中心 x 坐标       "y": "0.301350", //嘴巴中心 y 坐标     },   }, }, "img_id": "966f0f32c1f3813f0944f52427ed060b", //图片 id "url": "" //请求中的图片 url "session_id": "966f0f32c1f3813f0944f52427jjjkkk" //唯一标识请求 "img_width": "320" //图片宽度 "img_height": "240" //图片高度 } </pre>
	失败	<pre> {"error_code":xxx, "error_msg":"yyyy"} </pre>

## 四、媒体云人脸识别 API 错误码总表

说明：所属接口列，表明该错误码会在哪些接口中出现，1：检测。

错误码	错误信息	备注	所属接口
无	无	没有错误	
70001	Invalid Parameter!	请求参数错误	1
70503	Service Unavailable!	人脸服务不可用	1
70131	Internal Error!	内部处理错误	1
70132	Can not access image!	不能访问图片 url	1
70133	Quota not enough!	人脸识别 API 配额不足	1