



天翼云人脸识别产品

用户操作手册

天翼云科技有限公司

目 录

1 产品简介	1
1.1 产品定义	1
1.2 功能特性	1
1.2.1 人脸检测	1
1.2.2 人脸属性识别	1
1.2.3 人脸比对	1
1.2.4 人脸活体检测	2
1.2.5 是否戴口罩识别	2
1.3 产品优势	2
1.4 应用场景	2
1.4.1 门禁人脸识别	2
1.4.2 市场营销-企业商务	3
1.4.3 商业银行	3
2 快速入门	4
2.1 创建人脸识别应用	4
2.1.1 人脸识别产品订购入口	4
2.1.2 创建人脸识别应用订单	5
2.1.3 订单确认	6
2.2 人脸识别用户控制台	8
3 用户指南	10
3.1 产品价格	10

3.2 产品购买	11
3.3 产品续订	13
3.4 产品退订	13
4 API 参考	14
4.1 API 概览	14
4.1.1 概述	14
4.1.2 API 概览	14
4.1.3 状态码	14
4.2 如何调用 API	16
4.2.1 终端节点	16
4.2.2 构造请求	16
4.2.3 认证鉴权	17
4.2.4 Python 调用示例	20
4.2.5 Java 调用示例	23
4.3 API	23
4.3.1 人脸检测	23
4.3.2 人脸属性识别	32
4.3.5 人脸比对	37
4.3.5 人脸活体检测	42
4.3.5 是否戴口罩识别	47
4.3 更新历史	53

5 常见问题.....	53
5.1 计费类.....	53
5.2 购买类.....	54
5.3 操作类.....	54
5.4 使用限制	55

1 产品简介

1.1 产品定义

人脸识别是中国电信云公司自研 AI 平台提供的产品之一，通过自主研发图像识别人工智能模型，提供人脸系列 API 服务-人脸检测定位，人脸属性识别，人脸比对。通过订购天翼云人脸识别产品，可将此服务快速高效的部署到您的应用中。人脸检测定位就是识别图片中是否有人脸，有哪些人脸。人脸属性识别主要是通过人脸推算人的性别，年龄等。人脸对比主要是判断两张人脸是否同一个人。

天翼云用户通过订购以上人脸识别的应用，无需进行模型开发仅需 API 接口对接到 AI 上云，即可轻松、高效、准确的实现云上产品的人脸识别。人脸识别将高效帮助用户识别用户人脸信息，帮助用户在安防、人脸支付等领域实现智能化应用落地。

1.2 功能特性

1.2.1 人脸检测

通过用户输入人脸照片，高效快速地检测图像中的人脸，给出人脸的位置和可信度阈值，方便用户后续应用。

1.2.2 人脸属性识别

在人脸检测基础上，分析检测后的人脸性别、年龄属性，扩展用户使用人脸检测服务的应用场景。

1.2.3 人脸比对

通过对用户输入的两张人脸图像进行比对，分析是否为同一个人；同时可返回两张图像人脸的相似度以及比对结果等。

1.2.4 人脸活体检测

基于深度学习方法，分析人脸图像的摩尔纹、成像畸形等信息，实现静默活体判断，有效防止照片等非活体攻击。

1.2.5 是否戴口罩识别

准确高效识别图像中人脸是否正确佩戴口罩，支持多种类型口罩识别，并可对图片中多张人脸实时检测，有效应对疫情场景。

1.3 产品优势

识别准确率高：使用业界先进的人工智能技术，基于海量标注样本数据。

识别效率高：实时返回识别信息。

识别区域灵活：自动对规定区域进行检测，报告违规信息，可实时更改违规区域。

业务场景多样：业务场景包括工作服识别，电子围栏，安全帽正确佩戴识别，满足工厂多样化需求。

1.4 应用场景

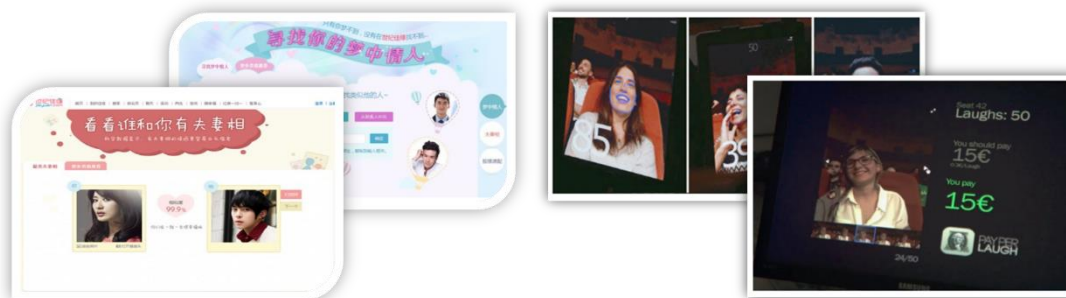
1.4.1 门禁人脸识别

随着人们生活水平的提高，人们更加注重家居环境的安全，安防观念不断加强；伴随着这种需求的提高，智能门禁系统应运而生，越来越多的企业、商铺、家庭都安装了各种各样的门禁系统。



1.4.2 市场营销-企业商务

面部识别技术在营销上主要有两方面的应用：首先，可以识别一个人的基本个人信息，例如性别、大致年龄，以及他们看过什么，看了多久等。户外广告公司，例如 Val Morgan Outdoor (VMO)，开始采用面部识别技术来收集消费者数据。其次，该技术可以用于识别已知个人，例如小偷，或者已经加入系统的会员。这方面的应用已经引起一些服务提供商和零售商的注意。



1.4.3 商业银行

1. 利用人脸识别技术防范网络风险

利用人脸识别技术准确认定持卡人的真实身份，确保持卡人的资金安全。另外，还可以通过人脸识别技术进一步锁定不法分子，有利于公安机关快速破案。



2. 脸识别技术在治理假钞方面的应用

银行的 ATM 机中没有假钞鉴别设备，只是在清机人员放入现金前做了鉴别，这样的措施并不够完善，且容易造成银行与持卡人之间的纠纷。即使是现金存款机（CRS）有假钞鉴别功能，但往往因为假钞识别特征提取的滞后，而被不法分子所利用。不法分子先存入假钞，然后马上在柜台或其他自助设备上提取真钞，以此手段谋取不法利益。

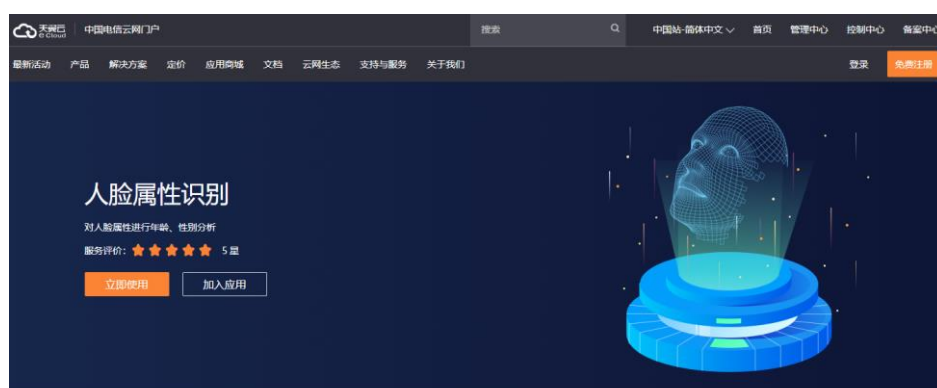


2 快速入门

2.1 创建人脸识别应用

2.1.1 人脸识别产品订购入口

首先，用户使用天翼云网门户账号登录天翼云网门户后，点击菜单栏“产品”项，进入【人工智能】，即可看到人脸识别产品相应服务。以“人脸属性识别”为例，点击“人脸属性识别”名称即可进入“人脸属性识别”产品详情页。



产品优势



准确高效



样本量大



小人脸识别



多种属性

点击【立即使用】按钮，进入人脸识别创建页面，点击“新的应用”



2.1.2 创建人脸识别应用订单

用户根据自身需要输入“应用名称”与“应用概述”。

应用名称：长度为 4-15 个字符。

A screenshot of a web form titled "新建应用" (New Application). The form has a close button (X) in the top right corner. It contains two input fields. The first is labeled "* 应用名称:" (Application Name) and has a placeholder text "人脸属性识别应用场景" (Face Attribute Recognition Application Scenario). The second is labeled "应用概述:" (Application Overview) and also has a placeholder text "人脸属性识别应用场景". Both fields have a small circular icon with the number 1 to their right. At the bottom of the form are two buttons: "取消" (Cancel) and "确定" (Confirm).

用户创建完应用后，点击“确定”和“下一步”，进入应用订购页面

我的能力

人脸识别	安全生产	图像识别	印刷文字识别	自然语言处理	内容安全
人脸属性识别					
API	状态	调用量限制	购买次数包	领取免费额度	
人脸属性识别	① 待付费	—	购买	立即领取	
人脸检测					
API	状态	调用量限制	购买次数包	领取免费额度	
人脸检测	① 待付费	—	购买	立即领取	
是否戴口罩识别					
API	状态	调用量限制	购买次数包	领取免费额度	
是否戴口罩识别	① 待付费	—	购买	立即领取	

2.1.3 订单确认

在“人脸属性识别”列点击“购买”，进入产品规格及数量配置页面。

人脸属性识别

选择配置

规格	公测截止时间	单价	购买数量
100万次	2020-11-30 23:59:59	¥0.00	<div>- 1 +</div>
1000万次	2020-11-30 23:59:59	¥0.00	<div>- 0 +</div>
5000万次	2020-11-30 23:59:59	¥0.00	<div>- 1 +</div>
10000万次	2020-11-30 23:59:59	¥0.00	<div>- 0 +</div>

配置清单

清空

规格	购买数量	价格
100万次	1	0
5000万次	1	0

费用合计

¥0元

配置费用 ¥ 0

参考价格，具体金额请以账单为准。了解详情

下一步：确认配置

用户根据需选择要开通的资源包规格及数量，点击“下一步：确认配置”，进入配置确认界面。

配置

产品名称	计费方式	规格	公测截止时间	购买数量	单价
人脸属性识别	次数包	100万次	2020-11-30 23:59:59	1	¥0.00
人脸属性识别	次数包	5000万次	2020-11-30 23:59:59	1	¥0.00

协议

☐ 我已阅读并同意相关协议，[中国电信天翼云公测服务协议（自研人脸识别产品）](#)

配置费用 ¥0

参考价格，具体扣费请以账单为准。[了解详情详情](#)

上一步

立即购买

用户勾选服务协议，点击“立即购买”，进入订单支付页面

订单号: 20201106180755918191 产品类型: 包周期新购 创建时间: 2020-11-06 18:57:55 更新时间: 2020-11-06 18:57:55

产品1		购买数量	规格	周期	金额 (元)
人脸识别	-	x 1	—	1次	0.00
人脸识别	-				0.00

费用合计: 0.00元

重要提示: 对云主机的使用请遵循国家相关法律法规之规定, 对于违反相关法律法规的行为, 服务商会记录并上报, 并保留追究法律责任的权利, 停止所有服务, 不退余额。弹性IP不支持降规格操作, 购买时请慎重选择。

- 支付订单前请确认所有商品已进行实名认证, 如元, 请点[立即实名认证](#)
- 云主机订单不能使用优惠券和代金券支付
- 订单不支持同时使用代金券和优惠券

支付方式 ☒ 预付费

账户余额支付 (元)

代金券支付 (元)

优惠券

订单费用: + 0.00元
账户余额支付: - 0.00元
代金券支付: - 0.00元
优惠券优惠: - 0.00元

需支付 0.00元

立即购买

用户确认订单费用是否准确，费用无误点击“立即支付”。完成支付后进入“支付结果”界面。

订单管理 - 云订单 > 订单详情 > 支付结果

支付成功

返回订单列表

订单号: 20201106185755918191

产品类型: 包周期新购

创建时间: 2020-11-06 18:57:55

更新时间: 2020-11-06 19:08:03

产品1	已完成					0.00元
产品	配置	订购数量	所属资源池	周期	金额 (元)	
人脸识别	-	x 1	-	1次	0.00	
人脸识别	-				0.00	

费用合计: 0.00元

订单费用: + 0.00元

账户余额支付: - 0.00元

代金券支付: - 0.00元

实际支付: 0.00元

2.2 人脸识别用户控制台

点击页面右上角“控制台”，进入控制中心。

天翼云 @ Cloud

中国电信云网门户

搜索

中国站-简体中文

首页

管理中心

控制台

天翼云诸葛AI平台

开放能力

AI开发中心

文档中心

支持与反馈

搜索

控制台

提供者中心

chenz@cf

总览

我的能力

我的应用

应用监控

应用告警

已开通能力

人脸识别

通用型OCR

工作版识别

人脸比对

图片鉴黄

能力调用排行榜

2020-11-05 ~ 2020-11-05

能力异常分布情况

2020-11-05 ~ 2020-

排名

API组合包

调用次数(次)

用户点击“我的应用”，可以查看已经开通的应用。

我的应用

新建应用

默认应用

人脸属性识别应用场景

情感分析应用场景

工作版识别应用场景1

工作版识别应用场景

人脸识别订购体验

OCR文字识别体验

通用OCR场景应用

人脸识别检测测试

人脸识别检测应用0114

人脸识别检测测试1104

8

用户点击“应用监控”，可以查看所创建应用的请求次数、响应时间与请求流量。



用户点击“应用告警”，可以了解应用使用过程中系统所反馈的异常情况。



3 用户指南

3.1 产品价格

人脸识别采用封顶资源包的方式订购：

产品规格	人脸检测定位、人脸属性识别价格 (元/年)	人脸比对价格 (元/年)
100万次/年	450	3100
1000万次/年	4100	30400
5000万次/年	17500	142500
1亿次/年	25000	255000

订购说明：

1、为满足客户不同业务使用量需求，每类 API 设置四档套餐，如：用户评估认为应用每年人脸属性识别的 API 调用大概为 500 万次，则可以购买五个 100 万次/年的调用次数包，若大概为 1000 万次，则可以购买一个 100 万次/年的调用次数包

2、人脸检测定位和人脸属性识别按照所检测的图像张数进行计数，人脸比对按照 API 调用次数进行收费（用户在比对时会指定两张照片，其中一张可能是系统中已存的，一张是用户新上传的，也有可能两张需比对的图像都是新上传的）

3. 用户预先购买 N 个固定额度的资源包，后续使用过程中产生的人脸检测定位或属性识别的张数，以及人脸识别的次数从资源包中抵扣，有效期内未使用的资源不会流转至下一年。有效期内资源包中的次数用完之后，用户无法再调用 AI 产品 API。对于有效期内未使用的资源包剩余调用次数，本产品不支持退订。

备注：产品价格以价格发文为准

3.2 产品购买

人脸识别服务包括人脸检测、人脸属性识别、人脸比对、人脸活体检测与是否戴口罩识别等 5 种能力，如下以“人脸属性识别”为例来进行操作说明。

首先，用户使用天翼云网门户账号登录天翼云网门户后，点击菜单栏“产品”项，进入【人工智能】，即可看到人脸识别产品相应服务。以“人脸属性识别”为例，点击“人脸属性识别”名称即可进入“人脸属性识别”产品详情页

在创建人脸识别应用和订单后（详见快速入门）

在“人脸属性识别”列点击“购买”，进入产品规格及数量配置页面。

人脸属性识别

选择配置

规格	公测截止时间	单价	购买数量
100万次	已截止	¥450.00	<div>- 0 +</div>
1000万次	已截止	¥4100.00	<div>- 0 +</div>
5000万次	已截止	¥17500.00	<div>- 0 +</div>
10000万次	已截止	¥25000.00	<div>- 0 +</div>

配置清单

清空

请在左侧选择次数包类型

配置费用 **¥ 0**

参考价格，具体扣费请以账单为准。[了解计费详情](#)

下一步：确认配置

用户根据需要进行选择的资源包规格及数量，点击“下一步：确认配置”，进入配置确认界面。

人脸属性识别

选择配置

规格	公测截止时间	单价	购买数量
100万次	已截止	¥450.00	<div>- 1 +</div>
1000万次	已截止	¥4100.00	<div>- 0 +</div>
5000万次	已截止	¥17500.00	<div>- 0 +</div>
10000万次	已截止	¥25000.00	<div>- 0 +</div>

配置清单

清空

规格	购买数量	价格
100万次	1	450

费用合计
¥450元

配置费用 **¥ 450**

参考价格，具体扣费请以账单为准。[了解计费详情](#)

下一步：确认配置

配置

产品名称	计费方式	规格	公测截止时间	购买数量	单价
人脸属性识别	次数包	100万次	已截止	1	¥450.00

协议

☒ 我已阅读并同意相关协议。 [中国电信天翼云公测服务协议（自研人脸识别产品）](#)

配置费用 ¥ 450

参考价格，具体扣费请以账单为准。 [了解计费详情](#)

上一步

立即购买

用户勾选服务协议，点击“立即购买”，进入订单支付页面。

产品1 待支付 450.00元

产品	配置	订购数量	所属资源池	周期	金额 (元)
人脸识别	人脸属性识别_100万次	x 1	—	1次	450.00元

费用合计: 450.00元

郑重提示：对云主机的使用请遵循国家相关法律法规之规定，对于违反相关法律法规的行为，服务商将关闭服务器，并视情况决定是否关闭用户帐号，停止所有服务，不退余款。弹性IP不支持降级操作，购买时请慎重选择。

1、支付订单请务必确认所有者已进行实名认证，如无，请点击 [立即实名认证](#)

2、云市场订单不能使用优惠券支付

3、订单不支持同时使用代金券和优惠券

支付方式 ☒ 预付费

账户余额支付 (元) 无可用账户余额
(账户余额 ¥ 0.00)

代金券支付 (元) 最多只可支付 20.00元

优惠券 无可用优惠券

订单费用: + 450.00元

账户余额支付: - 0.00元

代金券支付: - 0.00元

优惠券优惠: - 0.00元

需支付 450.00元

用户确认订单费用是否准确，费用无误点击“立即支付”。完成支付后进入“支付结果”界面。

12

✓ 支付成功

[返回订单列表](#)

订单号: 20220923145153779725 订单类型: 包周期新购 创建时间: 2022-09-23 14:51:56 更新时间: 2022-09-23 14:52:02

产品1	开通中					0.00元
产品	配置	订购数量	所属资源池	周期	金额 (元)	
人脸识别	是否佩戴口罩_1000万次	x 1	—	1次	0.00元	

费用合计: 0.00元

支付方式: 预付费
订单费用: + 0.00元
账户余额支付: - 0.00元
代金券支付: - 0.00元
在线支付: 0.00元

3.3 产品续订

续订说明: 当已订购的资源包订单即将到期或即将用完时, 可通过订购新的资源包进行续订。

3.4 产品退订

退订说明: 已购订单内的资源包不支持退订。

4 API 参考

4.1 API 概览

4.1.1 概述

本说明提供了人脸识别产品 API 的描述、语法、参数说明及示例等内容。

4.1.2 API 概览

功能名称	描述
人脸检测	用于检测输入图像中的人脸，输出人脸位置坐标
人脸属性识别	用于检测输入图像中的人脸年龄、性别等属性
人脸比对	用于检测输入的两张图像中的人脸相似度
人脸活体检测	用于检测输入图像中的人脸是否为活体
是否戴口罩识别	用于检测输入图像中的人脸是否戴口罩

4.1.3 状态码

1、 请求状态码

正常状态码	描述
-------	----

200	请求成功
3**	请求转移
4**	客户端错误
5**	服务端错误

2、 全局请求返回错误码

错误码	描述
10002	生成签名时官网 ak 信息错误
10020	签名错误
40002	缺少 appkey 头
40006	无效的 appkey
40008	不支持的请求类型
40009	IP 未被 APP 授权
40010	IP 未被 API 授权
50000	服务内部错误
50001	服务未注册

50002	应用未开通
50003	API 中无效的 URL 请求
51001	购买服务已过期
51002	收费 API 未购买
51003	API 可用次数已不足

4.2 如何调用 API

4.1.1 终端节点

<https://ai-global.ctapi.ctyun.cn>

4.1.2 构造请求

请求地址: {终端节点地址}+{对应接口 URI}

请求头:

Key	Value(说明)
Content-Type	application/json
ctyun-eop-request-id	用户请求 id, 通过 uuid 生成, 形如 33dfa732-b27b-464f-b15a-21ed6845afd5
Eop-Authorization	由天翼云官网 accessKey 和 securityKey 经签名后生成, 签名逻辑详见后续说明

eop-date	请求时间, 形如 yyyyymmddTHHMMSSZ, 例如 20211221T163014Z
host	终端节点域名
appkey	诸葛 AI 平台-控制台-我的应用中每个应用具有的 AppKey 信息, 鉴权时需要加入 header

4.1.3 认证鉴权

1 信息的获取

云网平台获取

登录云网门户, 在“控制台” -> “个人中心” -> “第三方账号绑定”, 通过创建或者查看获取 ak, sk。



2 基本签名流程

ctyun-eop-ak/ctyun-eop-sk 基本签名流程

1、待签字符串: 使用规范请求和其他信息创建待签字符串;

- 2、计算密钥：使用 HEADER、ctyun-eop-sk、ctyun-eop-ak 来创建 Hmac 算法的密钥；
- 3、计算签名：使用第三步的密钥和待签字符串在通过 hmacsha256 来计算签名。
- 4、签名应用：将生成的签名信息作为请求消息头添加到 HTTP 请求中。

3 创建待签名字符串

待签名字符串的构造规则如下：

待签名字符串=需要进行签名的 Header 排序后的组合列表 + "\n" + 排序的 query + "\n" + toHex(sha256(原封的 body))

需要进行签名的 Header 排序后的组合列表 (排序的 header)	header 以 header_name:header_value 来一个一个通过\n 拼接起来，EOP 是强制要求 ctyun-eop-request-id 和 eop-date 这个头作为 Header 中的一部分，并且必须是待签名 Header 里的一个。需要进行签名算法的 Header 需要进行排序（将它们的 header_name 以 26 个英文字母的顺序来排序），将排序后得到的列表进行遍历组装成待签名的 header。
排序的 query	query 以&作为拼接，key 和值以=连接，排序规则使用 26 个英文字母的顺序来排序，Query 参数全部都需要进行签名
toHex(sha256(原封的 body))	传进来的 body 参数进行 sha256 摘要，对摘要出来的结果转十六进制

排序的 header 例子：

假设你需要将 ctyun-eop-request-id、eop-date、host 都要签名，则待签名的 header 构造出来是：

```
ctyun-eop-request-id:123456789\neop-date:20210531T100101Z\nhost:1.1.1.1:9080\n
```

ctyun-eop-request-id、eop-date 和 host 的排序就是这个顺序，如果你加入一个 ccad 的 header；同时这个 header 也要是进行签名,则待签名的 header 组合：

```
ccda:123\nctyun-eop-request-id:123456789\neop-date:20210531T100101Z\nhost:1.1.1.1:9080\n
```

4 构造动态密钥

发起请求时，需要构造一个 eop-date 的时间，这个时间的格式是 yyyyymmddTHHMMSSZ;言简意赅一些，就是年月日 T 时分秒 Z

- 1、先是拿你申请来的 ctyun-eop-sk 作为密钥，eop-date 作为数据，算出 ktime
- 2、拿 ktime 作为密钥，你申请来的 ctyun-eop-ak 数据，算出 kAk;
- 3、拿 kAk 作为密钥，eop-date 的年月日值作为数据；算出 kdate

eop-date	yyyymmddTHHMMSSZ (20211221T163614Z) (年月日 T 时分秒 Z)
Ktime	使用 ctyun-eop-sk 作为密钥，eop-date 作为数据，算出 ktime; Ktime = hmacSha256(ctyun-eop-sk, eop-date)
kAk	使用 ktime 作为密钥，你申请来的 ctyun-eop-ak 数据，算出 kAk; kAk = hmacsha256(ktime,ctyun-eop-ak)
kdate	使用 kAk 作为密钥，eop-date 的年月日值作为数据；算出 kdate; kdate = hmacsha256(kAk, eop-date)

5 签名应用及示例

由“构造动态密钥”和“创建待签名字符串”分别的出来的待签名字符串 string_sigture、kdate 生成 Signature;

Signature	待签名字符串 string_sigture、kdate; 再根据 hmacsha256(kdate,string_sigture)得出的结果，再将结果进行 base64 编码得出 Sigture
Eop-Authorization	ctyun-eop-ak Header=你构造待签名字符串时的 header 排序 Signature (ctyun-eop-ak 后及 Signature 都有一个空格) header 排序以分号“;” 拼接 例子所述：你待签名的字符串 header 顺序是 eop-date 和 host; 那么 你加到 header 里的值就是 Eop-Authorization: ctyun-eop-ak Headers=eop-date;host Signature=xad01/ada

由上得到 Eop-Authorization，然后将数据整合成 HEADER 放在 http_client 内，发出即可。

http_client 所需请求头部如下:

Eop-Authorization: ctyun-eop-ak Headers= ctyun-eop-request-id;eop-date
Signature=xad01/ada

eop-date:20211221T163614Z

ctyun-eop-request-id: 123456789

(注: 若需要进行签名的 Header 不止默认的 ctyun-eop-request-id 和 eop-date, 需要在 http_client 的请求头部中加上, 并且 Eop-Authorization 中也需要增加)

4.1.4 Python 调用示例

```
import hmac
import base64
import hashlib
import json
import time
import uuid
import requests
from urllib.parse import urlparse

def sha256(content):
    x = hashlib.sha256()
    x.update(content.encode())
    return x.hexdigest().upper()

def hmac_sha256(key, content):
    sign = hmac.new(key, content, digestmod="sha256").digest()
    ret = base64.b64encode(sign)
    return ret

# 计算签名
def get_signature(ak, sk, app_key, params):
    # 创建待签名字符串
```



```
# 一、header 部分
```

```
# 主要包括 3 个 header 需要作为签名内容: appkey、ctyun-eop-request-id、eop-date
```

```
# 1. 首先通过 uuid 生成 ctyun-eop-request-id
```

```
request_id = str(uuid.uuid1())
```

```
# 2. 获取当前时间戳并对时间进行格式化
```

```
now_time = time.localtime()
```

```
eop_date = time.strftime("%Y%m%dT%H%M%S", now_time)
```

```
eop_date_simple = time.strftime("%Y%m%d", now_time)
```

```
# 3. 对 header 部分按照字母顺序进行排序并格式化
```

```
camp_header = "appkey:{0}\nctyun-eop-request-id:{1}\neop-date:{2}\n".format(app_key, request_id, eop_date)
```

```
# 二、query 部分
```

```
# 对 url 的 query 部分进行排序
```

```
parsed_url = urlparse(request_url)
```

```
query = parsed_url.query
```

```
query_params = sorted(query.split("&"))
```

```
after_query = ""
```

```
for query_param in query_params:
```

```
    if len(after_query) < 1:
```

```
        after_query += query_param
```

```
    else:
```

```
        after_query += "&" + query_param
```

```
# 三、body 参数进行 sha256 摘要
```

```
# sha256 body
```

```
content_hash = sha256(json.dumps(params)).lower()
```

```
# 完成创建待签名字符串
```

```
pre_signature = camp_header + "\n" + after_query + "\n" + content_hash
```

```
# 构造动态密钥
```

```
k_time = hmac_sha256(sk.encode("utf-8"), eop_date.encode("utf-8"))
```

```
k_ak = hmac_sha256(base64.b64decode(k_time), ak.encode("utf-8"))
```

```
k_date = hmac_sha256(base64.b64decode(k_ak), eop_date_simple.encode("utf-8"))
```

```
# 签名的使用
```

```
signature = hmac_sha256(base64.b64decode(k_date), pre_signature.encode("utf-8"))

# 将数据整合得到真正的 header 中的内容

sign_header = "{0} Headers=appkey;ctyun-eop-request-id;eop-date Signature={1}".format(ak, signature.decode())

# 返回 request-id eop-date 和 sign_header

return request_id, eop_date, sign_header

# 向服务发送请求

def do_post(url, headers, params):

    response = requests.post(url, data=json.dumps(params), headers=headers)

    try:

        print(response.status_code)

        print(response.json())

    except AttributeError:

        print("请求失败")

if __name__ == '__main__':

    # 请求地址

    request_url = "https://ai-global.ctapi.ctyun.cn/v1/aiop/api/2f6hqix09mv4/face/PERSON/person/detectFaceFromBase64"

    # 官网 accessKey

    ctyun_ak = accessKey

    # 官网 securityKey

    ctyun_sk = 'securityKey'

    # 诸葛 AI 官网-控制台-我的应用中获取的 appKey

    ai_app_key = 'appKey'

    # body 内容从本地文件中获取

    # 打开图片文件

    f = open('test.jpeg', 'rb')

    img_base64 = base64.b64encode(f.read()).decode()

    # body 内容

    params = {"imageContent": img_base64}

    # 调用 get_signature 方法获取签名

    request_id, eop_date, sign_header = get_signature(ctyun_ak, ctyun_sk, ai_app_key, params)

    # 生成请求 header
```

```
# 请求 header

headers = {

    'Content-Type': 'application/json;charset=UTF-8',

    'ctyun-eop-request-id': request_id,

    'appkey': ai_app_key,

    'Eop-Authorization': sign_header,

    'eop-date': eop_date,

    'host': 'ai-global.ctapi.ctyun.cn'

}

print("请求头部:")

print(headers)

# 执行 post 请求

do_post(request_url, headers, params)
```

4.1.5 Java 调用示例

java 的鉴权 sdk 及调用示例

请参考官网附件

ai-api-sign-java-sdk-1.0.0.zip

4.3 API

4.3.1 人脸检测

1、接口描述

用于检测输入图像中的人脸，输出人脸位置坐标

2、请求方法

POST

3、接口要求

- 图片大小限制：图片单张大小小于 2MB
- 图片格式限制：图片格式支持 jpg/jpeg/png/bmp/gif 格式

4、请求 URL

/v1/aiop/api/2f6hqix09mv4/face/PERSON/person/detectFaceFromBase64

5、请求参数

请求头 header 参数

参数	是否必填	参数类型	说明	示例	下级对象
Content-Type	是	string	json 格式	"application/json"	
appkey	是	string	诸葛 AI 应用 appkey	"562b89493b1a40e1b97ea05e50d8170"	

请求体 body 参数

参数	是否必填	参数类型	说明	示例	下级对象
imageContent	是	string	传入图片的 base64 编码，图片使用常规的 base64 编码方式，编码后，不包含前缀，剔除前缀例如 "data:image/jpeg;base64,"	-	

6、请求代码示例

```
Curl -X POST "https://ai-global.ctapi.ctyun.cn/v1/aiop/api/2f6hqix09mv4/face/PERSON/person/detectFaceFromBase64"  
-H "Content-Type: application/json"  
-H "ctyun-eop-request-id:33dfa732-b27b-464f-b15a-21ed6845afd5"  
-H "appkey:XXX"  
-H "Eop-Authorization:XXX"  
-H "eop-date:20211109T104641Z"  
-H "host:ai-global.ctapi.ctyun.cn"  
--data '{"imageContent":"AAAAAAAAA...."}'  
复制
```

7、返回值说明

请求成功返回响应参数

参数	是否必填	参数类型	说明	示例	下级对象
code	是	string	返回状态，返回 0 表示成功，返回错误代码参考下面的错误	"0"	

参数	是否必填	参数类型	说明	示例	下级对象
			代码列表。		
message	是	string	如果 code 为 0，返回 success；如果 code 非 0，则返回对应的可读错误信息。	"success"	
result	是	object	返回的人脸检测结果对象		result

表 result

参数	是否必填	参数类型	说明	示例	下级对象
face_num	是	int	图片中人脸的数量	-	
face_list	是	list	每个人脸的详细信息	-	
face_list[].face_loaction	是	object	人脸所处位置	-	

请求失败返回响应参数

参数	是否必填	参数类型	说明	示例	下级对象
code	是	string	错误码，放置 API 对应的错误码	"4101"	
message	是	string	请求失败时返回值固定为"error"	"error"	
details	是	string	返回对应的错误信息	"请求内容错误"	

8、返回值示例

请求成功返回值示例

```
{
  "code": 0,
  "message": "success",
  "result": {
    "face_num" : 1,
    "face_list" : [{
      "face_location" : {
        "top" : 36,
        "left" : 48,
        "width" : 58,
        "height" : 72
      }
    }]
  }
}
```

复制

请求失败返回值示例

```
{
  "code": "4101",
```

```
"message": "error",  
"details": "请求内容错误"  
}
```

复制

9、状态码

状态码	描述
200	表示请求成功

10、错误码说明

4 位错误码。4 开头为业务错误码，5 开头为服务错误码。

错误码	错误信息	错误描述
4101	请求内容错误	传入内容为空，或者传入的参数名错误
4102	请求参数格式错误	参数格式不满足要求，如请求参数字段类型错误等
4103	图片大小超过 2M	图片大小超过 2M
4104	图片解码失败	图片为空，base64 编码内容有误，或图片格式不支持
4105	未检测到人脸	上传图片中不包含人脸

11、base64 编码规则：使用常规的 safe base64 编码方式

- python 中推荐使用 `base64.urlsafe_b64encode()` 函数进行编码。
- java 中推荐使用 `BASE64.getUrlEncoder().encodeToString()` 函数进行编码。

1、接口描述

用于自动检测图片中的安全帽穿戴情况，并给出图像中安全帽的颜色及位置信息。

2、请求方法

POST

3、接口要求

- 目前仅支持 **base64** 编码方式请求，暂不支持 **url** 方式请求。一次请求仅支持一条数据。
- 图片格式限制：目前仅支持 **png、jpg、jpeg、bmp、gif、tif** 格式。
- 图片大小限制：**base64** 编码后图片大小不超过 **5 MB**。

4、请求 URL

/v1/aiop/api/2f3uquanbrpc/SafeHat/base/hat_detect

5、请求参数

请求头 header 参数

参数	是否必填	参数类型	说明	示例	下级对象
Content-Type	是	string	json 格式	"application/json"	
appkey	是	string	诸葛 AI 应用 appkey	"562b89493b1a40e1b97ea05e50dd8170"	

请求体 body 参数

参数	是否必填	参数类型	说明	示例	下级对象
image	是	string	一张图像 Base64 编码字符串。注意：图片需要使用常规 safe base64 编码方式，不包含前缀 "data:img/jpg;base64,"	"_9j_4AAQSkZJRg..."	

6、请求代码示例

```
Curl -X POST  
"https://ai-global.ctapi.ctyun.cn/v1/aiop/api/2f3uquanbrpc/SafeHat/base/hat_detect"
```

```
-H "Content-Type: application/json"
-H "ctyun-eop-request-id:33dfa732-b27b-464f-b15a-21ed6845afd5"
-H "appkey:XXX"
-H "Eop-Authorization:XXX"
-H "eop-date:20211109T104641Z"
-H "host:ai-global.ctapi.ctyun.cn"
--data '{"image":"AAAAAAAAA..."}' 复制
```

7、返回值说明

请求成功返回响应参数

参数	是否必填	参数类型	说明	示例	下级对象
code	是	int	返回状态，返回 0 表示成功，返回错误代码参考下面的错误代码列表。	0	
message	是	string	如果 code 为 0，返回 success；如果 code 非 0，则返回对应的可读错误信息。	"success"	
result	是	list	返回安全帽检测到的结果主要包括位置信息、颜色信息以及置信度		result

表 result

参数	是否必填	参数类型	说明	示例	下级对象
result[].box	是	list	车辆的位置信息，4 个值分别对应 top/left/bottom/right	[15, 72, 84, 129]	
result[].classname	是	string	安全帽对应的颜色，支持蓝色，白色，黄色，红色，其他	"blue_hat"	
result[].score	是	float	目标的置信度，在 0 到 1 区间。	0.9998	

请求失败返回响应参数

参数	是否必填	参数类型	说明	示例	下级对象
code	是	string	错误码，放置 API 对应的错误码	"4003"	

参数	是否必填	参数类型	说明	示例	下级对象
message	是	string	请求失败时返回值固定为"error"	"error"	
details	是	string	返回对应的错误信息	"base64 数据处理异常"	

8、返回值示例

请求成功返回值示例

```
{
  "code": 0
  "message": "success",
  "result": [
    {
      "box": [15, 72, 84, 129],
      "classname": "blue_hat",
      "score": 0.9998
    },
    {
      "box": [44, 107, 104, 200],
      "classname": "blue_hat",
      "score": 0.9984
    }
  ]
}
```

复制

请求失败返回值示例

```
{
  "code": 4003,
  "details": "base64 数据处理异常",
  "message": "error"
}
```

复制

9、状态码

状态码	描述
200	表示请求成功。

10、错误码说明

4 位错误码。4 开头为业务错误码，5 开头为服务错误码。

错误码	错误信息	错误描述
4001	请求未包含图片	请求关键词不正确或未正确包含请求数据

错误码	错误信息	错误描述
4002	图片文件大小不符合要求	该文件大小不符合要求，静态图片要求小于 5M
4003	base64 数据处理异常	客户端传递的 base64 格式在处理时发生异常，可以尝试在本地转换 base64 是否有异常
4006	请求数据格式错误	post body 非字典格式

11、base64 编码规则：使用常规的 safe base64 编码方式

- python 中推荐使用 `base64.urlsafe_b64encode()` 函数进行编码。
- java 中推荐使用 `BASE64.getUrlEncoder().encodeToString()` 函数进行编码。

4.3.2 人脸属性识别

1、接口描述

用于检测输入图像中的人脸年龄、性别等属性

2、请求方法

POST

3、接口要求

- 图片大小限制：图片单张大小小于 2MB
- 图片格式限制：图片格式支持 jpg/jpeg/png/bmp/gif 格式

4、请求 URL

`/v1/aiop/api/2f6hw5o5t7gg/face/PERSON/person/detectAgeGenderFromBase64`

5、请求参数

请求头 header 参数

参数	是否必填	参数类型	说明	示例	下级对象
Content-Type	是	string	json 格式	"application/json"	
appkey	是	string	诸葛 AI 应用 appkey	"562b89493b1a40e1b97ea05e50dd8170"	

请求体 body 参数

参数	是否必填	参数类型	说明	示例	下级对象
imageContent	是	string	传入图片的 base64 编码，图片使用常规的 base64 编码方式，编码后，不包含前缀，剔除前缀例如 "data:image/jpeg;base64,"	-	

6、请求代码示例

```
Curl -X POST "https://ai-global.ctapi.ctyun.cn/v1/aiop/api/2f6hw5o5t7gg/face/PERSON/person/detectAgeGenderFromBase64"
-H "Content-Type: application/json"
-H "ctyun-eop-request-id:33dfa732-b27b-464f-b15a-21ed6845afd5"
-H "appkey:XXX"
-H "Eop-Authorization:XXX"
```

```
-H "eop-date:20211109T104641Z"
-H "host:ai-global.ctapi.ctyun.cn"
--data '{"imageContent":"AAAAAAAAA...."}'
```

复制

7、返回值说明

请求成功返回响应参数

参数	是否必填	参数类型	说明	示例	下级对象
code	是	string	返回状态，返回 0 表示成功，返回错误代码参考下面的错误代码列表。	"0"	
message	是	string	如果 code 为 0，返回 success；如果 code 非 0，则返回对应的可读错误信息。	"success"	
result	是	object	返回的人脸属性识别结果对象		result

表 result

参数	是否必填	参数类型	说明	示例	下级对象
face_num	是	int	图片中人脸的数量	-	

参数	是否必填	参数类型	说明	示例	下级对象
face_list	是	list	每个人脸的详细信息	-	
face_list[].face_loaction	是	object	人脸所处位置	-	
face_list[].gender	是	string	人脸的性别属性	-	
face_list[].age	是	string	人脸的年龄属性	-	

请求失败返回响应参数

参数	是否必填	参数类型	说明	示例	下级对象
code	是	string	错误码，放置 API 对应的错误码	"4101"	
message	是	string	请求失败时返回值固定为"error"	"error"	
details	是	string	返回对应的错误信息	"请求内容错误"	

8、返回值示例

请求成功返回值示例

```
{
```

```
{
  "code": 0,
  "message": "success",
  "result": {
    "face_num": 1,
    "face_list": [{
      "face_location": {
        "top": 36,
        "left": 48,
        "width": 58,
        "height": 72
      },
      "gender": "Male",
      "age": "23"
    }]
  }
}
```

复制

请求失败返回值示例

```
{
  "code": "4101",
  "message": "error",
  "details": "请求内容错误"
}
```

复制

9、状态码

状态码	描述
200	表示请求成功

10、错误码说明

4 位错误码。4 开头为业务错误码，5 开头为服务错误码。

错误码	错误信息	错误描述
4101	请求内容错误	传入内容为空，或者传入的参数名错误
4102	请求参数格式错误	参数格式不满足要求，如请求参数字段类型错误等

错误码	错误信息	错误描述
4103	图片大小超过 2M	图片大小超过 2M
4104	图片解码失败	图片为空，base64 编码内容有误，或图片格式不支持
4105	未检测到人脸	上传图片中不包含人脸

11、base64 编码规则：使用常规的 safe base64 编码方式

- python 中推荐使用 `base64.urlsafe_b64encode()` 函数进行编码。
- java 中推荐使用 `BASE64.getUrlEncoder().encodeToString()` 函数进行编码。

4.3.5 人脸比对

1、接口描述

用于检测输入的两张图像中的人脸相似度

2、请求方法

POST

3、接口要求

- 图片大小限制：图片单张大小小于 2MB
- 图片格式限制：图片格式支持 jpg/jpeg/png/bmp/gif 格式

4、请求 URL

/v1/aiop/api/2f7awxekgvls/face/compare/PERSON/person/compareFromBase64

5、请求参数

请求头 header 参数

参数	是否必填	参数类型	说明	示例	下级对象
Content-Type	是	string	json 格式	"application/json"	
appkey	是	string	诸葛 AI 应用 appkey	"562b89493b1a40e1b97ea05e50dd8170"	

请求体 body 参数

参数	是否必填	参数类型	说明	示例	下级对象
img1Base64	是	string	人脸比对的第一张图片，图片使用常规的 base64 编码方式，编码后，不包含前缀，剔除前缀例如" data:image/jpeg;base64,"	-	
img2Base64	是	string	人脸比对的第二张图片，图片使用常规的 base64 编码方式，编码后，不包含前缀，剔除前缀例如" data:image/jpeg;base64,"	-	

6、请求代码示例

```
Curl -X POST "https://ai-global.ctapi.ctyun.cn/v1/aiop/api/2f7awxekgvls/face/compare/PERSON/person/compareFromBase64"
-H "Content-Type: application/json"
-H "ctyun-eop-request-id:33dfa732-b27b-464f-b15a-21ed6845afd5"
-H "appkey:XXX"
-H "Eop-Authorization:XXX"
-H "eop-date:20211109T104641Z"
-H "host:ai-global.ctapi.ctyun.cn"
--data '{"img1Base64':"AAAAA...',"img2Base64':"BBBBBBBBB..."}'
```

复制

7、返回值说明

请求成功返回响应参数

参数	是否必填	参数类型	说明	示例	下级对象
code	是	string	返回状态，返回 0 表示成功，返回错误代码参考下面的错误代码列表。	"0"	
message	是	string	如果 code 为 0，返回 success；如果 code 非 0，则返回对应的可读错误信息。	"success"	
result	是	float	两张人脸的相似度	0.99999	result

请求失败返回响应参数

参数	是否必填	参数类型	说明	示例	下级对象
code	是	string	错误码，放置 API 对应的错误码	"4101"	
message	是	string	请求失败时返回值固定为"error"	"error"	
details	是	string	返回对应的错误信息	"请求内容错误"	

8、返回值示例

请求成功返回值示例

```
{
  "code": 0,
  "message": "success",
  "result": 0.99999
}
```

复制

请求失败返回值示例

```
{
  "code": "4101",
  "message": "error",
  "details": "请求内容错误"
}
```

复制

9、状态码

状态码	描述
200	表示请求成功

10、错误码说明

4 位错误码。4 开头为业务错误码。

错误码	错误信息	错误描述
4101	请求内容错误	传入内容为空，或者传入的参数名错误
4102	请求参数格式错误	参数格式不满足要求，如请求参数字段类型错误等
4103	图片一大小超过 2M	图片一大小超过 2M
	图片二大小超过 2M	图片二大小超过 2M
4104	图片一解码失败	图片为空，base64 编码内容有误，或图片格式不支持
	图片二解码失败	图片为空，base64 编码内容有误，或图片格式不支持
4105	图片一未检测到人脸	上传图片一中不包含人脸
4106	图片二未检测到人脸	上传图片二中不包含人脸

11、base64 编码规则：使用常规的 safe base64 编码方式

- python 中推荐使用 `base64.urlsafe_b64encode()` 函数进行编码。
- java 中推荐使用 `BASE64.getUrlEncoder().encodeToString()` 函数进行编码。

4.3.5 人脸活体检测

1、接口描述

用于检测输入图像中的人脸是否为活体

2、请求方法

POST

3、接口要求

- 图片大小限制：图片单张大小小于 2MB
- 图片格式限制：图片格式支持 jpg/jpeg/png/bmp/gif 格式

4、请求 URL

/v1/aiop/api/2hfksnibjaos/face-fas-action/person/detectFasFromBase64

5、请求参数

请求头 header 参数

参数	是否必填	参数类型	说明	示例	下级对象
Content-Type	是	string	json 格式	"application/json"	
appkey	是	string	诸葛 AI 应用	"562b89493b1a40e1b97ea05e50dd8170"	

参数	是否必填	参数类型	说明	示例	下级对象
			appkey		

请求体 body 参数

参数	是否必填	参数类型	说明	示例	下级对象
imageContent	是	string	传入图片的 base64 编码，图片使用常规的 base64 编码方式，编码后，不包含前缀，剔除前缀例如 "data:image/jpeg;base64,"	-	

6、请求代码示例

```
Curl -X POST "https://ai-global.ctapi.ctyun.cn/v1/aiop/api/2hfksnibjaos/face-fas-action/person/detectFasFromBase64"
-H "Content-Type: application/json"
-H "ctyun-eop-request-id:33dfa732-b27b-464f-b15a-21ed6845afd5"
-H "appkey:XXX"
-H "Eop-Authorization:XXX"
-H "eop-date:20211109T104641Z"
-H "host:ai-global.ctapi.ctyun.cn"
--data '{"imageContent":"AAAAAAAAA...."}'
```

复制

7、返回值说明

请求成功返回响应参数

参数	是否必填	参数类型	说明	示例	下级对象
code	是	string	返回状态，返回 0 表示成功，返回错误代码参考下面的错误代码列表。	"0"	
message	是	string	如果 code 为 0，返回 success；如果 code 非 0，则返回对应的可读错误信息。	"success"	
result	是	object	返回的人脸活体检测结果对象		result

表 result

参数	是否必填	参数类型	说明	示例	下级对象
face_num	是	int	图片中人脸的数量	-	

参数	是否必填	参数类型	说明	示例	下级对象
face_list	是	list	每个人脸的详细信息	-	
face_list[].face_loaction	是	object	人脸所处位置	-	
face_list[].FaceAntiSpoofing	是	string	活体人脸/非活体人脸/人脸清晰度差，无法判断	-	

请求失败返回响应参数

参数	是否必填	参数类型	说明	示例	下级对象
code	是	string	错误码，放置 API 对应的错误码	"4101"	
message	是	string	请求失败时返回值固定为"error"	"error"	
details	是	string	返回对应的错误信息	"请求内容错误"	

8、返回值示例

请求成功返回值示例

```
{
  "code": 0,
  "message": "success",
  "result": {
    "face_num": 1,
    "face_list": [{
      "face_location": {
        "top": 36,
        "left": 48,
        "width": 58,
        "height": 72
      },
      "FaceAntiSpoofing": "活体人脸/非活体人脸/人脸清晰度较差, 无法判断",
    }]
  }
}
```

复制

请求失败返回值示例

```
{
  "code": "4101",
  "message": "error",
  "details": "请求内容错误"
}
```

复制

9、状态码

状态码	描述
200	表示请求成功

10、错误码说明

4 位错误码。4 开头为业务错误码，5 开头为服务错误码。

错误码	错误信息	错误描述
4101	请求内容错误	传入内容为空，或者传入的参数名错误
4102	请求参数格式错误	参数格式不满足要求，如请求参数字段类型错误等
4103	图片大小超过 2M	图片大小超过 2M
4104	图片解码失败	图片为空，base64 编码内容有误，或图片格式不支持
4105	未检测到人脸	上传图片中不包含人脸
5000	服务内部错误	接口服务出现未知错误

11、base64 编码规则：使用常规的 safe base64 编码方式

- python 中推荐使用 `base64.urlsafe_b64encode()` 函数进行编码。
- java 中推荐使用 `BASE64.getUrlEncoder().encodeToString()` 函数进行编码。

4.3.5 是否戴口罩识别

1、接口描述

用于检测输入图像中的人脸是否戴口罩

2、请求方法

POST

3、接口要求

- 图片大小限制：图片单张大小小于 2MB
- 图片格式限制：图片格式支持 jpg/jpeg/png/bmp/gif 格式

4、请求 URL

/v1/aiop/api/2f6hycj3a9z4/face/PERSON/person/detectMaskFromBase64

5、请求参数

请求头 header 参数

参数	是否必填	参数类型	说明	示例	下级对象
Content-Type	是	string	json 格式	"application/json"	
appkey	是	string	诸葛 AI 应用 appkey	"562b89493b1a40e1b97ea05e50dd8170"	

请求体 body 参数

参数	是否必填	参数类型	说明	示例	下级对象
imageContent	是	string	传入图片的 base64 编码，图片使用常规的 base64 编码方式，编码后，不包含前缀，剔除前缀例如 "data:image/jpeg;base64,"	-	

6、请求代码示例

```
Curl -X POST "https://ai-global.ctapi.ctyun.cn/v1/aiop/api/2f6hycj3a9z4/face/PERSON/person/detectMaskFromBase64"  
-H "Content-Type: application/json"  
-H "ctyun-eop-request-id:33dfa732-b27b-464f-b15a-21ed6845afd5"  
-H "appkey:XXX"  
-H "Eop-Authorization:XXX"  
-H "eop-date:20211109T104641Z"  
-H "host:ai-global.ctapi.ctyun.cn"  
--data '{"imageContent":"AAAAAAAAA...."}'  
复制
```

7、返回值说明

请求成功返回响应参数

参数	是否必填	参数类型	说明	示例	下级对象
code	是	string	返回状态，返回 0	"0"	

参数	是否必填	参数类型	说明	示例	下级对象
			表示成功，返回错误代码参考下面的错误代码列表。		
message	是	string	如果 code 为 0，返回 success；如果 code 非 0，则返回对应的可读错误信息。	"success"	
result	是	object	返回的是否戴口罩识别结果对象		result

表 result

参数	是否必填	参数类型	说明	示例	下级对象
face_num	是	int	图片中人脸的数量	-	
face_list	是	list	每个人脸的详细信息	-	
face_list[].face_loaction	是	map	人脸所处位置	-	
face_list[].Mask	是	bool	是否佩戴口罩	-	

参数	是否必填	参数类型	说明	示例	下级对象
			罩		

请求失败返回响应参数

参数	是否必填	参数类型	说明	示例	下级对象
code	是	string	错误码，放置 API 对应的错误码	"4101"	
message	是	string	请求失败时返回值固定为"error"	"error"	
details	是	string	返回对应的错误信息	"请求内容错误"	

8、返回值示例

请求成功返回值示例

```
{
  "code": 0,
  "message": "success",
  "result": {
    "face_num" : 1,
    "face_list" : [{
      "face_location" : {
        "top" : 36,
        "left" : 48,
        "width" : 58,
        "height" : 72
      },
    },
  ],
}
```

```
"Mask" : false
  ]
}
```

复制

请求失败返回值示例

```
{
  "code": "4101",
  "message": "error",
  "details": "请求内容错误"
}
```

复制

9、状态码

状态码	描述
200	表示请求成功

10、错误码说明

4 位错误码。4 开头为业务错误码，5 开头为服务错误码。

错误码	错误信息	错误描述
4101	请求内容错误	传入内容为空，或者传入的参数名错误
4102	请求参数格式错误	参数格式不满足要求，如请求参数字段类型错误等
4103	图片大小超过 2M	图片大小超过 2M
4104	图片解码失败	图片为空，base64 编码内容有误，或图片格式不支持
4105	未检测到人脸	上传图片中不包含人脸

11、base64 编码规则：使用常规的 safe base64 编码方式

- python 中推荐使用 `base64.urlsafe_b64encode()` 函数进行编码。
- java 中推荐使用 `BASE64.getUrlEncoder().toString()` 函数进行编码。

4.3 更新历史

更新日期	更新内容
2020-12-11	第一次正式发布。
2022-10-25	第二次正式发布。本次更新说明如下：修改 API 文档格式：
2022-11-29	第三次正式发布。本次更新说明如下：修改 API 文档格式：
.....

5 常见问题

5.1 计费类

人脸识别支持哪些计费方式？

我们只提供封顶资源包的计费方式。

人脸识别是否支持续订？

可以支持，当已订购的资源包订单即将到期或即将用完时，可通过订购新的资源包进行续订。

人脸识别是否支持退订？

不支持，已购订单内的资源包不支持退订。

5.2 购买类

人脸识别服务如何开通？

通过 API 加入应用，开通服务，可参考[创建人脸识别应用](#)。

人脸识别产品什么时候进入开通状态？

当您支付费用且系统扣款成功后，将自动为您开通服务。

人脸识别服务是否支持代金券付款？

支持。

人脸识别服务是否支持试用？

在天翼云官网订购，可以支持一个包含有限次数的资源包（100/1000/1 万次，根据能力不同）试用。

5.3 操作类

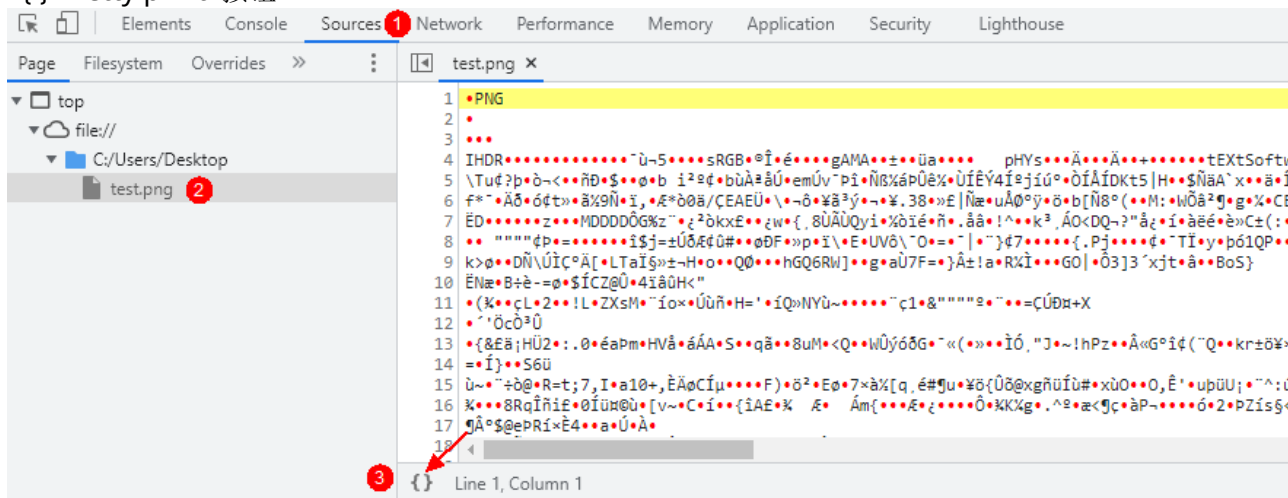
人脸识别在什么时候进入可使用状态？

A：当您支付费用且系统扣款成功后，将自动为您开通服务。

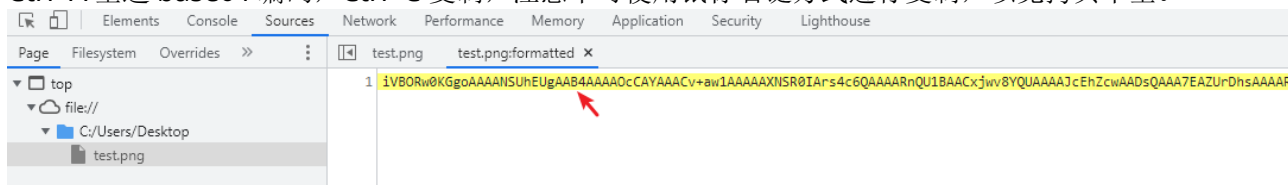
如何获取图片 base64 编码？

1. 将需转码的图片拖拽至 Chrome 浏览器中，浏览器上显示该图片。

- 按下“F12”，在弹出窗口中选择“Sources”，在左侧导航树中选择需要编码的图片所在路径，单击“{ } Pretty print”按钮。



- 图片的 base64 编码显示在右侧界面中，如下图箭头中内容，选中图片的 base64 编码信息，Ctrl+A 全选 base64 编码，Ctrl+C 复制，注意不可使用鼠标右键方式进行复制，以免拷贝不全。



人脸识别服务支持批量识别吗？

OCR 服务只支持调用一次接口识别一张图片，批量识别需要进行二次开发，编码循环调用 API，实现批量调用服务识别图片。

人脸识别的并发是多少？

公有云服务，线上用户资源共享，并发量会根据线上用户的调用情况动态调整。

如遇到突发高峰导致的并发量不够用的情况，您可以尝试以下两种解决方法：

- 通过重试机制，碰到并发错误可以延时一小段时间（如 2-5s）重试请求。
- 后端检查上一请求结果，上一个请求返回后再发送下一次，避免请求过于频繁。
- 如果需要更大的常态性并发需求，请通过天翼云客服联系我们。

5.4 使用限制

人脸识别对于上传的图片是否有要求？

- 图片单张大小不超过 2MB；

(2) 图片格式支持 jpg、png、bmp 等。

人脸识别是 HTTP GET 请求还是 HTTP POST 请求？

HTTP POST 请求。

人脸比对的 API 怎么用？上传几张图片？

人脸对比的 API 主要是对比两张图片中的人脸是否为同一人，因此上传至少上传 2 张人脸图片（即两张图片为一组，可以为多组，进行批量人脸的对比）。

人脸识别的请求图像放置在 http 请求的哪部分？

请求图片应当放置于 http 的 body 中，不能放置于 query 或者 header 中

人脸检测的 api 接口是否支持单次请求多张图片？

接口只允许单张图片请求，不允许图片 list，且按请求中的图片数量来进行计费。