

**天翼云印刷文字识别**

**用户操作手册**

天翼云科技有限公司

**目 录**

[1 产品简介 1](#_Toc19337)

[1.1 产品定义 1](#_Toc25461)

[1.2 功能特性 1](#_Toc31141)

[1.3 产品优势 1](#_Toc27305)

[1.4 应用场景 2](#_Toc21655)

[2 计费说明 4](#_Toc6351)

[2.1 计费模式 4](#_Toc23374)

[2.2 产品续订 5](#_Toc26321)

[2.3 产品退订 5](#_Toc2552)

[3 用户指南 6](#_Toc21344)

[3.1 购买服务 6](#_Toc4794)

[3.2 创建应用及开通服务 8](#_Toc9575)

[3.3. 印刷文字识别控制台 12](#_Toc20708)

[4 API 参考 15](#_Toc25300)

[4.1API 概览 15](#_Toc29438)

[4.1.1 概述 15](#_Toc9983)

[4.1.2 API 概览 15](#_Toc31279)

[4.1.3 状态码 15](#_Toc7416)

[4.2 如何调用 API 17](#_Toc25023)

[4.1.1 终端节点 17](#_Toc17787)

[4.1.2 构造请求 17](#_Toc6083)

[4.1.3 认证鉴权 18](#_Toc11619)

[4.1.4 Python3调用示例 22](#_Toc30788)

[4.3API 24](#_Toc8401)

[4.3.1 通用型OCR 24](#_Toc17998)

[4.3.2 身份证识别 31](#_Toc2140)

[4.3.4 车牌识别 38](#_Toc15055)

[4.3.4 营业执照识别 44](#_Toc18702)

[4.4 更新历史 50](#_Toc16184)

[5 常见问题 51](#_Toc1272)

[5.1 计费类 51](#_Toc23086)

[5.2 购买类 52](#_Toc18843)

[5.3 操作类 53](#_Toc18136)

[5.4 使用限制 55](#_Toc25219)

**1** 产品简介

### 产品定义

印刷文字识别是中国电信云公司自研 A I能力开发平台提供的产品之一，刷文字识别（OCR）提供在线文字识别服务，将图片中的文字识别成可编辑的文本。支持通用类识别、证件类识别、行业类识别等，满足各种客户的识别需求。

### 1.2 功能特性

**通用型 OCR**

通用型 OCR 是适用于多场景、多颜色的OCR 文字检测和识别服务，识别准确率高，识别 速度快。识别支持表格、文档、网络图片等任意格式图片上文字信息的自动化识别，自适应分析各种版面和表格，快速实现各种文档电子化。

**车牌识别**

针对图片中的车牌，进行 OCR 检测，返回检测到的车牌内容及车牌位置坐标。

**身份证识别**

针对图片中的身份证，进行 OCR 检测及后处理，支持返回身份证姓名、住址、身份证号 等信息。

**营业执照识别**

识别营业执照图片的内容，并结构化输出统一社会信用代码、名称、住所等字段。

### 1.3 产品优势

天翼云印刷文字识别产品具有识别精度高，使用场景丰富，服务质量稳定等优势。

**识别精度高**

海量标注样本进行训练，配合国内领先的人工智能算法，针对业务场景优化，打造高精度的印刷文字识别产品。

**数据安全**

少人工干预，降低隐私风险，保护数据资产。

**快速高效**

24h 服务无疲劳，识别速度高于人工 10 倍以上。

**使用便捷**

提供标准 API 接口调用简单，便于被集成。

**使用场景丰富**

依据客户需求，提供多种类别的使用场景，采用深度学习算法。

**服务质量稳定**

依托天翼云平台，提供稳定稳定高可用的服务品质。

**部署方式灵活**

可以提供公共云、私有云、端服务等部署方式，满足客户部署需求。

### 1.4 应用场景

印刷文字识别主要用于办公场景，媒体场景，政务场景，交通场景。

**各类产品的应用场景**

通用 OCR 识别应用场景：文件存档的录入，客户通用文档文字识别。

车牌识别应用场景：停车场及社区出入口，高速公路收费站，城市交通以及公路卡口道路。

卡证识别应用场景：用户注册 ，银行开户，交通出行，身份认证。

各类场景中的可以实现的应用功能

办公场景：文件存档截图的录入，书籍照片的录入，客户通用文档文字识别。

媒体场景：结合图片进行内容审核监管。

政务场景：政务云上各类卡证照的识别。

交通场景：中国大陆车牌的识别。

**2** 计费说明

### 2.1 计费模式

印刷文字识别产品采用封顶资源包的计费方式，一些在售卖的API服务会免费赠送调用额度供测试使用，若有已购买且在有效期内的商品资源包，则可进行抵扣。

目前印刷文字识别的身份证识别、通用型OCR、车牌识别这三个能力处于商用状态，需要付费购买才能使用。

目前印刷文字识别的营业执照识别暂时为限时免费状态，开通即可使用。

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身份证识别的API调用大概为50万次，则可以购买五个10万次/年的调用资源包，若大概为100万次，则可以购买一个100万次/年的调用资源包。

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接口调用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 ：表格价格仅供参考 ，具体购买价格以价格发文为准。

### 2.2 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。

### 2.3 产品退订

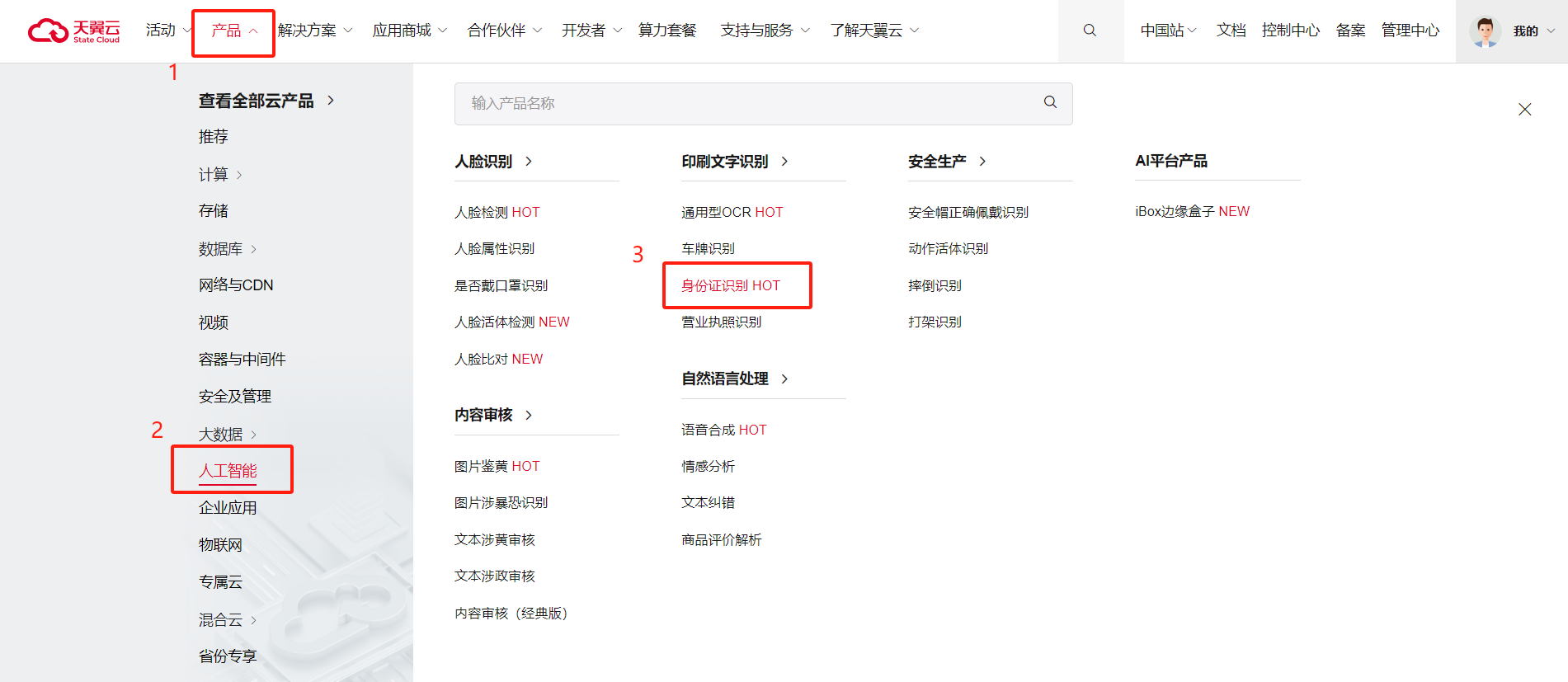
退订说明 ：已购订单内的资源包不支持退订。

**3** 用户指南

### 3.1 购买服务

**1、进入产品详情页**

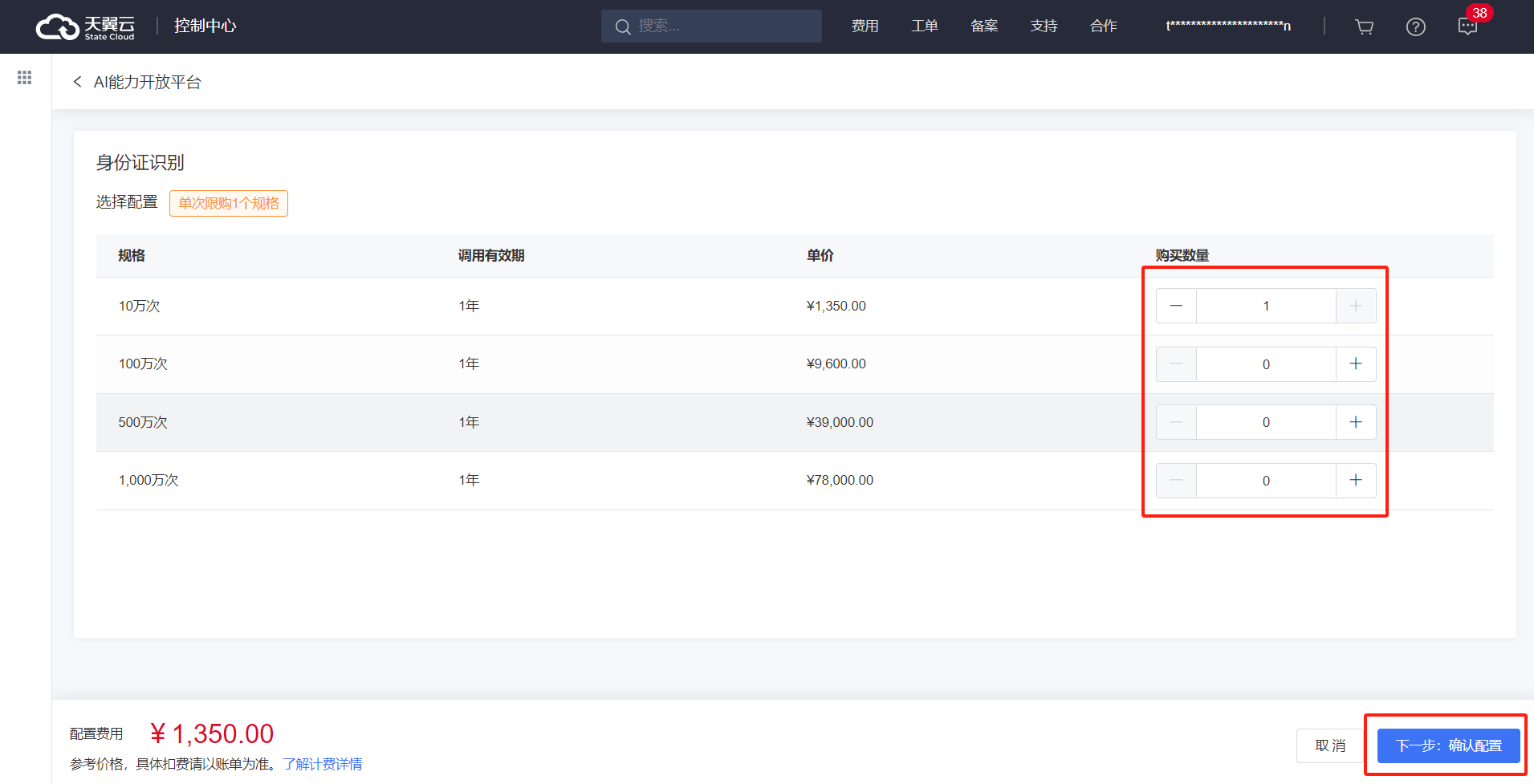
首先，用户使用天翼云网门户账号登录天翼云网门户后，点击菜单栏“产品”项，进入【人工智能】 ， 即可看到印刷文字识别产品相应服务 。 以“ 身份证识别 ”为例， 点击“ 身份证识别 ”名称即 可进入产品详情页。

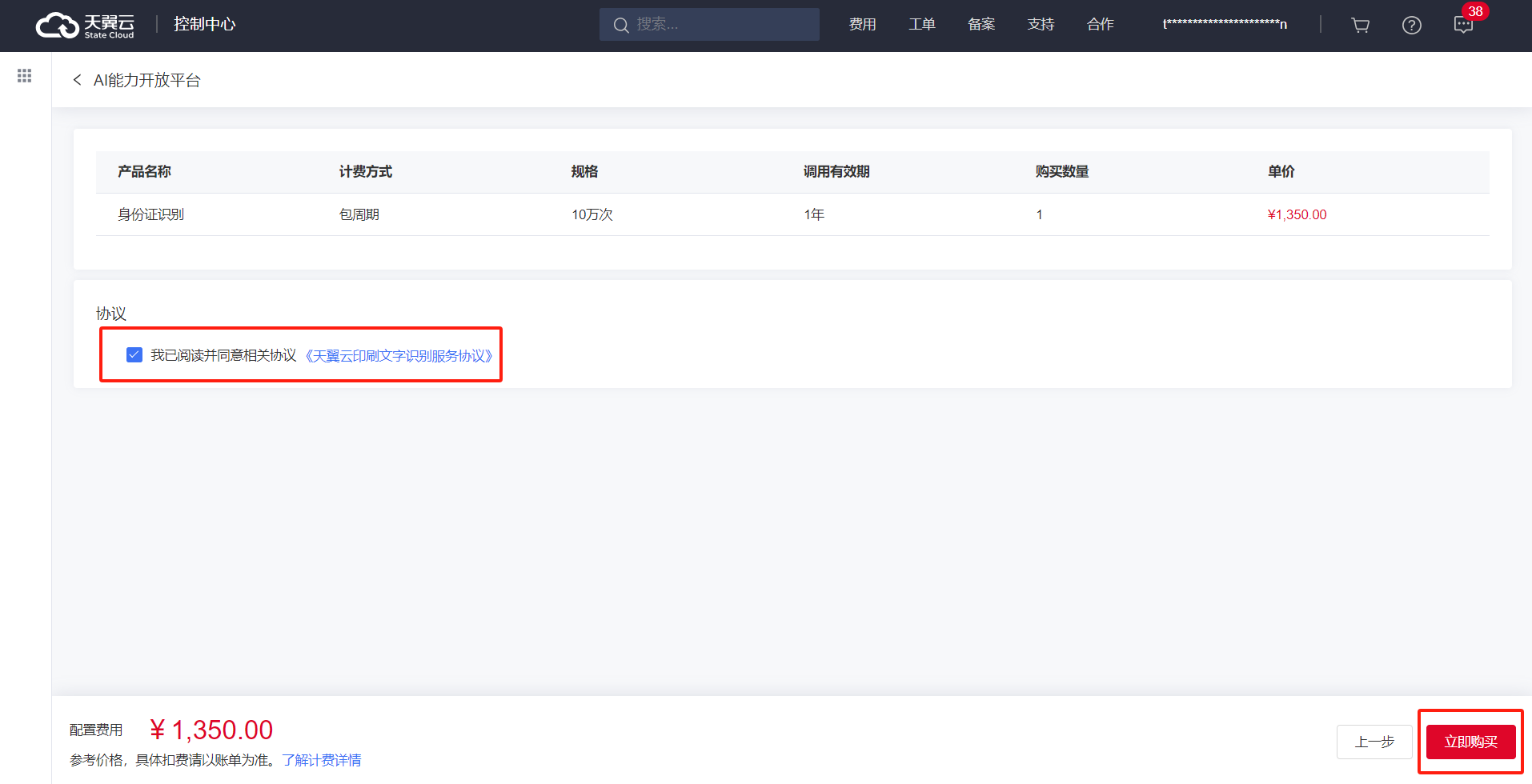


**2.购买资源包**

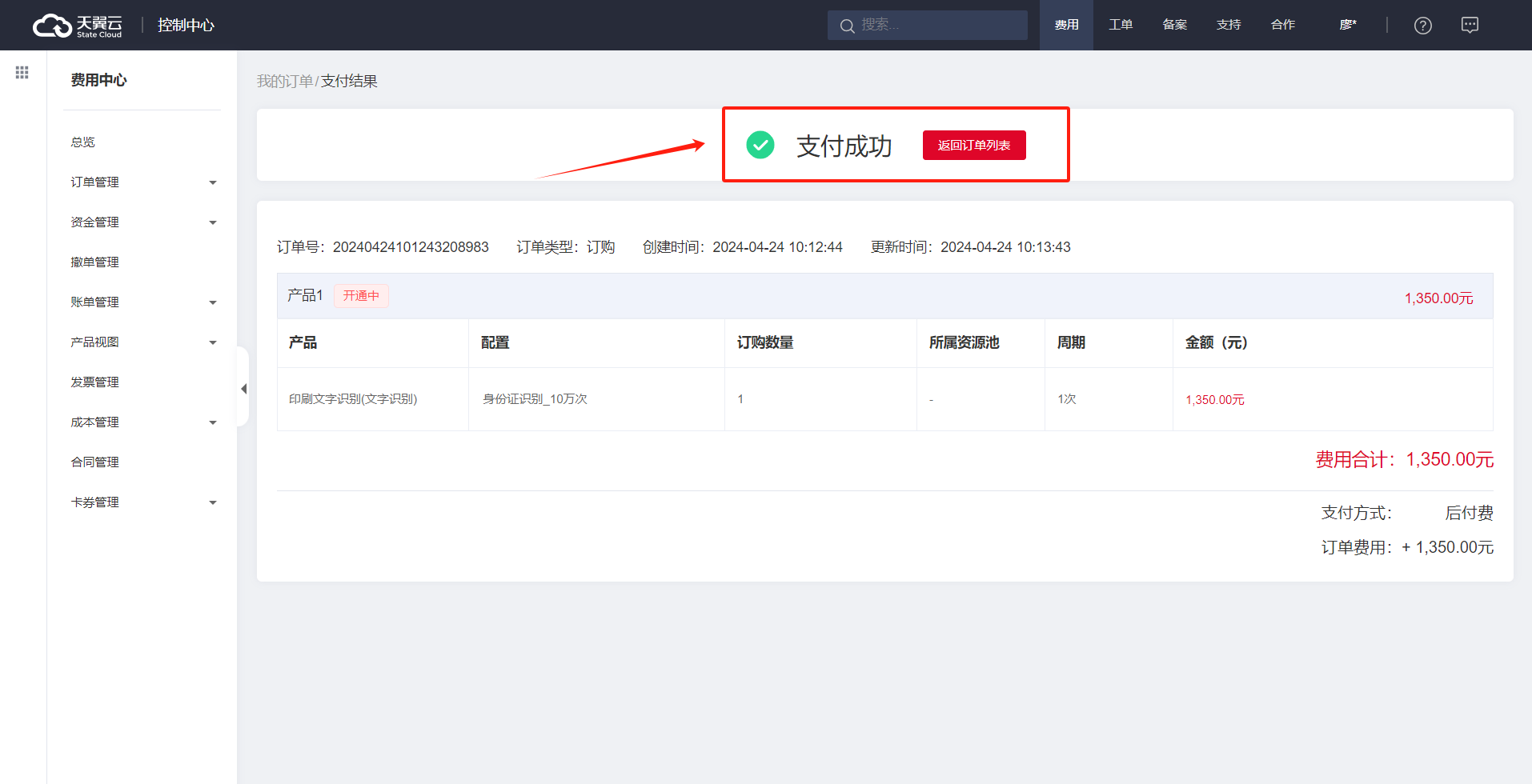
点击【立即购买】选择相应规格，确认订单费用是否准确，费用确认无误后，点击“ 立即支付 ”。会跳转到第三方支付页面进行 支付， 目前支持支付宝 、微信 、 以及银行卡等支付方式。





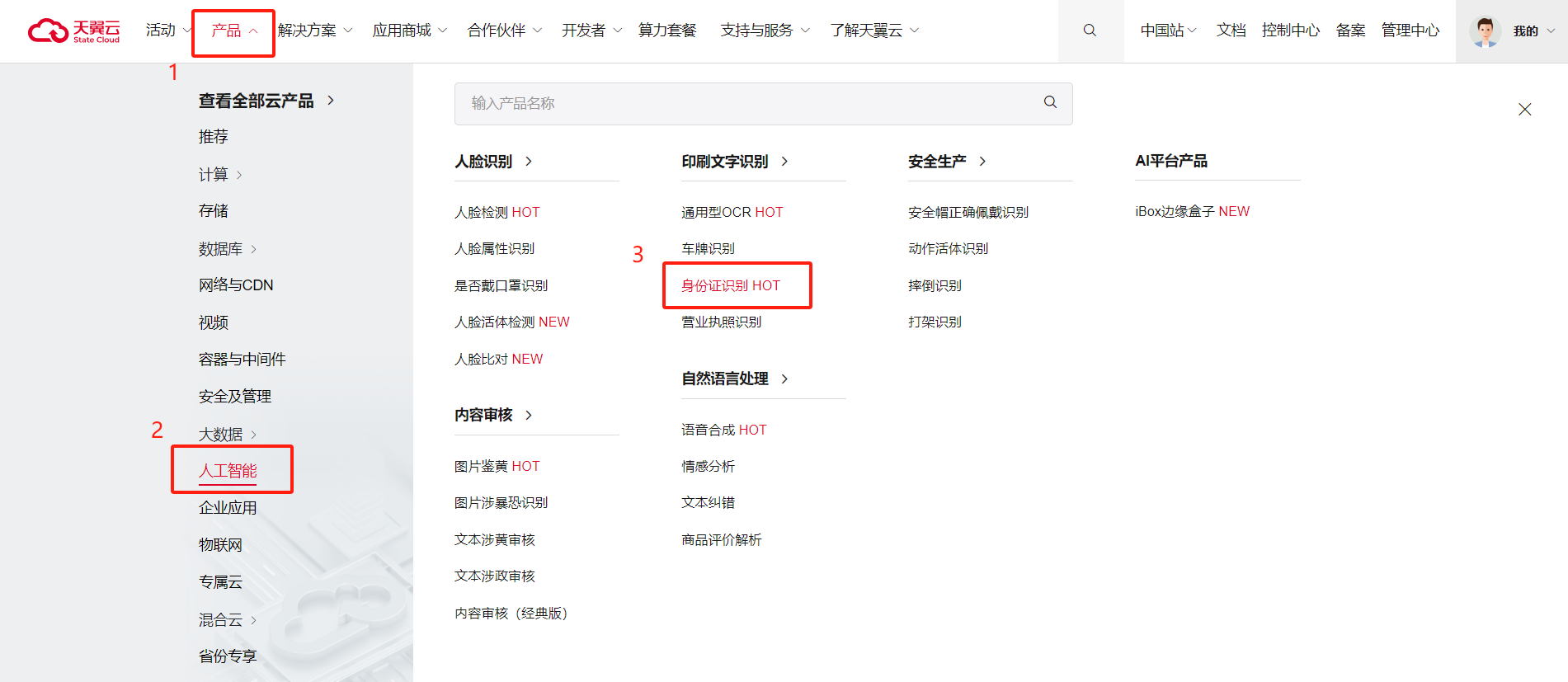






### 3.2 创建应用及开通服务

1.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】 ， 即可看到内容审核产品相应服务 。 以“身份证识别 ”为例， 点击“ 身份证识别”名称即 可进入“ 身份证识别”产品详情页。

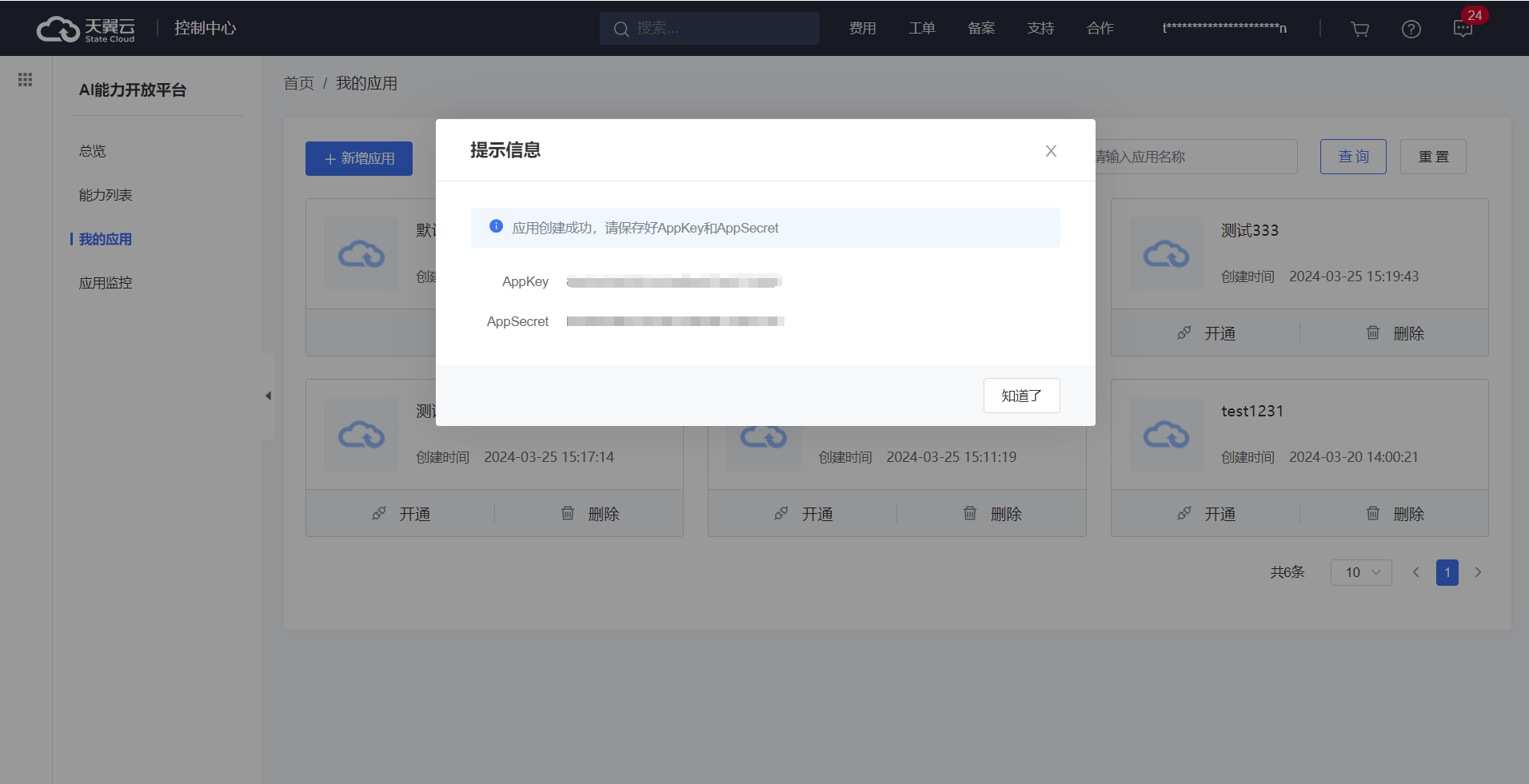


2.点击【立即开通】按钮，进入控制台【我的应用】，点击【新增应用】，页面出现新建应用的弹窗，填写应用名称及应用概述，点击【确定】按钮。

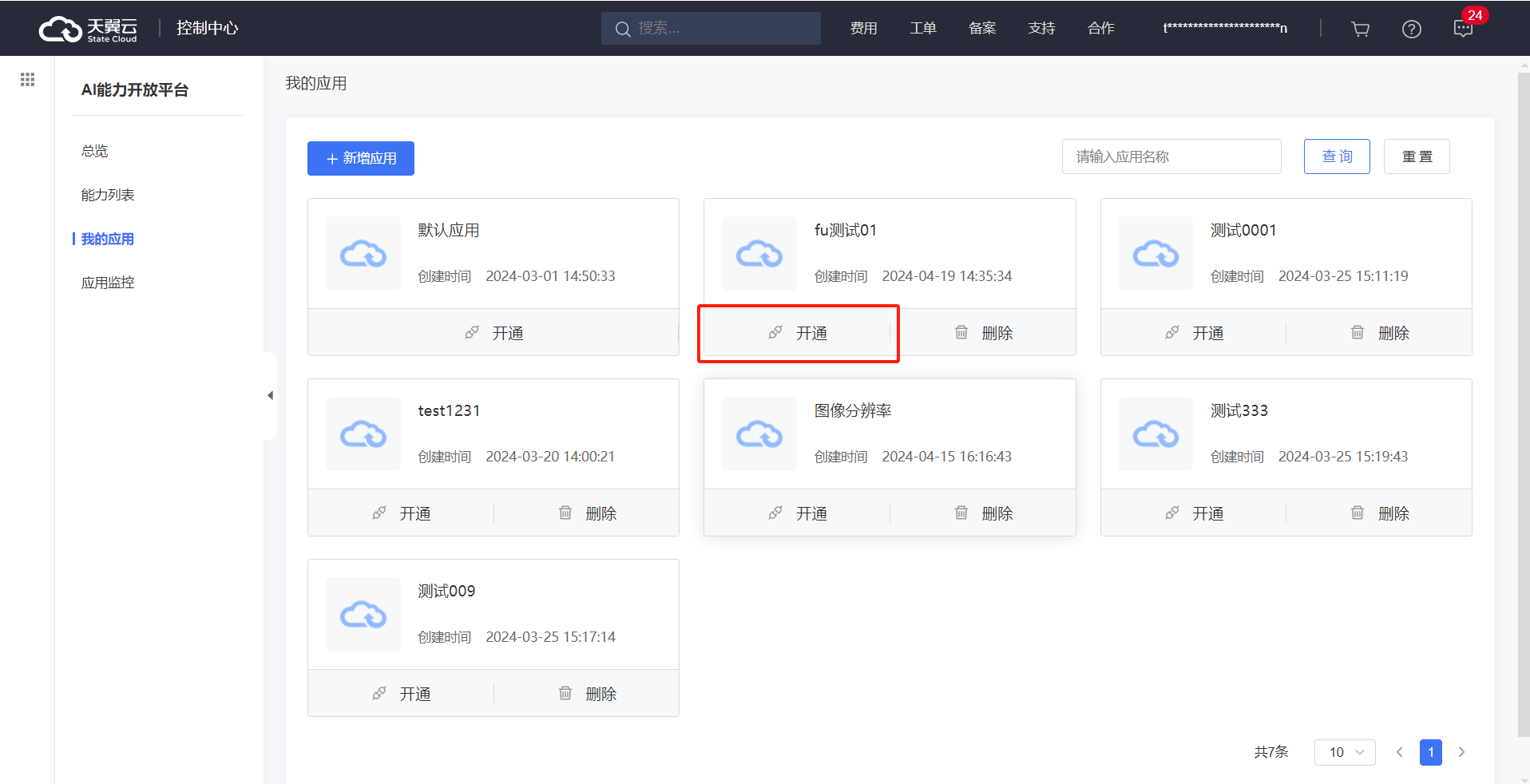




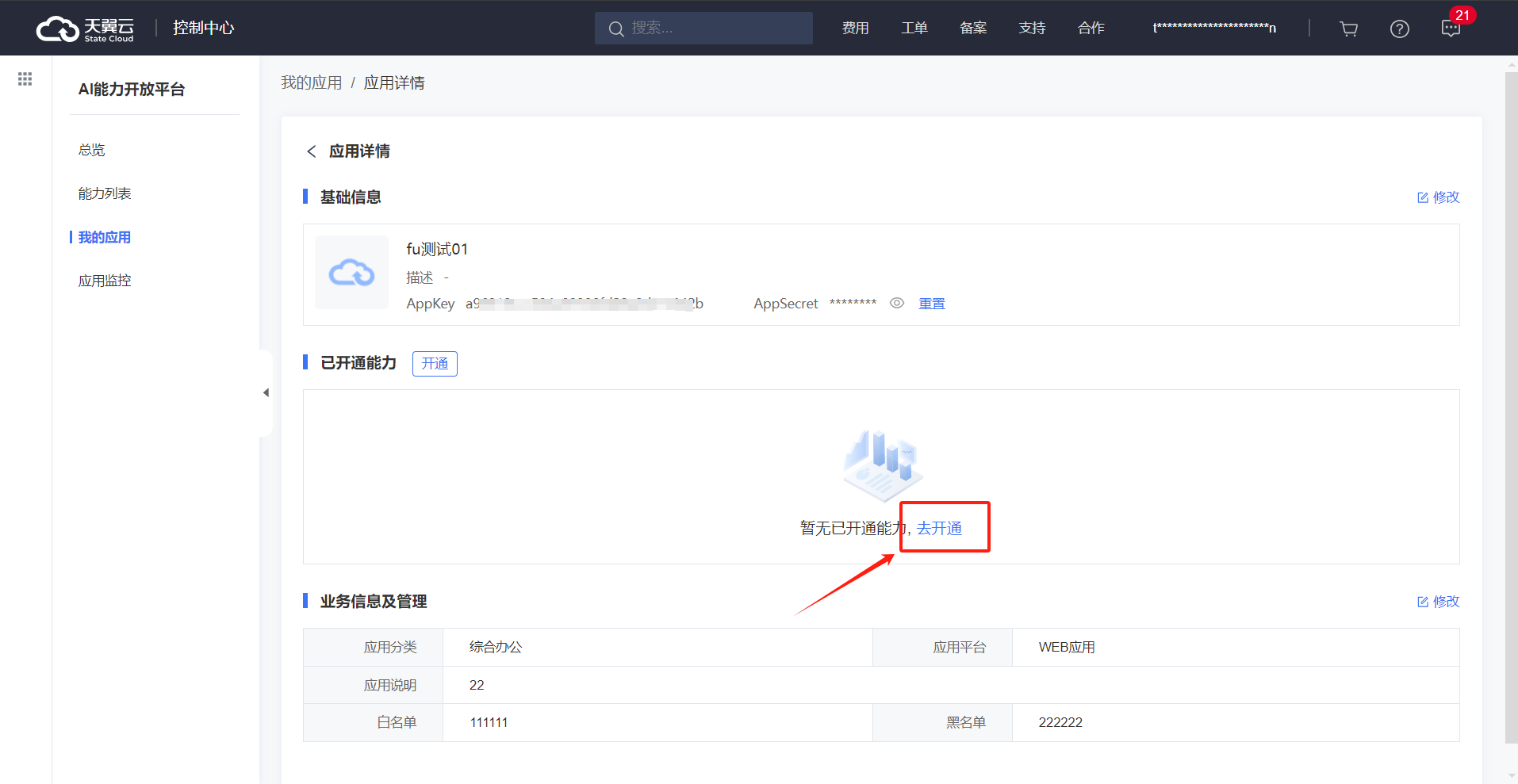
3.再次弹出小弹窗，提示应用创建成功，请保存好AppKey和AppSecret，点击【知道了】即新应用创建成功。

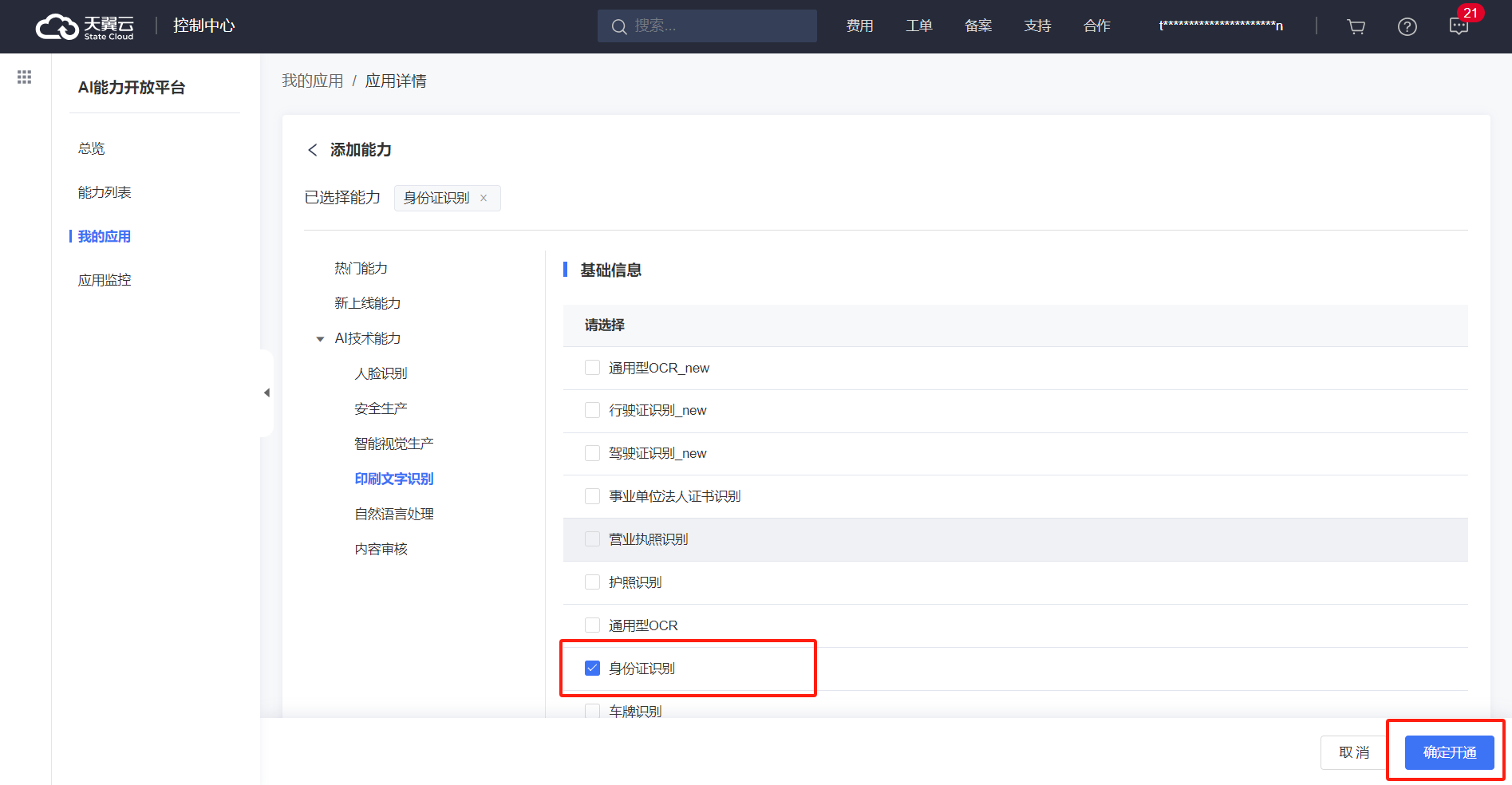


4.根据需要选择应用，点击【开通】，跳转到应用详情页。

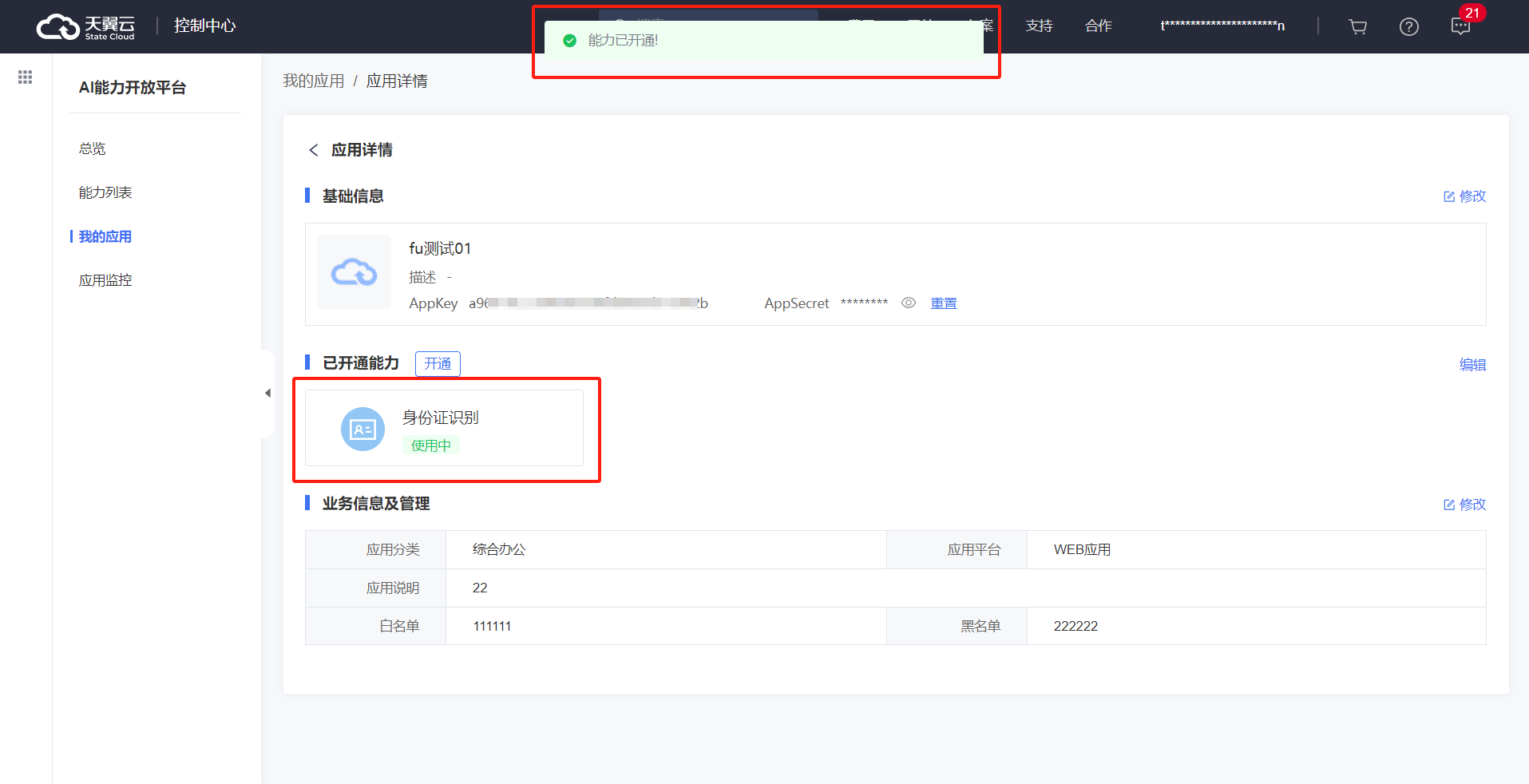


5.点击【去开通】，添加能力。





6.点击确认开通，即反馈【能力已开通】，并返回应用详情，在能力管理中展示了已开通的能力。

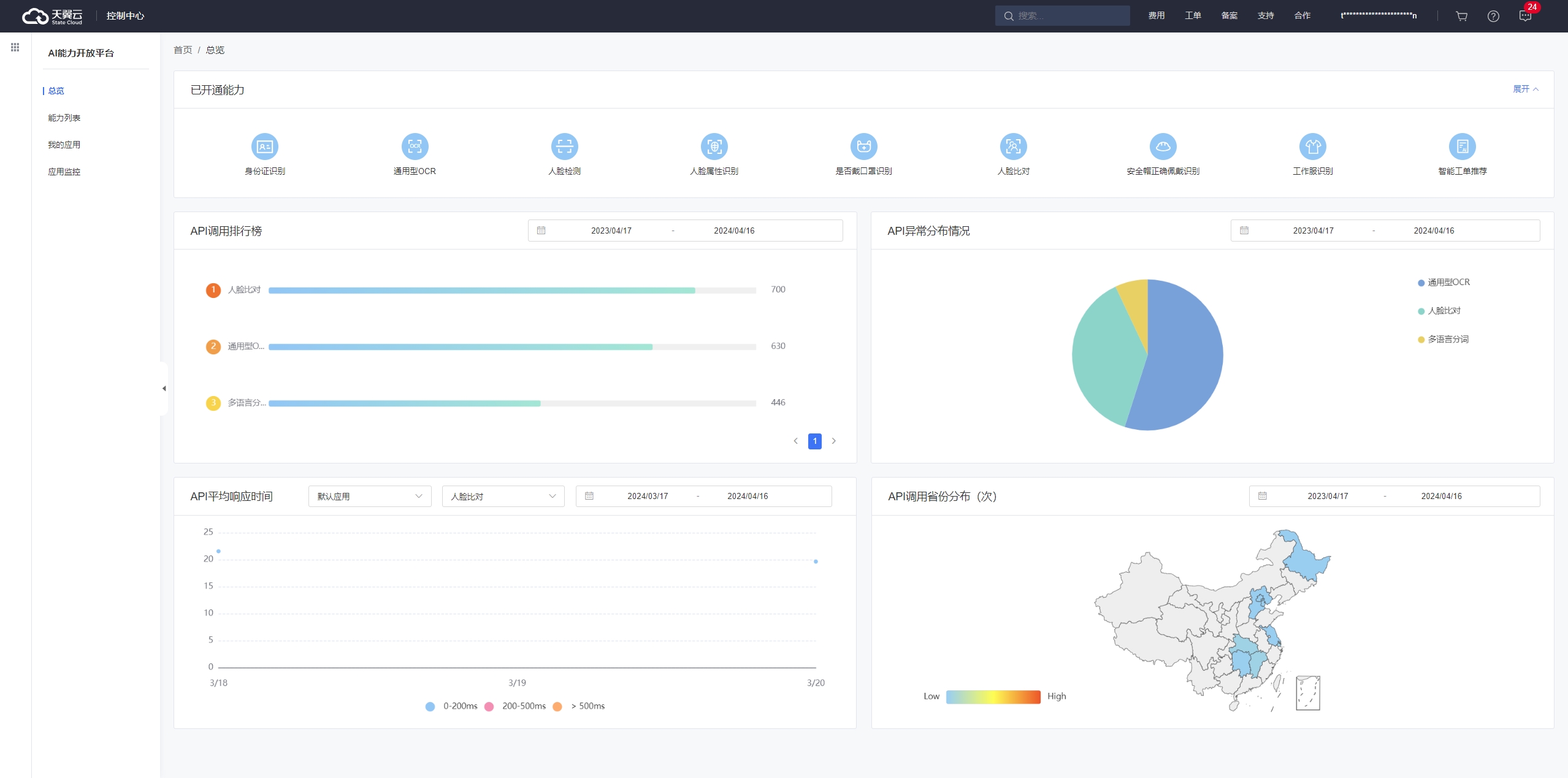


7.点击应用，在应用详情页可以查看对应的AppKey、AppSecret。

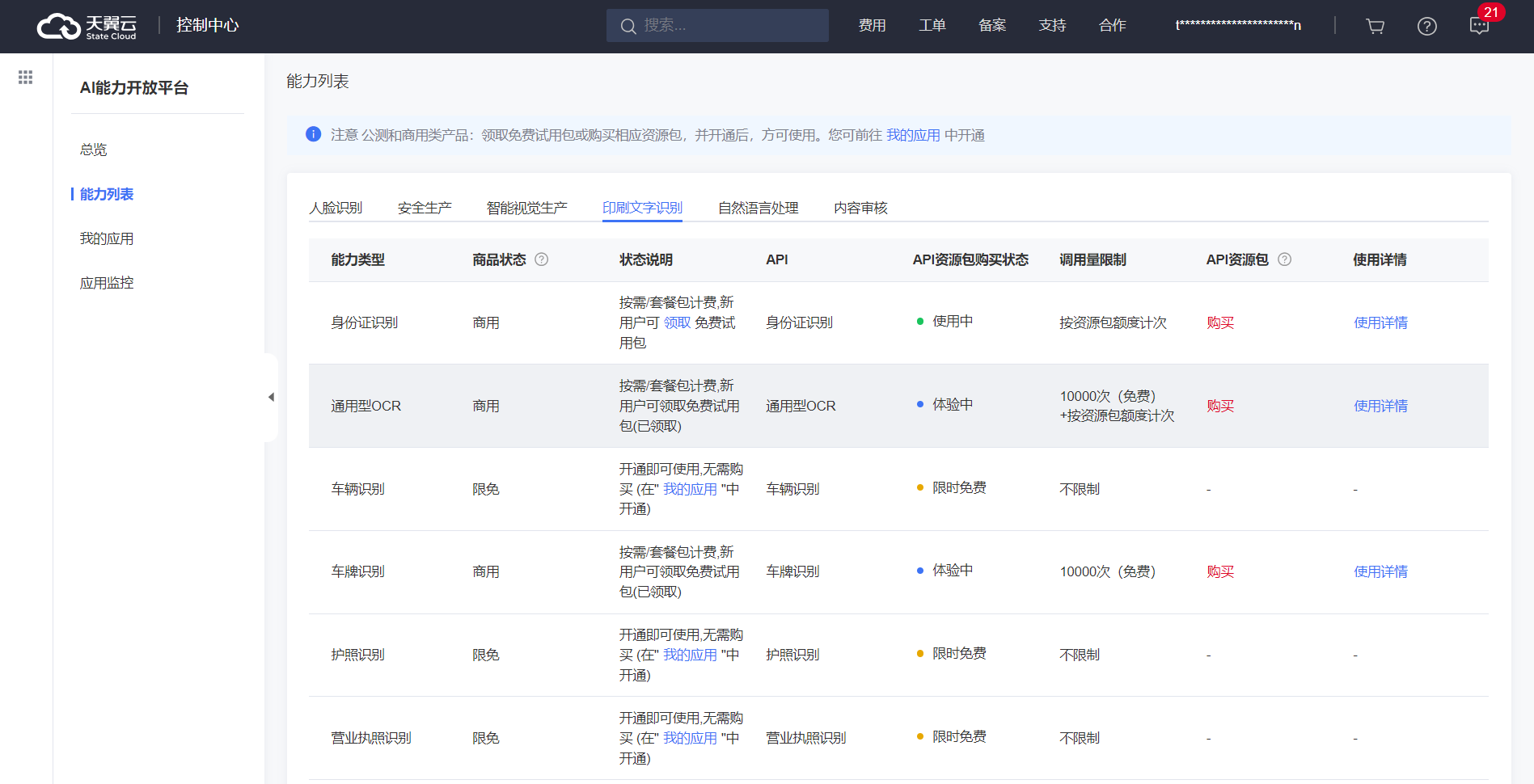


### 3.3. 印刷文字识别控制台

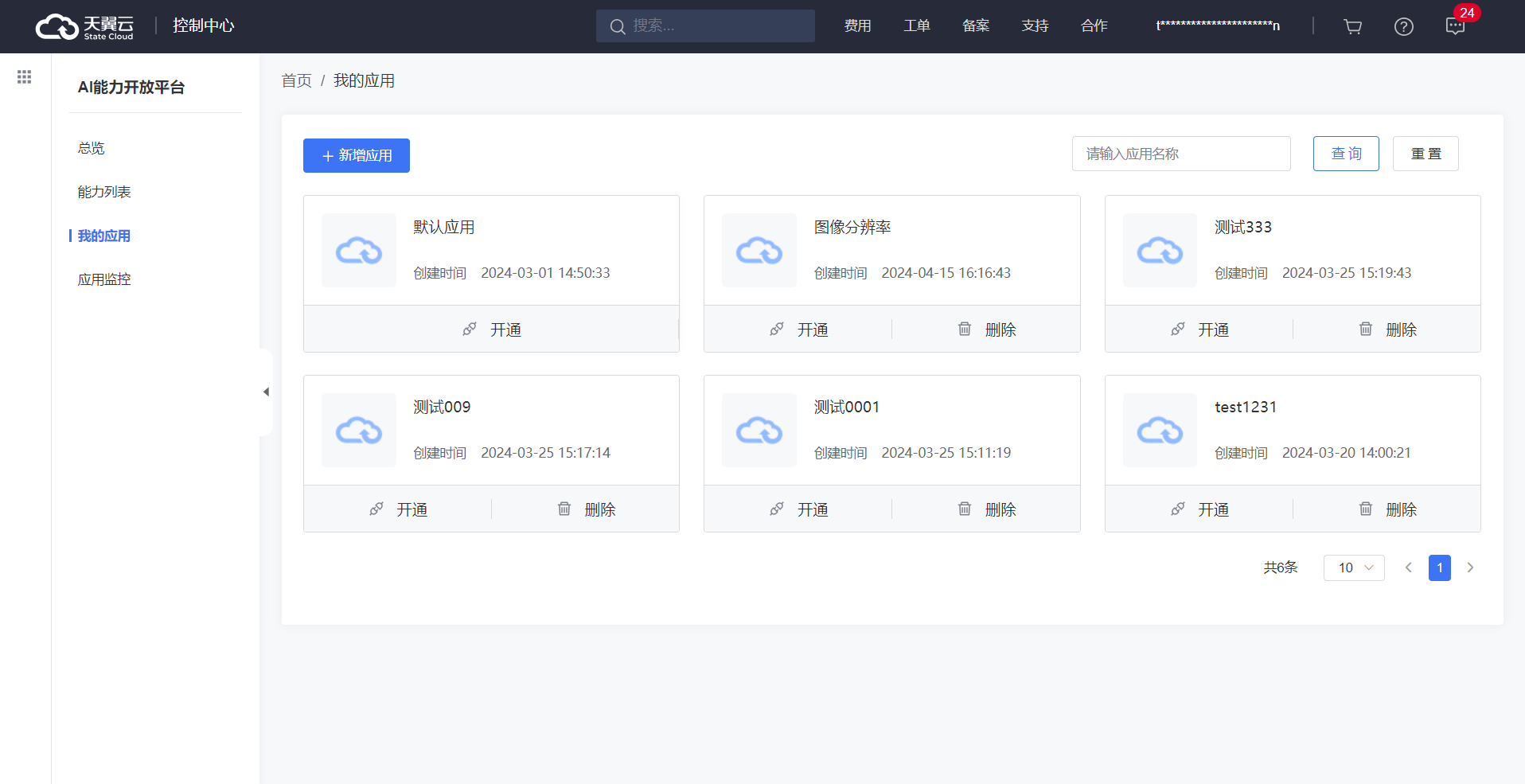
1、入口：点击产品详情页左上角的【管理控制台】，页面跳转到控制台总览页面。

2、总览：点击左侧菜单栏【总览】，可以查看已开通能力、能力调用排行榜以及平均响应时间等内容。

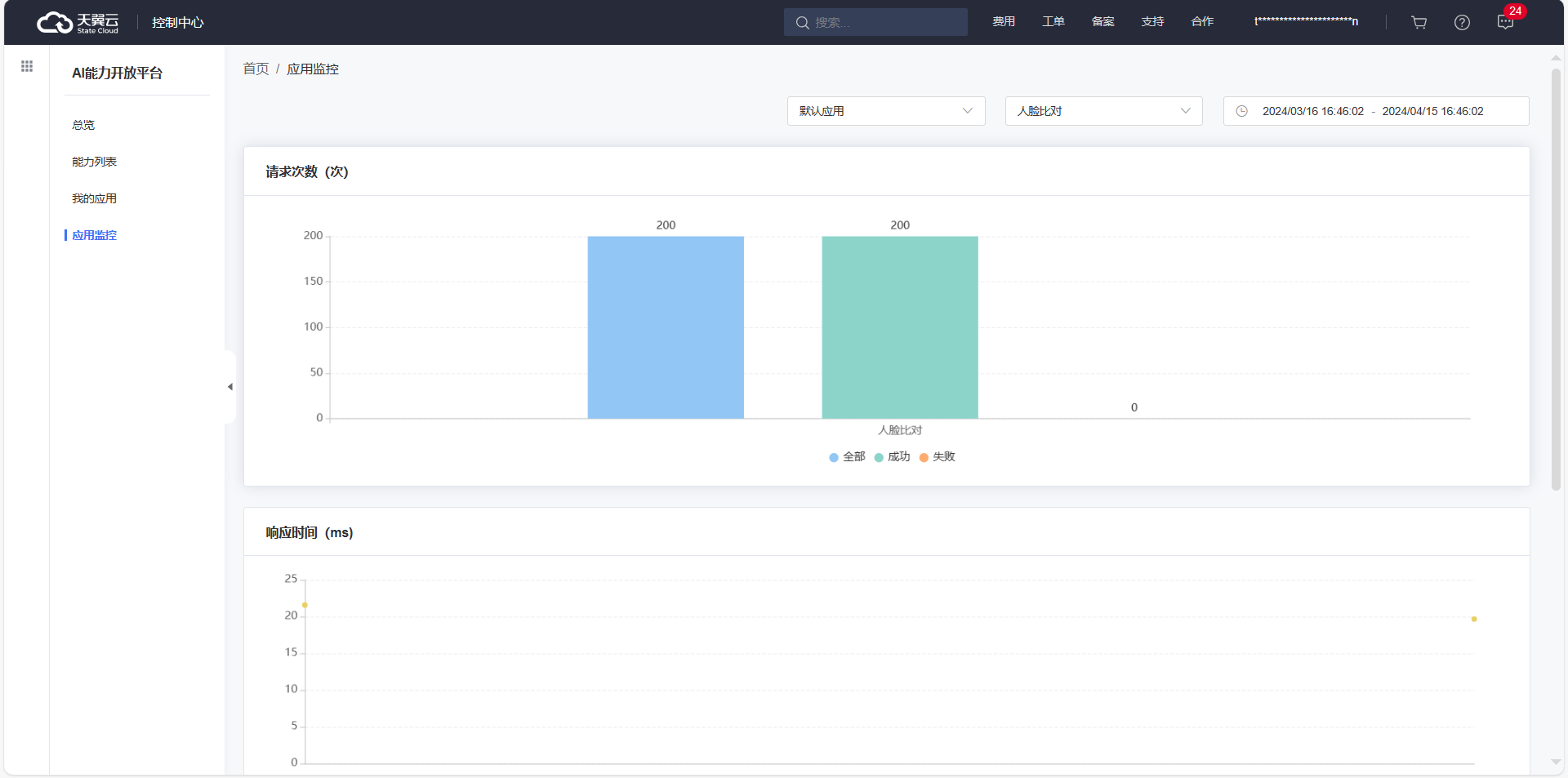
3、能力列表：点击左侧菜单栏【能力列表】，可以查看已有能力。



4、我的应用：点击左侧菜单栏【我的应用】，可以查看已经创建的应用。



5、应用监控：点击左侧菜单栏【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。



**4** API 参考

### 4.1API 概览

#### 4.1.1 概述

本说明提供了印刷文字识别产品 API 的描述、语法、参数说明及示例等内容。

#### 4.1.2 API 概览

|  |  |
| --- | --- |
| **类型** | **描述** |
| 通用型 OCR | 针对图片中的文字，进行 OCR 检测，返回检测到的文字内容及坐标信息 |
| 身份证识别 | 针对图片中的身份证，进行 OCR 检测，返回检测到的姓名、身份证号码等信息 |
| 车牌识别 | 针对图片中的车牌，进行 OCR 检测，返回检测到的车牌内容及车牌位置坐标 |
| 营业执照识别 | 针对图片中的营业执照，进行 OCR 检测，返回检测到的统一社会信用代码、名称、住所等信息 |

#### 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

* 主要作用：用户信息的发送和接收，信令信息的控制处理、安全保护等作用。
* 终端节点地址：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  yyyymmddTHHMMSSZ ，例如20211221T163014Z |
| host | 终端节点域名 |
| appkey | 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

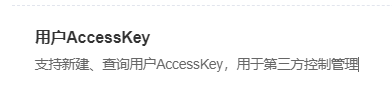
**信息的获取**

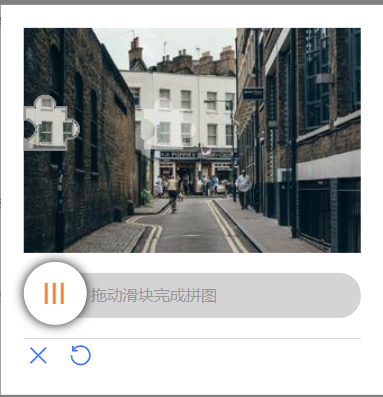
云网平台获取

登录云网门户 ，在“控制台”- >“个人中心”- >“安全设置” - >“用户 AccessKey”点击“查看”获取。

如之前没有获取，请点击“新建”。



IMG_257



**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。

**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 He ader 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(sha25 6(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

**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 yyyymmddTHHMMSSZ;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 ktim e；  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 k Ak；  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) |

**签名应用及示例**

由“构造动态秘钥”和“创建待签名字符串”分别的出来的待签名字符串 string\_sigture、kdate 生成出 Signature。

|  |  |
| --- | --- |
| **Signature** | **待签名字符串 string\_sigture、kdate；再根据 hmacsha256(k date,string\_sigture)得出的结果，再将结果进行 base64 编码得出 Sigture** |
| Eop-Authorizatio n | ctyun-eop-ak Header=你构造待签名字符串时的 he ader 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 h ost，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 Python3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%SZ", 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'

    #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**(r'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)

## 4.3API

### 4.3.1 通用型OCR

**1、接口描述**

针对图片中的文字，进行OCR检测，返回检测到的文字内容及坐标信息。

**2、请求方法**

POST

**3、接口要求**

* 图片格式限制：图片格式支持 png、jpg、jpeg、bmp格式
* 图片大小限制：单张图片大小不超过 10MB, 批量请求单次不超过 50张
* 图片尺寸限制：图片像素尺寸应大于 32x32 ，小于 5000x5000

**4、请求 URL**

/v1/aiop/api/2f3p1pnxpqm8/ocrdetect/ocr/v1/image.json

**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 | 应用appkey | 562b89493b1a40e1b97ea05e50dd8170 |  |
| ctyun-eop-request-id | 是 | string | 用户请求 id，由用户构造，用户可以通过 uuid 等方法自行生成唯一字符串，用于日志请求追踪。 | 3dfa732-b27b-464f-b15a-21ed6845afd5 |  |
| eop-date | 是 | string | 请求时间，由用户构造，形如 yyyymmddTHHMMSSZ | 20211221T163014Z |  |
| host | 是 | string | 终端节点域名，固定字段 | ai-global.ctapi.ctyun.cn |  |
| Eop-Authorization | 是 | string | 由天翼云官网 accessKey 和 securityKey 经签名后生成，参与签名生成的字段包括天翼云官网 accessKey 、securityKey、平台的 appkey（非必须），用户请求 id（非必须），请求时间，终端节点域名（非必须）以及请求体内容 |  |  |

请求体 body 参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| data | 是 | List | 一张或多张图像Base64编码数据构成的list。注意：图片需要使用常规base64编码方式 | \_9j\_4AAQSkZJRg... |  |

1. **请求代码示例**

Curl -X POST

"https://ai-global.ctapi.ctyun.cn/v1/aiop/api/2f3p1pnxpqm8/ocrdetect/ocr/v1/image.json"

-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 '{"data":["AAAAAAAAA...."]}'

**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** |
| statusCode | 是 | Int | 请求响应状态码 | 0 |  |
| message | 是 | Object | 请求信息结构体，message["success"]代表请求list中的成功数量，message["fail"]代表请求list中的失败数量。 | success |  |
| returnObj | 是 | Object | 识别的结果，按照列表形式排列，每个元素为图片对应的鉴定结果键值对，元素排序按照传入顺序排列，元素中包括识别到的文本行数目，位置以及文本行内容 |  | returnObj |

表message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| success | 否 | Int | 成功标识 | 1 |  |
| fail | 否 | Int | 错误标识 | 0 |  |

表 returnObj

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| text\_line | 否 | Int | 图片识别到的文字行数 | - |  |
| detail | 否 | Object | 按照列表形式排列，每个元素为图片中每一个识别到的文本行的具体信息，包含位置以及文本行内容 | - | detail |

表detail

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| name | 否 | String | 序号 | - |  |
| text | 否 | String | 识别到的文字 | - |  |
| box | 否 | List | 文字所在的坐标为8个浮点数构成的数组，8个浮点数依次分别表示左上、右上、右下、左下四个点的x坐标，y坐标。 | - |  |

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** |
| statusCode | 是 | Int | 错误码，参见本文档错误码说明部分 | 400003 |  |
| message | 是 | String/Object | 返回对应的可读消息，值为error | error | message |
| details | 是 | String | 返回对应的错误信息 | 请求体请求数据为空，没有包含内容 |  |
| returnObj | 否 | Object | 返回对应的错误编码和错误信息 |  | returnObj |

表Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| success | 否 | Int | 成功标识 | 1 |  |
| fail | 否 | Int | 错误标识 | 0 |  |

表returnObj

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| err\_code | 是 | Int | 错误码 | 400011 |  |
| err\_msg | 是 | String | 错误信息 | data 字段的 base64 字符串转换字节码异常 |  |

**8、返回值示例**

请求成功返回值示例

{

'statusCode': 0,

'message': {'success' : 1, 'fail' : 0},

'returnObj': [

{

'text\_line': 2,

'detail': [

{

'name': '0',

'text': '社会主义核心价值观',

'box': [

43.3963039677,

157.3534469991,

610.5843128949,

169.5419591656,

609.1529514034,

236.1498488994,

41.9649424762,

223.9613367328

]

},

{

'name': '1',

'text': '富强、民主、文明、和谐',

'box': [

84.6885821036,

244.0079416001,

573.8530589198,

248.174096093,

573.4117596933,

299.9887625014,

84.2472828771,

295.8226080086

]

}

]

}

]

}

299.9887625014,

84.2472828771,

295.8226080086

]

}

]

}

]

}

请求失败返回值示例

{

"statusCode": 400003,

"message": "error",

"details": "请求体请求数据为空，没有包含内容",

"error":"AI\_OP\_400003"

}

**9、状态码**

|  |  |
| --- | --- |
| **状态码** | **描述** |
| 200 | 表示请求成功 |

**10、错误码说明**

6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** |
| --- | --- | --- |
| **AI\_OP\_400001** | 请求路径错误 | 请求路径错误 |
| **AI\_OP\_400002** | 请求方法错误 | 请求方法错误，请使用 POST 请求 |
| **AI\_OP\_400003** | 请求体内容为空 | 请求体请求数据为空，没有包含内容 |
| **AI\_OP\_400004** | 请求体非 json 格式 | 请求体内容需要符合 json 要求 |
| **AI\_OP\_400005** | 请求体类型错误 | 请求体需为字典，不能为其他类型 |
| **AI\_OP\_400006** | 必传的参数未传 | 必须的参数（data）未传 |
| **AI\_OP\_400008** | 请求体的参数字段类型错误 | data 字段应该是 list 类型 |
| **AI\_OP\_400009** | 请求体的参数字段值为空 | data 列表为空 |
| **AI\_OP\_400011** | base64 数据处理异常 | data 字段的 base64 字符串转换字节码异常 |
| **AI\_OP\_400012** | 文件格式不合法 | 仅支持 jpeg/png/jpg/bmp 格式 |
| **AI\_OP\_400013** | 文件大小不符合要求 | 该文件大小不符合要求，图片要求小于 10M |
| **AI\_OP\_400018** | 超过个数限制 | data 超过个数限制，最多传 50 张图片 |
| **AI\_OP\_410001** | 图片解码错误 | 字节码解码为图片错误 |
| **AI\_OP\_410002** | 图片尺寸不符合要求 | 分辨率长宽尺寸应不高于 5000 不低于 32 |
| **AI\_OP\_500001** | OCR服务接口异常，请联系管理员 | 需要联系管理员处理 |

**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 身份证识别

**1、接口描述**

针对图片中的身份证，进行OCR检测，返回检测到的姓名、身份证号码等信息。

**2、请求方法**

POST

**3、接口要求**

图片格式限制：目前仅支持 jpg、jpeg、png、bmp 等常见格式。

图片大小限制：图片单张大小不超过 10MB，批量请求单次不超过 50张。

图片尺寸限制：图片像素尺寸应大于 32x32，小于 5000x5000。

**4、请求 URL**

/v1/aiop/api/2f3os7qq79xc/IdentityCard/ocr/v1/idcard.json

**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 | 应用appkey | 562b89493b1a40e1b97ea05e50dd8170 |  |
| ctyun-eop-request-id | 是 | string | 用户请求 id，由用户构造，用户可以通过 uuid 等方法自行生成唯一字符串，用于日志请求追踪。 | 3dfa732-b27b-464f-b15a-21ed6845afd5 |  |
| eop-date | 是 | string | 请求时间，由用户构造，形如 yyyymmddTHHMMSSZ | 20211221T163014Z |  |
| host | 是 | string | 终端节点域名，固定字段 | ai-global.ctapi.ctyun.cn |  |
| Eop-Authorization | 是 | string | 由天翼云官网 accessKey 和 securityKey 经签名后生成，参与签名生成的字段包括天翼云官网 accessKey 、securityKey、appkey（非必须），用户请求 id（非必须），请求时间，终端节点域名（非必须）以及请求体内容 |  |  |

请求体 body 参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| data | 是 | string | 一张或多张图像Base64编码字符串构成的list。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/2f3os7qq79xc/IdentityCard/ocr/v1/idcard.json"

-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 '{"data":["\_9j\_4AAQSkZJRg..."]}'

**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 | Object | 请求信息结构体，message["success"]代表请求list中的成功数量，message["fail"]代表请求list中的失败数量 |  | message |
| returnObj | 是 | List | 识别的结果，按照列表形式排列，每个元素为图片对应的鉴定结果键值对，元素排序按照传入顺序排列，元素中包括身份证正反面信息，以及姓名、性别、身份证号码等信息 |  | returnObj |

表 returnObj

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类** **型** | **说明** | **示例** | **下级对象** |
| note | 是 | String | 身份的正面或反面信息，国徽面为正面：front，人像面为反面：back | back |  |
| detail | 是 | String | 返回结果键值集合，包含姓名、性别、身份证号码等信 |  | detail |
| idcard\_face | 是 | String | 此API无法截取人像面身份证的人像子图，此字段固定为'no face' | no face |  |
| demo\_image | 是 | String | 无效字段（可忽略） | demo\_image |  |
| risk | 是 | String | 此API无法对身份证复印件进行判断，此字段固定为'normal' | normal |  |

表detail

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类** **型** | **说明** | **示例** | **下级对象** |
| name | 是 | String | 姓名 | 陈XX |  |
| sex | 是 | String | 性别 | 男 |  |
| nation | 是 | String | 民族 | 汉 |  |
| birthday | 是 | String | 出生日期 | 1995XX25 |  |
| idn | 是 | String | 身份证号 | 3101001995XX25XX16 |  |
| addr | 是 | String | 住址 | 上海市XX区XX路1号 |  |
| org | 是 | String | 签发机关 | 上海市公安局XX分局 |  |
| validperiod | 是 | String | 有效期限 | 2007.07.27-2027.07.27 |  |

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** |
| statusCode | 是 | Int | 错误码，放置API对应的错误码 | 4004 |  |
| message | 是 | String/Object | 请求失败时返回值为'error'或{'success':0,'fail':1} | error | message |
| details | 否 | String | 返回对应的错误描述 | data字段图片数据不是list格式 |  |
| returnObj | 否 | List | 返回对应的错误编码和错误信息 |  | returnObj |
| error | 是 | String | 返回对应的错误码 | AI\_OP\_4004 |  |

表message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| success | 否 | Int | 成功标识 | 1 |  |
| fail | 否 | Int | 错误标识 | 0 |  |

表returnObj

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| err\_code | 是 | Int | 错误码 | 4008 |  |
| err\_msg | 是 | String | 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

{

"statusCode": 0,

"message": {"success" : 1, "fail" : 0},

"returnObj": [{

"note": "front",

"detail": {

"name": "",

"sex": "",

"nation": "",

"birthday": "",

"idn": "",

"addr": "",

"org": "兰州市公安局城关分局",

"validperiod": "2007.07.27-2027.07.27"

},

"idcard\_face": "no face",

"demo\_image": "demo\_image",

"risk": "normal"

}]

}

请求失败返回值示例

示例1:

{

"statusCode": 4004,

"message": "error",

"details": "data字段图片数据不是list格式",

"error":"AI\_OP\_4004"

}

示例2:处理失败返回值示例

{

'statusCode': 0,

'message': {'success': 0, 'fail': 1},

'returnObj': [{'err\_code': 4008, 'err\_msg': 'base64数据处理异常'}]

}

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** |
| --- | --- | --- |
| **AI\_OP\_4001** | 请求JSON处理异常 | 请求JSON处理异常 |
| **AI\_OP\_4002** | body传入的不是字典 | body传入的不是字典 |
| **AI\_OP\_4003** | 请求中未包data字段 | 请求中未包data字段 |
| **AI\_OP\_4004** | data字段图片数据不是list格式 | data字段图片数据不是list格式 |
| **AI\_OP\_4005** | 图片list内图片数量为0 | 图片list内图片数量为0 |
| **AI\_OP\_4006** | 图片list内图片数量超过50张 | 图片list内图片数量超过50张 |
| **AI\_OP\_4008** | base64数据处理异常 | base64数据处理异常 |
| **AI\_OP\_4009** | 请求文件格式不合法，仅支持 jpeg/png/jpg/bmp 格式 | 请求文件格式不合法，仅支持 jpeg/png/jpg/bmp 格式 |
| **AI\_OP\_4010** | 单张图片大小超过10M | 单张图片大小超过10M |
| **AI\_OP\_4011** | 图片尺寸不符合要求，分辨率长宽尺寸应小于5000大于32 | 图片尺寸不符合要求，分辨率长宽尺寸应小于5000大于32 |
| **AI\_OP\_4012** | 请求方法错误，请使用POST请求 | 请求方法错误，请使用POST请求 |
| **AI\_OP\_4013** | 无效的请求路径，请确保请求路径正确 | 无效的请求路径，请确保请求路径正确 |
| **AI\_OP\_5000** | OCR服务接口异常，请联系管理员 | OCR服务接口异常，请联系管理员 |

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

### 4.3.4 车牌识别

**1、接口描述**

对图片中的车牌（仅限中国大陆境内的蓝牌、黄牌（单层）、新能源绿牌），进行OCR检测，返回检测到的车牌内容及车牌位置坐标。

**2、请求方法**

POST

**3、接口要求**

图片格式限制：目前仅支持 jpg、jpeg、png、bmp 等常见格式

图片大小限制：图片单张大小不超过 10MB，批量请求单次不超过 50张

图片尺寸限制：图片像素尺寸应大于 32x32，小于 5000x5000

**4、请求 URL**

/v1/aiop/api/2gt54ed8660w/driven-plate-ocr/platelicense.json

**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 | 应用appkey | 562b89493b1a40e1b97ea05e50dd8170 |  |
| ctyun-eop-request-id | 是 | string | 用户请求 id，由用户构造，用户可以通过 uuid 等方法自行生成唯一字符串，用于日志请求追踪 | 3dfa732-b27b-464f-b15a-21ed6845afd5 |  |
| eop-date | 是 | string | 请求时间，由用户构造，形如 yyyymmddTHHMMSSZ | 20211221T163014Z |  |
| host | 是 | string | 终端节点域名，固定字段 | ai-global.ctapi.ctyun.cn |  |
| Eop-Authorization | 是 | string | 由天翼云官网 accessKey 和 securityKey 经签名后生成，参与签名生成的字段包括天翼云官网 accessKey 、securityKey、平台的 appkey（非必须），用户请求 id（非必须），请求时间，终端节点域名（非必须）以及请求体内容 |  |  |

请求体 body 参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| data | 是 | List | 一张或多张图像Base64编码字符串构成的list。注意：图片需要使用常规safe base64编码方式 | \_9j\_4AAQSkZJRg... |  |

**6、请求代码示例**

Curl -X POST

"https://ai-global.ctapi.ctyun.cn/v1/aiop/api/2gt54ed8660w/driven-plate-ocr/platelicense.json"

-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 '{"data":["\_9j\_4AAQSkZJRg..."]}'

**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** |
| statusCode | 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 | Object | 请求信息结构体，message["success"]代表请求list中的成功数量，message["fail"]代表请求list中的失败数量。 | success |  |
| returnObj | 是 | Object | 识别的结果，按照列表形式排列，每个元素为图片对应的鉴定结果键值对，元素排序按照传入顺序排列，元素中包括识别到的车牌数目，位置以及车牌内容及其识别的置信度 |  | returnObj |

表message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| success | 否 | Int | 成功标识 | 1 |  |
| fail | 否 | Int | 错误标识 | 0 |  |

表returnObj

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| number of plate licenses | 是 | Int | 图片中识别到的车牌数量 |  |  |
| detail | 是 | Int | 图片内识别到的车牌list的具体信息，list内每一项包括车牌位置、牌照类型、车牌内容、置信度等 |  | detail |

表detail

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| box | 是 | List | 识别到的车牌8坐标位置信息，包含四个数组元素，依次分别对应车牌左上、右上、右下、左下四个点，每个数组元素包含两个整数，代表坐标点的x坐标，y坐标 |  |  |
| plate\_type | 是 | String | 识别到的车牌种类信息，现阶段只支持中国大陆境内的蓝牌、黄牌（单层）、新能源绿牌 | 蓝牌 |  |
| platelicense | 是 | String | 识别到的车牌内容 | 冀D137G5 |  |
| confidence | 是 | Float | 识别到的车牌置信度 | 1.0 |  |

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** |
| statusCode | 是 | Int | 错误码，放置API对应的错误码 | 4003 |  |
| message | 是 | String/Object | 请求失败时返回值为'error'或{'success':0,'fail':1} | error | message |
| details | 是 | String | 返回对应的错误描述 | 请求中未包data字段 |  |
| returnObj | 否 | Object | 返回对应的错误编码和错误信息 |  | returnObj |
| error | 是 | String | 返回对应的错误码 | AI\_OP\_4003 |  |

表returnObj

| **参数** | **是否必填** | | **参数类型** | **说明** | **示例** | **下级对象** |
| --- | --- | --- | --- | --- | --- | --- |
| errorCode | 是 | String | | 错误码 | 4008 |  |
| errorMessage | 是 | String | | 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

{

"statusCode": 0,

"message": {"success" : 1, "fail" : 0},

"returnObj": [{

"number of plate licenses": 1,

"detail": [{

"box": [

[123, 191],[214, 195],[213, 226],[122, 222]

],

"plate\_type": "蓝牌",

"platelicense": "浙C80000",

"confidence": 0.99

}]

}]

}

请求失败返回值示例

示例1:

{

"statusCode": 4004,

"message": "error",

"details": "data字段图片数据不是list格式",

"error":"AI\_OP\_4004"

}

示例2:处理失败返回值示例

{

'statusCode': 0,

'message': {'success': 0, 'fail': 1},

'returnObj': [{'err\_code': 4008, 'err\_msg': 'base64数据处理异常'}]

}

**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** |
| --- | --- | --- |
| **AI\_OP\_4001** | 请求JSON处理异常 | 请求JSON处理异常 |
| **AI\_OP\_4002** | body传入不是字典 | body传入不是字典 |
| **AI\_OP\_4003** | 请求中未包data字段 | 请求中未包data字段 |
| **AI\_OP\_4004** | data字段图片数据不是list格式 | data字段图片数据不是list格式 |
| **AI\_OP\_4005** | 图片list内图片数量为0 | 图片list内图片数量为0 |
| **AI\_OP\_4006** | 图片list内图片数量超过50张 | 图片list内图片数量超过50张 |
| **AI\_OP\_4008** | base64数据处理异常 | base64数据处理异常 |
| **AI\_OP\_4009** | 请求文件格式不合法，仅支持 jpeg/png/jpg/bmp 格式 | 请求文件格式不合法，仅支持 jpeg/png/jpg/bmp 格式 |
| **AI\_OP\_4010** | 单张图片大小超过10M | 单张图片大小超过10M |
| **AI\_OP\_4011** | 图片尺寸不符合要求,分辨率长宽尺寸应不高于5000不低于32 | 图片尺寸不符合要求,分辨率长宽尺寸应不高于5000不低于32 |
| **AI\_OP\_5000** | OCR服务接口异常，请联系管理员 | OCR服务接口异常，请联系管理员 |

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

### 4.3.4 营业执照识别

**1、接口描述**

针对营业执照图片，进行OCR检测，返回检测到的统一社会信用代码，注册号，名称，类型，住所，法定代表人，注册资金，成立日期，营业期限，实收资本。

**2、请求方法**

POST

**3、接口要求**

图片格式限制：图片格式支持 png、jpg、jpeg、bmp格式

图片大小限制：单张图片大小不超过 7MB

图片尺寸限制：图片像素尺寸应不小于 32x32，不大于 5000x5000

**4、请求 URL**

/v1/aiop/api/2z0yhhrzgv0g/tts/predict

**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 | 应用appkey | 562b89493b1a40e1b97ea05e50dd8170 |  |
| ctyun-eop-request-id | 是 | string | 用户请求 id，由用户构造，用户可以通过 uuid 等方法自行生成唯一字符串，用于日志请求追踪。 | 3dfa732-b27b-464f-b15a-21ed6845afd5 |  |
| eop-date | 是 | string | 请求时间，由用户构造，形如 yyyymmddTHHMMSSZ | 20211221T163014Z |  |
| host | 是 | string | 终端节点域名，固定字段 | ai-global.ctapi.ctyun.cn |  |
| Eop-Authorization | 是 | string | 由天翼云官网 accessKey 和 securityKey 经签名后生成，参与签名生成的字段包括天翼云官网 accessKey 、securityKey、平台的 appkey（非必须），用户请求 id（非必须），请求时间，终端节点域名（非必须）以及请求体内容 |  |  |

请求体 body 参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| Action | 是 | String | 请求参数放置于http body中，action代表需要请求的具体算法，只能为"BusinessCardOCR" | BusinessCardOCR |  |
| ImageData | 是 | String | 一张图像Base64编码数据string。图片需要使用常规safe base64编码方式，不包含前缀"data:img/jpg;base64," |  |  |

1. **请求代码示例**

Curl -X POST "https://ai-global.ctapi.ctyun.cn/v1/aiop/api/2k8jo3ghhjwg/businesslicense/ocr/v1/BusinessLicense.json"

-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 '{"Action": "BusinessCardOCR","ImageData": "9j\_4AAQSkZJRgABAQAAAQABAAD\_4QAwRXhpZgAATU0AKgAAA..."}'

**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** |
| statusCode | 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statusCode为0，返回success；如果statusCode非0，则返回对应的可读错误信息 | success |  |
| returnObj | 是 | Object | 识别的结果字典，字典内包营业执照的整体判断信息以及统一社会信用代码、名称、住所等详细键值对信息 |  | returnObj |

表returnObj

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| license\_type | 是 | String | 营业执照的整体判断信息，'Business License'表示图片是营业执照，'Not Business License'表示图片不是营业执照 | Business License |  |
| detail | 是 | Object | 返回结果键值集合，包含统一社会信用代码、名称、住所等信息 | - | detail |

表detail

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **是否必填** | **参数类型** | **说明** | **示例** | **下级对象** |
| union\_id | 是 | String | 统一社会信用代码 | - |  |
| regist\_no | 是 | String | 注册号 | - |  |
| name | 是 | String | 名称 | - |  |
| type | 是 | String | 类型 | - |  |
| addr | 是 | String | 住所 | - |  |
| legal\_rep | 是 | String | 法定代表人 | - |  |
| regist\_captial | 是 | String | 注册资金 | - |  |
| establish\_date | 是 | String | 成立日期 | - |  |
| operation\_period | 是 | String | 营业期限 | - |  |
| real\_captial | 否 | String | 实收资本（检测到双列营业执照返回此项，单列营业执照无此项信息） | - |  |

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** |
| statusCode | 是 | Int | 错误码，参见本文档错误码说明部分 | 400009 |  |
| message | 是 | String | 返回对应的错误信息 | Action值设置错误 |  |
| details | 是 | String | 返回对应的错误描述 | Action值设置错误 |  |
| error | 是 | String | 返回对应的错误码 | AI\_OP\_400009 |  |

**8、返回值示例**

请求成功返回值示例

{

"statusCode": 0,

"message": "success",

"returnObj": {

"license\_type": "Business License",

"detail": {

"union\_id": "",

"regist\_no": "32050XXXXXX8749",

"name": "苏州市XXXX有限公司",

"type": "有限公司（自然人控股）",

"addr": "苏州市相城区XXXXXXXXX",

"legal\_rep": "潘X",

"regist\_captial": "200万元人民币",

"establish\_date": "2005年6月9日",

"operation\_period": "2005年06月09日至2015年06月08日",

"real\_captial": "200万元人民币"

}

}

}

请求失败返回值示例

{

"statusCode": 400009,

"message": "Action值设置错误",

"details": "Action值设置错误",

"error":"AI\_OP\_400009"

}

**9、状态码**

|  |  |
| --- | --- |
| **状态码** | **描述** |
| 200 | 表示请求成功 |

**10、错误码说明**

6 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** |
| --- | --- | --- |
| **AI\_OP\_400003** | body内容为空 | Body请求数据为空，没有包含内容 |
| **AI\_OP\_400004** | body请求体非json格式 | body内容需要符合json要求 |
| **AI\_OP\_400005** | 请求体类型错误 | 请求体需为字典，不能为其他类型 |
| **AI\_OP\_400006** | 必传的参数未传 | 必须的参数（'Action'、'ImageData'）未传 |
| **AI\_OP\_400007** | 传递非法参数 | body字典内有除（'Action'、'ImageData'）外的参数 |
| **AI\_OP\_400008** | 请求体的字段类型错误 | 请求体的字段（'Action'、'ImageData'）类型错误，类型只能为字符串，不能为其他类型 |
| **AI\_OP\_400009** | Action值设置错误 | Action值设置错误 |
| **AI\_OP\_400010** | 'ImageData'字段值为空字符 | 'ImageData'字段的base64字符串转换字节码异常 |
| **AI\_OP\_400011** | 'ImageData'字段base64数据处理异常 | 'ImageData'字段base64数据处理异常 |
| **AI\_OP\_400012** | 请求文件格式不合法 | 仅支持 jpeg/png/jpg/bmp 格式 |
| **AI\_OP\_400013** | 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7M |
| **AI\_OP\_400014** | 图片解码错误 | 字节码解码为图片错误 |
| **AI\_OP\_400015** | 图片尺寸不符合要求 | 分辨率长宽尺寸应不高于5000不低于32 |
| **AI\_OP\_500001** | 服务接口异常，请联系管理员 | 需要联系管理员处理 |

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

## 4.4 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 |
| 2023-12-21 | 第三次正式发布。本次更新说明如下：修改API文档内容 |

**5** 常见问题

### 5.1 计费类

**印刷文字识别支持哪些计费方式？**

我们提供封顶资源包的计费方式，资源包有效期一年，根据调用次数不同，费用也有所不同。

* 印刷文字识别API服务免费额度表

| **API名称** | **免费额度** | **具体说明** |
| --- | --- | --- |
| 通用OCR识别 | 10000次 | 以单个API为统计维度，有效期从领取之日起一年内有效，过期作废 |
| 身份证识别 | 10000次 | 以单个API为统计维度，有效期从领取之日起一年内有效，过期作废 |
| 车牌识别 | 10000次 | 以单个API为统计维度，有效期从领取之日起一年内有效，过期作废 |

* 印刷文字识别API资源包价格表

| **API名称** | **10万次** | **100万次** | **500万次** | **1000万次** |
| --- | --- | --- | --- | --- |
| 通用OCR识别 | 450元 | 3200元 | 11500元 | 22000元 |
| 身份证识别 | 1350元 | 9600元 | 39000元 | 78000元 |
| 车牌识别 | 1350元 | 9600元 | 39000元 | 78000元 |

**印刷文字识别是否支持续订？**

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。

**OCR服务开通后能否关闭？不使用是否会产生费用？**

OCR服务开通后无法关闭，如您后续无使用需求，不调用已开通的OCR接口即无相关费用产生。

同时，请您保管好AppKey和AppSecret信息，保证后期无人调用。

### 5.2 购买类

**印刷文字识别服务是否支持代金券付款？**

支持代金券付款，同时还支持优惠券、提货券付款。具体使用方法：在我的>费用中心>卡券管理，选择相应的代金券进行使用。

**印刷文字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试用。

* 印刷文字识别API服务免费额度表

| **API名称** | **免费额度** | **具体说明** |
| --- | --- | --- |
| 通用OCR识别 | 10000次 | 以单个API为统计维度，有效期从领取之日起一年内有效，过期作废 |
| 身份证识别 | 10000次 | 以单个API为统计维度，有效期从领取之日起一年内有效，过期作废 |
| 车牌识别 | 10000次 | 以单个API为统计维度，有效期从领取之日起一年内有效，过期作废 |

**资源包买错了可以退款吗？**

若因产品介绍说明有误导致的购买错误且资源包尚未使用 ，您可通过通过[天翼云官网工单](https://www.ctyun.cn/console/smartservice/ticket/workorder/submit)或者客服电话【400-810-9889】沟通申请手动退款，可能需要数个工作日审批，请耐心等待，款项会原路退回。在审批期间，您不能使用该资源包，一旦使用资源包就会导致退款失败。

**资源包使用到一半可以退吗？**

资源包一经售卖且产生正常调用抵扣，不允许进行退款。资源包属于预付费产品，建议您在购买资源包的时候按实际调用情况预估调用量以免造成资源浪费，谢谢配合。

### 5.3 操作类

**印刷文字识别在什么时候进入可使用状态？**

限免类产品：将该产品加入到应用后，即可进入使用状态。

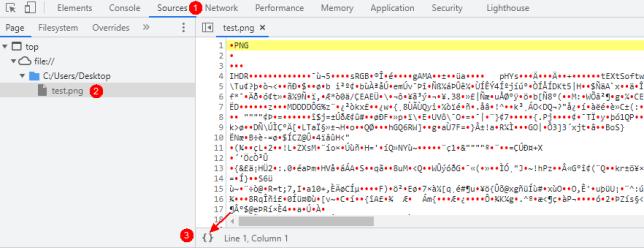
公测及商用类产品：将该产品加入到应用，并领取免费试用额度或购买相应资源包，方可进入使用状态。

**如何获取图片 base64 编码？**

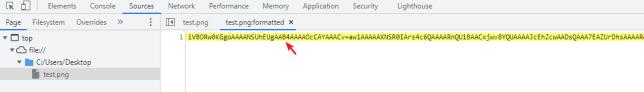
1. 将需转码的图片拖拽至 Chrome 浏览器中，浏览器上显示该图片。

2. 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。



3. 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。



**印刷文字识别 API 是否可以混用？**

不建议混用，例如有专门的身份证识别，就不建议使用“通用 OCR”识别身份证图片，不同接口底层算 法不同，混用会导致识别效果差。

**印刷文字识别服务支持批量识别吗？**

车牌识别、通用型OCR、身份证识别服务支持批量识别，每次最多支持50张图片的识别，可大大提高处理效率，节省时间和精力。

**印刷文字识别服务识别结果可以转化为 Word、TXT、pdf 吗？**

OCR 提取之后返回的结果是 JSON 格式，需要用户通过编程，将结果保存为 Word、TXT、pdf 格式。

**印刷文字识别的并发是多少？**

默认支持5个QPS，建议您在程序中进行一定的请求限制，避免收到大量限流报错。

如果因业务需要QPS超过5个，请提前线下咨询沟通再购买下单。

注意：如果您的程序在失败时有重试机制，当您扩大并发量后接口返回错误码时，请不要重试，否则可能加重限流报错情况。

### 5.4 使用限制

**印刷文字识别前端展示页面对于上传的图片是否有要求？**

（1）图片单张大小不超过 2MB；

（2）图片格式支持 jpg、png、bmp 等。

**印刷文字识别 API 接口对于上传的图片是否有要求？**

图片格式限制：目前仅支持 jpg、jpeg、png、bmp 等常见格式。

图片大小限制：图片单张大小不超过 10MB，批量请求单次不超过 50张。

图片尺寸限制：图片像素尺寸应大于 32x32，小于 5000x5000。

**印刷文字识别是 HTTP GET 请求还是 HTTP POST 请求？**

印刷文字识别以API的方式提供服务，支持HTTP POST请求。

HTTP POST提供了加密传输、身份验证和授权以及请求参数验证和过滤等机制，从而可以更好的保障用户的数据安全。

**印刷文字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。

**印刷文字识别的 api 接口是否支持单次请求多张图片？**

印刷文字识别的API接口对单次请求的图片限制不一致，具体可参考【API参考】-【API】-具体产品（如通用型OCR）“接口要求”中的详细说明。

**注意**

如用户操作手册说明与官网帮助中心说明不一致，以官网帮助中心说明为准。