文字识别服务

API 参考

文档版本 01

发布日期 2025-08-29



华为云计算技术有限公司

版权所有 © 华为云计算技术有限公司 2025。 保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明


和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为云计算技术有限公司

地址： 贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址： https://www.huaweicloud.com/

目 录

1 使用前必读 1
1.1 概述 1
1.2 调用说明 1
1.3 终端节点 1
1.4 约束与限制 3
1.5 基本概念 3
2 API 概览 5
3 如何调用 API 8
3.1 总览 8
3.2 开通服务 9
3.3 构造请求 12
3.4 认证鉴权 15
3.5 返回结果 18
4 API 20
4.1 智能文档解析 20
4.2 通用表格识别 38
4.3 通用文字识别 51
4.4 网络图片识别 64
4.5 智能分类识别 77
4.6 手写文字识别 96
4.7 印章识别 108
4.8 身份证识别 119
4.9 户口本识别 141
4.10 行驶证识别 153
4.11 驾驶证识别 169
4.12 护照识别 182
4.13 银行卡识别 194
4.14 营业执照识别 204
4.15 道路运输证识别 215
4.16 不动产证识别 226
4.17 车辆合格证识别 236
4.18 道路运输从业资格证识别 249

 版权所有 © 华为云计算技术有限公司

4.19 车牌识别 262
4.20 名片识别 272
4.21 VIN 码识别 283
4.22 增值税发票识别 292
4.23 发票验真 312
4.24 机动车销售发票识别 331
4.25 出租车发票识别 347
4.26 火车票识别 358
4.27 定额发票识别 372
4.28 车辆通行费发票识别 382
4.29 飞机行程单识别 392
4.30 电子面单识别 405
4.31 保险单识别 416
4.32 财务报表识别 429
4.33 承兑汇票识别 440
4.34 银行回单识别 453
4.35 泰文身份证识别 464
4.36 哥伦比亚身份证识别 478
4.37 泰国车牌识别 487
4.38 自定义模板 OCR 497
5 状态码 508
6 错误码 511
7 附录 518
7.1 获取项目 ID 518
7.2 获取账号 ID 519
7.3 配置 OBS 访问权限 519

 版权所有 © 华为云计算技术有限公司

1 使用前必读

1.1 概述

文字识别（Optical Character Recognition，简称OCR）是指将图片、扫描件或PDF、OFD文档中的打印字符进行检测识别成可编辑的文本格式，以JSON格式返回识别结果。

OCR以开放API（Application Programming Interface，应用程序编程接口）的方式提供给用户，用户通过实时访问和调用API获取推理结果，帮助用户自动采集关键数据，打造智能化业务系统，提升业务效率。调用API时，需要用户网络可以访问公网。

您可以使用本文档提供文字识别服务API的描述、语法、参数说明及样例等内容，进行相关操作，例如文字识别包含的通用类、证件类和票据类等具体接口使用说明。不同区域所部署的接口不同，详情请参见**终端节点**；接口所支持的全部操作请参见**2 API概览**。

在调用文字识别API之前，请确保已经充分了解文字识别相关概念。

OCR还提供多种编程语言的SDK供您使用，SDK使用方法请参考《**SDK参考**》。

1.2 调用说明

文字识别提供了REST（Representational State Transfer）风格的API，支持您通过HTTPS请求调用，调用方法请参见**如何调用API**。

调用API时，需要用户网络可以访问公网。

同时文字识别还提供多种编程语言的SDK供您使用，SDK的使用方法请参见《**SDK参考**》。

1.3 终端节点

终端节点即调用API的请求地址，不同服务不同区域的终端节点不同，您可以从**地区和终端节点**中查询所有服务的终端节点。

文字识别的终端节点如**表1-1**所示，请您根据业务需要选择对应区域的终端节点。

版权所有 © 华为云计算技术有限公司

业务的服务器部署区域与OCR服务区域不一致，只要服务器可以访问公网，即可使用OCR服务。一般情况下，建议就近选择靠近您或者您的目标用户的区域，这样可以减少网络时延，提高访问速度。

表 1-1 文字识别的终端节点

区域名称	区域	终端节点（Endpoint）	部署服务
华北-北京四	cn-north-4	ocr.cn-north-4.myhuaweicloud.com ocr.cn-north-4.myhuaweicloud.cn	身份证识别、驾驶证识别、行驶证识别、增值税发票识别、机动车销售发票识别、通用表格识别、通用文字识别、手写文字识别、车辆通行费发票识别、飞机行程单识别、定额发票识别、出租车发票识别、火车票识别、营业执照识别、银行卡识别、护照识别、道路运输证识别、智能分类识别、车牌识别、名片识别、VIN码识别、网络图片识别、发票验真、保险单识别、智能文档解析、不动产证识别、承兑汇票识别、车辆合格证识别、印章识别、户口本识别、银行回单识别
华南-广州	cn-south-1	ocr.cn-south-1.myhuaweicloud.com ocr.cn-south-1.myhuaweicloud.cn	网络图片识别、不动产证识别、承兑汇票识别
华东-上海一	cn-east-3	ocr.cn-east-3.myhuaweicloud.com ocr.cn-east-3.myhuaweicloud.cn	网络图片识别、身份证识别、驾驶证识别、行驶证识别、银行卡识别、通用表格识别、发票验真、财务报表识别、保险单识别、机动车销售发票识别、道路运输证识别、增值税发票识别、通用文字识别、营业执照识别、车牌识别、道路运输从业资格证识别
亚太-曼谷	ap-southeast-2	ocr.ap-southeast-2.myhuaweicloud.com ocr.ap-southeast-2.myhuaweicloud.cn	泰文身份证识别、泰国车牌识别
拉美-圣地亚哥	la-south-2	ocr.la-south-2.myhuaweicloud.com ocr.la-south-2.myhuaweicloud.cn	哥伦比亚身份证识别

版权所有 © 华为云计算技术有限公司

区域名称	区域	终端节点（Endpoint）	部署服务
拉美-墨西哥城二	la-north-2	ocr.la-north-2.myhuaweicloud.com	哥伦比亚身份证识别

1.4 约束与限制

更详细的限制请参见具体API的说明和**产品介绍约束与限制**。

1.5 基本概念

- 账号

  用户注册华为云时的账号，账号对其所拥有的资源及云服务具有完全的访问权限，可以重置用户密码、分配用户权限等。由于账号是付费主体，为了确保账号安全，建议您不要直接使用账号进行日常管理工作，而是创建用户并使用他们进行日常管理工作。

- 用户

  由账号在IAM中创建的用户，是云服务的使用人员，具有身份凭证（密码和访问密钥）。

  在**我的凭证**下，您可以查看账号ID和用户ID。通常在调用API的鉴权过程中，您需要用到账号、用户和密码等信息。

- 区域（Region）

  从地理位置和网络时延维度划分，同一个Region内共享弹性计算、块存储、对象存储、VPC网络、弹性公网IP、镜像等公共服务。Region分为通用Region和专属Region，通用Region指面向公共租户提供通用云服务的Region；专属Region指只承载同一类业务或只面向特定租户提供业务服务的专用Region。

  详情请参见**区域和可用区**。

- 可用区（AZ，Availability Zone）

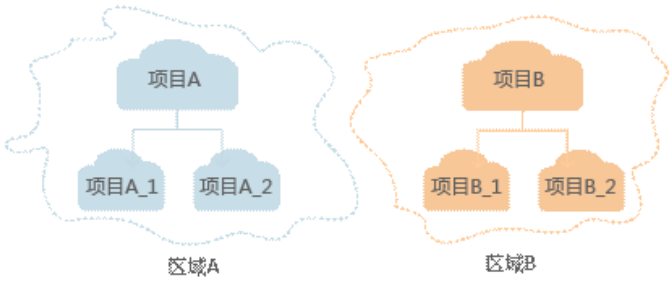
  一个AZ是一个或多个物理数据中心的集合，有独立的风火水电，AZ内逻辑上再将计算、网络、存储等资源划分成多个集群。一个Region中的多个AZ间通过高速光纤相连，以满足用户跨AZ构建高可用性系统的需求。

- 项目

  华为云的区域默认对应一个项目，这个项目由系统预置，用来隔离物理区域间的资源（计算资源、存储资源和网络资源），以默认项目为单位进行授权，用户可以访问您账号中该区域的所有资源。如果您希望进行更加精细的权限控制，可以在区域默认的项目中创建子项目，并在子项目中购买资源，然后以子项目为单位进行授权，使得用户仅能访问特定子项目中资源，使得资源的权限控制更加精确。

版权所有 © 华为云计算技术有限公司

图 1-1 项目隔离模型



版权所有 © 华为云计算技术有限公司

2 API 概览

通过使用文字识别服务的自研API，您可以使用文字识别服务的所有功能，如表2-1所示。

文字识别服务当前支持通用类、证件类、票据类和智能分类四种不同类型的接口。您可以通过在线体验，体验接口的识别效果。

服务部署区域是从地理位置和网络时延维度划分，同一个区域内共享弹性计算、块存储、对象存储、VPC网络、弹性公网IP、镜像等公共服务。一般情况下，建议就近选择靠近您或者您的目标用户的区域，这样可以减少网络时延，提高访问速度。对于OCR服务，不同的区域之间资源包不互通，请根据您的实际需求慎重选择。

表 2-1 接口说明

类别	API	说明
智能文档解析	智能文档解析	对证件、票据、表单等任意版式文档进行键值对提取、文字识别、以及表格识别等任务，实现进阶高效的自动化结构化返回。
通用类	通用表格识别	识别表格图片上的文字内容，并返回识别的结构化结果。
	通用文字识别	识别图片上的文字内容，并返回识别的文字和坐标。
	网络图片识别	识别网络图片中的文字内容，并以JSON格式返回识别的结构化结果。支持横向、竖向、艺术字识别。
	智能分类识别	检测定位图片上指定要识别的票证（票据、证件或其他文字载体），并以JSON格式返回识别的结构化结果。
	手写文字识别	识别手写文字图片中的文字内容。
证件类	身份证识别	识别身份证图片中正面与反面的文字内容，并返回识别的文字和坐标。
	户口本识别	识别户口本中的文字信息，并返回识别的结构化结果。

版权所有 © 华为云计算技术有限公司

类别	API	说明
	行驶证识别	识别行驶证图片中主页与副页的文字内容，并返回识别的文字和坐标。
	驾驶证识别	识别驾驶证图片中主页与副页的文字内容，并返回识别的文字和坐标。
	护照识别	识别护照首页图片中的文字信息，并以JSON格式返回识别的结构化结果。
	银行卡识别	识别银行卡上的关键文字信息，并以JSON格式返回识别的结构化结果。
	营业执照识别	识别营业执照首页图片中的文字信息，并以JSON格式返回识别的结构化结果。
	道路运输证识别	识别道路运输证首页中的文字信息，并以JSON格式返回识别的结构化结果。
	车牌识别	识别车牌图片中的车牌信息，并返回其坐标和内容。
	名片识别	识别名片图片上的文字信息，并返回识别的结构化结果。支持对多种不同版式名片进行结构化信息提取。
	VIN码识别	识别图片中的车架号信息，并将识别结果返回给用户。
	电子面单识别	识别用户上传的电子面单图片中的文字内容，并将识别的结果返回给用户。
	道路运输从业资格证识别	识别道路运输从业资格证上的关键文字信息，并返回识别的结构化结果。
	泰文身份证识别	识别泰国身份证中的文字信息，并以JSON格式返回识别的结构化结果。
	不动产证识别	识别不动产证中的文字信息，并返回识别的结构化结果。
	车辆合格证识别	识别车辆合格证中的文字信息，并返回识别的结构化结果。
	哥伦比亚身份证识别	识别哥伦比亚身份证中的文字信息，并将识别的结构化结果返回给用户。
票据类	增值税发票识别	识别增值税发票图片中的文字内容，并返回识别的结构化结果。
	发票验真	支持10种增值税发票的信息核验。
	机动车销售发票识别	识别机动车销售发票图片中的文字内容，并返回识别的结构化结果。
	出租车发票识别	识别出租车发票中的文字信息，并以JSON格式返回识别的结构化结果。

版权所有 © 华为云计算技术有限公司

类别	API	说明
	火车票识别	识别火车票中的文字信息，并以JSON格式返回识别的结构化结果
	定额发票识别	识别定额发票中的文字信息，并以JSON格式返回识别的结构化结果。
	车辆通行费发票识别	识别车辆通行费发票中的关键文字信息，并以JSON格式返回识别的结构化结果。
	飞机行程单识别	识别飞机行程单中的文字信息，并以JSON格式返回识别的结构化结果。
	财务报表识别	识别用户上传的表格图片中的文字内容，并将识别的结果返回给用户。
	承兑汇票识别	识别承兑汇票识别中的关键字段, 并以json格式返回结构化结果。
	银行回单识别	支持对银行回单版式进行文字识别及键值对提取，实现高效的自动化结构化返回。
行业类	保险单识别	识别保险单图片上的文字信息，并将识别的结构化结果返回给用户。支持对多板式保险单的扫描图片及手机照片进行结构化信息提取。
自定义模板	自定义模板OCR	自定义模板OCR，支持用户自定义模板，对于版式固定的各种票据和卡证，通过可视化界面操作，指定需要识别的关键字段，实现用户特定格式图片的自动识别和结构化提取。

版权所有 © 华为云计算技术有限公司

3 如何调用 API

3.1 总览

调用API可以通过Postman或SDK进行调试。包含开通服务、配置环境、认证鉴权和构造请求（调用服务）。

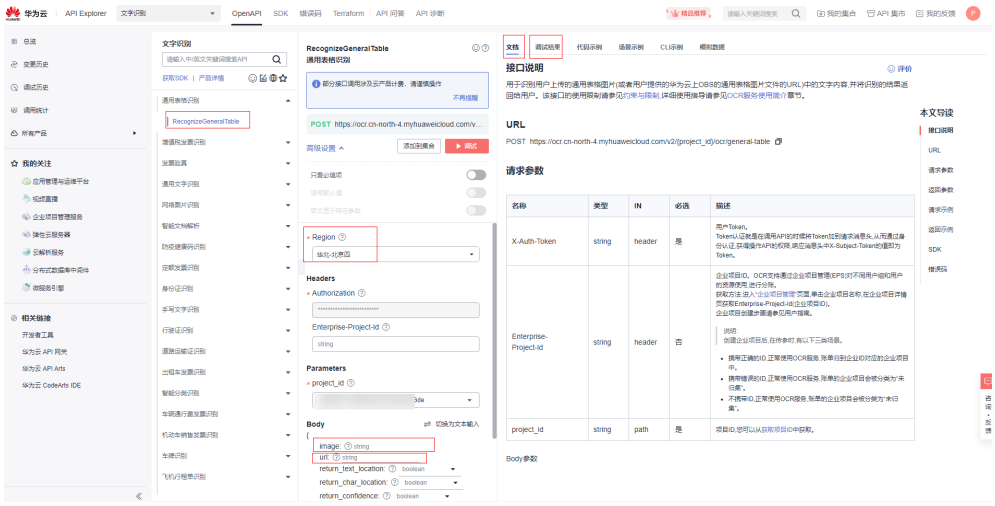
1. 开通服务：包含注册登录、创建IAM账号（可选）、开通服务，详情请参考**开通服务**章节。
2. 配置环境：可选择**下载Postman**或**下载SDK**。
3. 认证鉴权：可选择**Token认证**（Postman环境）或**AK/SK认证**(SDK环境)。根据步骤二配置的环境选择认证鉴权的方式。
4. 构造请求：Postman环境的构造请求请参考**构造请求**章节，SDK环境的API调用详见**SDK参考**。

除上述两种方式外，还可通过**APIE在线调试**的方式调用API，步骤如下。

1. **开通服务**后进入APIE界面。
2. 选择已开通的服务和开通服务对应的Region。
3. Body部分填入image的base64编码或url均可，其它参数根据实际情况选填，参数部分可参考APIE界面的“接口文档”。
4. 单击调试，在“调试结果”中查看调用结果。

版权所有 © 华为云计算技术有限公司

图 3-1 APIE 在线调试



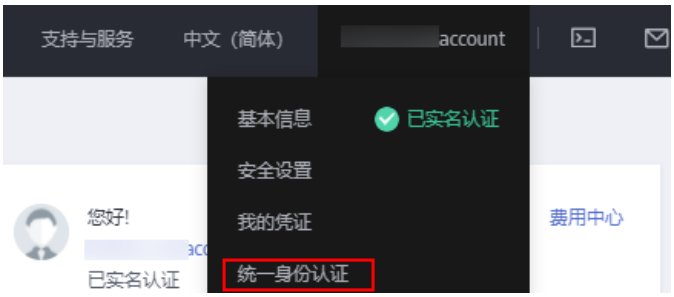
3.2 开通服务

步骤1 在**华为云官网**注册华为云账号，并登录。账号不能处于欠费或冻结状态。

步骤2 创建IAM账号（可选）：

1. 登录华为云**控制台**，在右上角用户名处选择“统一身份认证”。

图 3-2 统一身份认证



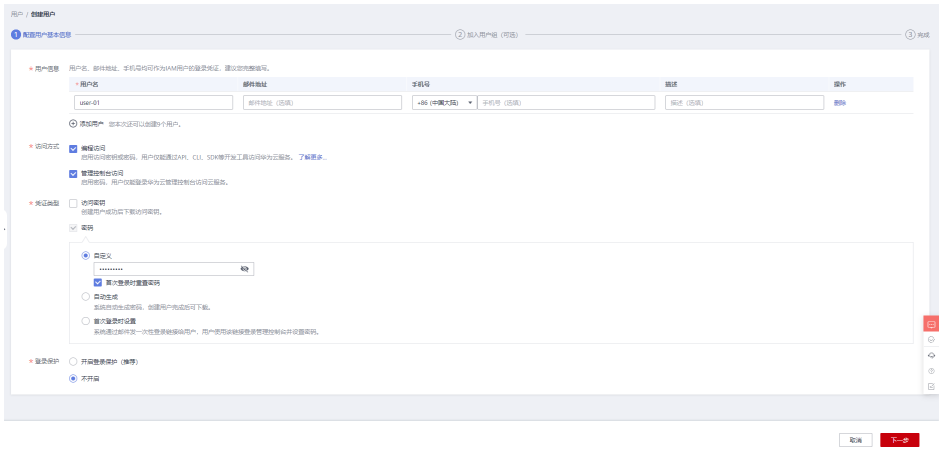
2. 单击“创建用户”，并设置IAM账户的用户名、密码，创建IAM用户。

图 3-3 创建用户



版权所有 © 华为云计算技术有限公司

图 3-4 设置用户名、密码



3. 给子账户授予权限请参考**权限管理**。

图 3-5 设置用户组



步骤3 开通服务：

1. 单击“控制台”>“服务列表”，找到“文字识别OCR”，进入**文字识别控制台**。

图 3-6 文字识别控制台



2. 在文字识别控制台“总览”页面，选择对应产品的“终端节点”。不同区域OCR产品部署情况请参考**终端节点**。

图 3-7 终端节点



3. 选择产品类别，如：通用类、证件类、票据类等，然后选择需要使用的服务，执行开通操作。计费方式为按需付费。

版权所有 © 华为云计算技术有限公司

图 3-8 服务开通

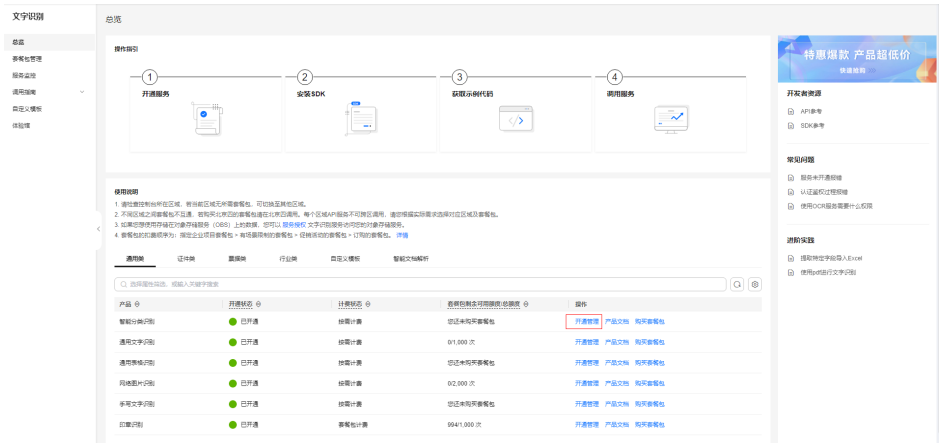


图 3-9 开通管理



说明

- 若主账号授予某个IAM子账号所有权限（OCR FullAccess），则主账号或该IAM子账号均可开通OCR服务。未被授予所有权限的子账号不可自行开通服务。
- 若主账号授予IAM子账号只读访问权限（OCR ReadOnlyAccess）或不授予任何权限，该IAM子账号若需开通OCR服务，则需要主账号（或者被授予所有权限的子账号）进行开通管理。

或者直接单击购买套餐包也可开通服务。计费方式为套餐包付费。

版权所有 © 华为云计算技术有限公司

图 3-10 购买套餐包



4. 服务开通成功后，开通状态显示“已开通”。

图 3-11 已开通



说明

- OCR开通服务时，计费规则默认为“按需计费”，按需计费时，不使用OCR服务，则不收费。如果您购买了套餐包，套餐包扣减规则请参见**计费说明**，套餐包购买后暂不支持退款。
- 如未开通服务，直接调用OCR API会提示ModelArts.4204报错。
- 使用文字识别服务时，可以通过配置OBS访问权限搭配使用。服务只需要配置一次即可，后面使用时无需再次配置，详细信息请参见**配置OBS访问权限**章节。

----结束

3.3 构造请求

本节介绍REST API请求的组成，并以调用**网络图片识别**接口说明如何调用API。

请求 URI

请求URI由如下部分组成。

{URI-scheme} :// {endpoint} / {resource-path} ? {query-string}

版权所有 © 华为云计算技术有限公司

尽管请求URI包含在请求消息头中，但大多数语言或框架都要求您从请求消息中单独传递它，所以在此单独强调。

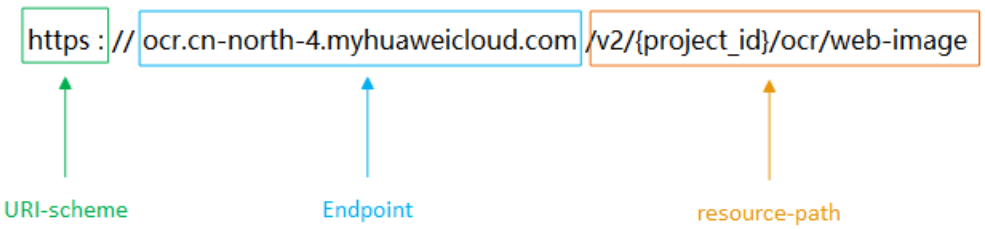
表 3-1 URI 中的参数说明

参数	说明
URI-scheme	表示用于传输请求的协议，当前所有API均采用**HTTPS**协议。
endpoint	指定承载REST服务端点的服务器域名或IP，不同服务不同区域的endpoint不同，您可以从**终端节点**章节查看。 例如OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
resource-path	资源路径，也即API访问路径。 从具体API的URI章节获取，例如**网络图片识别**API的“resource-path”为“v2/{project_id}/ocr/web-image”。
query-string	查询参数，是可选部分，并不是每个API都有查询参数。查询参数前面需要带一个“？”，形式为“参数名=参数取值”，例如“limit=10”，表示查询不超过10条数据。

例如，在“华北-北京四”区域调用**网络图片识别**API，则需使用“华北-北京四”区域的endpoint（ocr.cn-north-4.myhuaweicloud.com），拼接起来如下所示。

```
https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/web-image
```

图 3-12 URI 示意图



说明

为查看方便，在每个具体API的URI部分，只给出resource-path部分，并将请求方法写在一起。这是因为URI-scheme都是HTTPS，而endpoint在同一个区域也相同，所以简洁起见将这两部分省略。

请求方法

HTTP请求方法（也称为操作或动词），它告诉服务你正在请求什么类型的操作。

表 3-2 HTTP 方法

方法	说明
GET	请求服务器返回指定资源。

版权所有 © 华为云计算技术有限公司

方法	说明
PUT	请求服务器更新指定资源。
POST	请求服务器新增资源或执行特殊操作。
DELETE	请求服务器删除指定资源，如删除对象等。
HEAD	请求服务器资源头部。
PATCH	请求服务器更新资源的部分内容。 当资源不存在的时候，PATCH可能会去创建一个新的资源。

例如，**网络图片识别**URI部分，您可以看到其请求方法为“POST”，则其请求为：

```
POST https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/web-image
```

请求消息头

附加请求头字段，如指定的URI和HTTP方法所要求的字段。例如定义消息体类型的请求头“Content-Type”，请求鉴权信息等。

- **Content-Type**：消息体的类型（格式），必选，默认取值为“application/json”，有其他取值时会在具体接口中专门说明。
- **X-Auth-Token**：用户Token，可选，当使用Token方式认证时，必须填充该字段。用户Token请参考认证鉴权章节。

添加消息头后的请求如下所示。

```
POST https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/web-image
Content-Type: application/json
x-auth-token: MIIaBgYJKoZIhvcNAQcC…
```

请求消息体

请求消息体通常以结构化格式发出，与请求消息头中Content-type对应，传递除请求消息头之外的内容。若请求消息体中参数支持中文，则中文字符必须为UTF-8编码。

每个接口的请求消息体内容不同，也并不是每个接口都需要有请求消息体（或者说消息体为空），GET、DELETE操作类型的接口就不需要消息体，消息体具体内容需要根据具体接口而定。

例如，对于**网络图片识别**接口，您可以从接口的请求部分看到所需的请求参数及参数说明。将消息体加入后的请求如下所示。

```
POST https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/web-image
Request Header:
Content-Type: application/json
X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAgABwESAAMAA..."
}
```

到这里为止这个请求需要的内容就具备齐全了，您可以使用curl、Postman或直接编写代码等方式发送请求调用API。

版权所有 © 华为云计算技术有限公司

3.4 认证鉴权

调用接口有如下两种认证方式，您可以选择其中一种进行认证鉴权。

- Token认证：通过Token认证调用请求。
- AK/SK认证：通过AK（Access Key ID）/SK（Secret Access Key)加密调用请求。

Token 认证

Token在计算机系统中代表令牌（临时）的意思，拥有Token就代表拥有某种权限。Token认证就是在调用API的时候将Token加到请求消息头，从而通过身份认证，获得操作API的权限。

说明

- Token的有效期为24小时，需要使用一个Token鉴权时，可以先缓存起来，避免频繁调用。
- 如果您的华为云账号已升级为华为账号，将不支持获取账号Token。建议为您自己创建一个IAM用户，获取IAM用户的Token。详细说明请参见**帐密报错**。

Token可通过调用“获取用户Token”接口获取，调用本服务API需要project级别的Token，即调用获取用户Token接口时，请求body中“auth.scope”的取值需要选择“project”，请求示例如下所示。

username、***domainname***、***project name***可登录控制台**“我的凭证 > API凭证”**页面获取。***password***为用户密码。



- 伪码

```
POST https://iam.cn-north-4.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
{
    "auth": {
        "identity": {
            "methods": [
                "password"
            ],
            "password": {
                "user": {
                    "name": "username", //IAM用户名
                    "password": "********", //密码
                    "domain": {
                        "name": "domainname" //账号名
                    }
                }
            }
        },
        "scope": {
            "project": {
```


版权所有 © 华为云计算技术有限公司

```
                "name": "project name" //替换为实际的project name，如cn-north-4
            }
        }
    }
}
```

- Java

```
package xxx; // GetToken类所在的项目路径

import okhttp3.MediaType;
import okhttp3.OkHttpClient;
import okhttp3.Request;
import okhttp3.RequestBody;
import okhttp3.Response;

public class GetToken {
    public static void main(String[] args) throws Exception {
        OkHttpClient client = new OkHttpClient().newBuilder().build();
        MediaType mediaType = MediaType.parse("application/json");
        String requestBody //填写正确的账号名、IAM用户名、密码
            = "{\"auth\": {\"identity\": {\"methods\": [\"password\"],\"password\": {\"user\": {\"name\":
\"********\",\"password\": \"********\",\"domain\": {\"name\": \"********\"}}}},\"scope\": {\"project\":
{\"name\": \"cn-north-4\"}}}}";
        RequestBody body = RequestBody.create(requestBody, mediaType);
        Request request = new Request.Builder().url("https://iam.cn-north-4.myhuaweicloud.com/v3/
auth/tokens")
            .method("POST", body)
            .addHeader("Content-Type", "application/json")
            .build();
        Response response = client.newCall(request).execute();
        System.out.println(response.header("X-Subject-Token"));
    }
}
```

- Python

```
import requests
import json

url = "https://iam.cn-north-4.myhuaweicloud.com/v3/auth/tokens"
payload = json.dumps({
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",
          "password": "********",
          "domain": {
            "name": "domainname"
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "cn-north-4"
      }
    }
  }
})
headers = {
  'Content-Type': 'application/json'
}

response = requests.request("POST", url, headers=headers, data=payload)

print(response.headers["X-Subject-Token"])
```


版权所有 © 华为云计算技术有限公司

如下图所示，返回的响应消息头中“x-subject-token”就是需要获取的用户Token。获取Token之后，您就可以使用Token认证调用OCR服务API。

图 3-13 获取 Token 示例

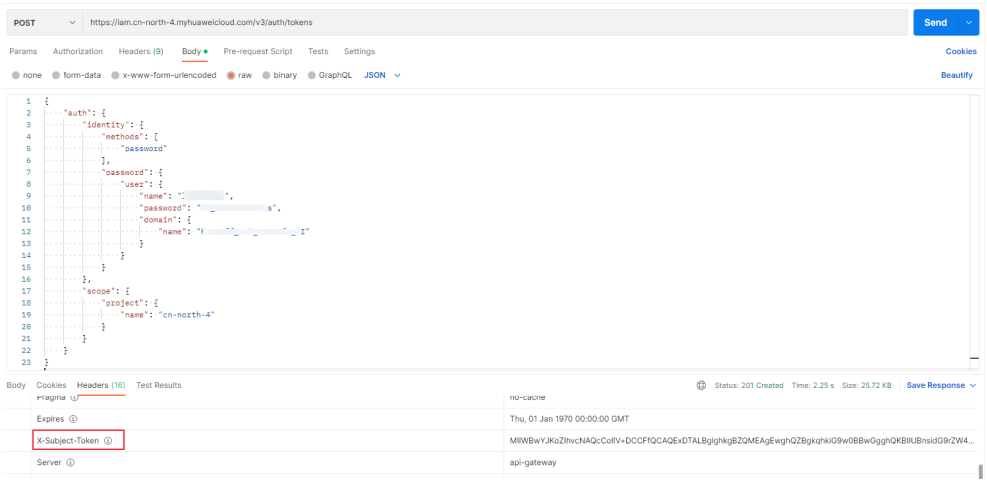


图 3-14 获取用户 Token 响应消息头



AK/SK 认证

说明

AK/SK签名认证方式仅支持消息体大小12M以内，12M以上的请求请使用Token认证。

AK/SK认证就是使用AK/SK对请求进行签名，在请求时将签名信息添加到消息头，从而通过身份认证。

- AK(Access Key ID)：访问密钥ID。与私有访问密钥关联的唯一标识符；访问密钥ID和私有访问密钥一起使用，对请求进行加密签名。
- SK(Secret Access Key)：与访问密钥ID结合使用的密钥，对请求进行加密签名，可标识发送方，并防止请求被修改。

版权所有 © 华为云计算技术有限公司

使用AK/SK认证时，您可以基于签名算法使用AK/SK对请求进行签名，也可以使用专门的签名SDK对请求进行签名。详细的签名方法和SDK使用方法请参见**API签名指南**。

如果之前没有生成过AK/SK，可登录**“我的凭证”**界面，选择“管理访问密钥 > 新增访问密钥”来获取。

须知

签名SDK只提供签名功能，与服务提供的SDK不同，使用时请注意。

AK/SK获取方式请参考**获取AK/SK**。

3.5 返回结果

状态码

请求发送以后，您会收到响应，包含状态码、响应消息头和消息体。

状态码是一组数字代码，状态码表示了请求响应的状态，如果调用后返回状态码为2xx，则表示调用成功。完整的状态码列表请参见**状态码**。

响应消息头

对应请求消息头，响应同样也有消息头，如“Content-type”。例如**表3-3**所示。

表 3-3 响应消息头

名称	说明
Content-Length	响应消息体的字节长度，单位为Byte。
Date	系统响应的时间。
Content-Type	响应消息体的MIME类型。

响应消息体

响应消息体通常以结构化格式返回，与响应消息头中Content-type对应，传递除响应消息头之外的内容。

对于**网络图片识别**接口，返回如下消息体。

```
{
  "result": {
     "words_block_count": 2,
     "words_block_list": [
        {
           "words": "文字块1",
           "confidence": 0.9950,
           "location": [
              [13, 476],
              [91, 332],
              [125, 351],
              [48, 494]
```


版权所有 © 华为云计算技术有限公司

```
            ]
        },
        {
            "words": "文字块2",
            "confidence": 0.9910,
            "location": [
                [13, 476],
                [91, 332],
                [125, 351],
                [48, 494]
            ]
        }
    ],
    "extracted_data": {}
 }
}
```

当接口调用出错时，会返回错误码及错误信息说明，错误响应的Body体格式如下所示。

```
{
   "error_code": "AIS.0103",
   "error_msg": "The image size does not meet the requirements."
}
```

其中，error_code表示错误码，error_msg表示错误描述信息。

 版权所有 © 华为云计算技术有限公司

4 API

4.1 智能文档解析

功能介绍

对证件、票据、表单等多版式文档进行文字识别、版面分析、键值对提取以及表格识别等任务，并以json格式返回识别的结构化结果。

约束与限制

- 支持中英文及部分中文繁体字。
- 只支持识别PNG、JPG、JPEG、BMP、GIF、TIFF、WEBP、PCX、ICO、PSD格式的图片以及PDF文档，PDF只支持单页识别，可通过pdf_page_number参数指定页码。
- 图像各边的像素大小在15px到30000px之间。单个图片、PDF文件其对应的Base64编码不超过10MB。
- 图像中识别区域有效占比超过80%，保证所有文字及其边缘包含在图像内。
- 支持图像任意角度的水平旋转。
- 单页字符数不大于1800，以获得较优的识别效果。
- 目前不支持复杂背景（如户外自然场景、防伪水印等）和文字扭曲图像的文档解析。
- 文字识别服务属于公有云服务，线上用户资源共享，如果需要多并发请求，请提前**联系我们**。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

版权所有 © 华为云计算技术有限公司

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/smart-document-recognizer

表 4-1 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-2 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 Token认证就是在调用API的时候将Token加到请求消息头，从而通过身份认证，获得操作API的权限，响应消息头中X-Subject-Token的值即为Token。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。 企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-3 请求 Body 参数

参数	是否必选	参数类型	描述
data	否	String	与url二选一。单个图片、PDF文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。图像尺寸不小于15×15像素，最长边不超过30000像素，支持JPG/PNG/BMP/TIFF格式。PDF以150dpi的分辨率转为图像进行文档解析，需符合上述图像尺寸规定。若PDF有多页，当前仅对第1页进行识别。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
url	否	String	与data二选一。url中单个图片、PDF文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
single_orientation_mode	否	Boolean	单朝向模式开关。可选值包括： • true：打开单朝向模式 • false：关闭单朝向模式 图片文字方向一致时，打开该开关可提升识别精度；图片文字方向不一致时，关闭该开关可支持多朝向文字识别。未传入该参数时默认为true，既默认图片中的字段为单朝向。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
language	否	String	语种选择，未传入该参数时默认为中英文识别模式。 ● zh：中英文 ● es：西班牙语 ● ar：阿拉伯语 ● pt：葡萄牙语 ● tr：土耳其语 ● th：泰语 **说明** ● 表格、表单识别和版面分析支持西班牙语、阿拉伯语、葡萄牙语、土耳其语和泰语识别。 ● 键值对提取能力（KV）仅支持中英文模式，暂不支持其他语种。
kv	否	Boolean	是否进行键值对（key-value）提取。若是，结果会以“kv_result”这一关键字返回。
table	否	Boolean	是否进行表格识别。此处表格特指逻辑表格，通常具有M行N列的形式，且第一行或第一列为表头。若是，结果会以“table_result”这一关键字返回。
layout	否	Boolean	是否进行版面分析。若是，结果会以“layout_result”这一关键字返回。
return_excel	否	Boolean	仅当table为True时有效。是否返回表格转换Microsoft Excel的Base64编码字段。
form	否	Boolean	是否进行有线表单识别。有线表单指关键信息以有线单元格形式进行呈现，例如户口本、机动车发票等。若是，结果会以"form_result"这一关键字返回。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
formula	否	Boolean	是否进行公式识别，识别结果为latex序列。若是，结果会以“formula_result”这一关键字返回。 • 开启公式识别后会降低响应速度。 • 当前仅支持3行以内公式识别，不支持3行以上的多行公式。
kv_map	否	String	需要传入字典的json序列化后字符串，用于对kv_result中的特定key值进行归一化映射。例如，kv_result中包含{"名称"：“小明"}的键值对，若传入{"名称"：“姓名"}的kv_map，则返回结果为{“姓名”：“小明”}。 **说明** 参数传入示例： • "kv_map":"{"名称":"姓名"}"
erase_seal	否	Boolean	是否进行印章擦除。开启后，可提升印章遮挡区域的文字识别精度。
pdf_page_number	否	Integer	指定PDF页码识别。传入该参数时，则识别指定页码的内容。如果不传该参数，则默认识别第1页。

响应参数

状态码： 200

表 4-4 响应 Body 参数

参数	参数类型	描述
result	Array of **SmartDocumentRecognizerResult** objects	调用成功时返回的结果列表，按页面顺序返回，列表第一项为第一页识别结果，依次类推。 调用失败时无此字段。

版权所有 © 华为云计算技术有限公司

表 4-5 SmartDocumentRecognizerResult

参数	参数类型	描述
ocr_result	**SmartDocumentRecognizerOcrResult** object	文字识别结果。
kv_result	**SmartDocumentRecognizerKvResult** object	键值对提取结果。当输入参数“kv”为“true”时，才返回该参数。
table_result	**SmartDocumentRecognizerTableResult** object	表格识别结果。当输入参数“table”为“true”时，才返回该参数。
layout_result	**SmartDocumentRecognizerLayoutResult** object	版面分析结果。当输入参数“layout”为“true”时，才返回该参数。
form_result	**SmartDocumentRecognizerFormResult** object	有线表单识别结果。当输入参数"form"为true时，才返回该参数。
formula_result	**SmartDocumentRecognizerFormulaResult** object	公式识别结果。

表 4-6 SmartDocumentRecognizerOcrResult

参数	参数类型	描述
direction	Float	图片朝向。
words_block_count	Integer	识别文字块数目。
words_block_list	Array of **SmartDocumentRecognizerWordsBlockList** objects	识别文字块列表，输出顺序从左到右，先上后下。

版权所有 © 华为云计算技术有限公司

表 4-7 SmartDocumentRecognizerWordsBlockList

参数	参数类型	描述
words	String	文字块识别结果。
location	Array<Array<Integer>>	文字块的区域位置信息，列表形式，包含文字区域四个顶点的二维坐标（x,y）;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。 **说明** 输入数据格式是PDF时，返回的字段坐标仅用于参考，表示字段间的相对位置关系。
confidence	Float	文字块识别结果的置信度。

表 4-8 SmartDocumentRecognizerKvResult

参数	参数类型	描述
kv_block_count	Integer	模型识别到的键值对数量。
kv_block_list	Array of **SmartDocumentRecognizerKVBlock** objects	键值对识别结果列表。

表 4-9 SmartDocumentRecognizerKVBlock

参数	参数类型	描述
key	String	key-value对（键值对）中的key，例如“姓名：小明”中的“姓名”
value	String	key-value对（键值对）中的value，例如“姓名：小明”中的“小明”
words_block_count	Integer	该键值对中所包含的文本框数量。
words_block_list	Array of **SmartDocumentRecognizerKVWordsBlock** objects	文本框识别结果列表。

版权所有 © 华为云计算技术有限公司

表 4-10 SmartDocumentRecognizerKVWordsBlock

参数	参数类型	描述
words	String	文字块识别结果。
location	Array<Array<Integer>>	文字块的区域位置信息，列表形式，包含文字区域四个顶点的二维坐标（x,y）;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。 **说明** 输入数据格式是PDF时，返回的字段坐标仅用于参考，表示字段间的相对位置关系。
type	String	类型。

表 4-11 SmartDocumentRecognizerTableResult

参数	参数类型	描述
table_count	Integer	模型识别到的表格数量。
table_list	Array of **SmartDocumentRecognizerTableBlock** objects	表格识别结果列表。

表 4-12 SmartDocumentRecognizerLayoutResult

参数	参数类型	描述
layout_block_count	Integer	模型识别到的文档版面区域数量。
layout_block_list	Array of **SmartDocumentRecognizerLayoutBlock** objects	文档版面区域识别结果列表。

表 4-13 SmartDocumentRecognizerLayoutBlock

参数	参数类型	描述
location	Array<Array<Integer>>	文字块的区域位置信息，列表形式，包含文字区域四个顶点的二维坐标（x,y）;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。 **说明** 输入数据格式是PDF时，返回的字段坐标仅用于参考，表示字段间的相对位置关系。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
type	String	文档区域类别，包含text、title、sub_title、image、image_caption、form、table、table_caption、header、footer、page_number、reference、formula、stamp、directory共15个类别。
text	String	文档区域文字内容。对于表格与图像，不返回其中的文字内容。
words_ids	Array of integers	文字识别结果索引列表，表示ocr_result的words_block_list中哪些文本框位于该文档区域内。
table_id	Integer	仅当type为"table"且入参table为True时返回该字段，表示当前逻辑表格区域对应table_result中哪一项识别结果。
form_id	Integer	仅当type为"form"且入参form为True时返回该字段，表示当前有线表单区域对应form_result中哪一项识别结果。

表 4-14 SmartDocumentRecognizerFormResult

参数	参数类型	描述
form_count	Integer	模型识别到的有线表单数量。
form_list	Array of **SmartDocumentRecognizerTableBlock** objects	有线表单识别结果列表。

表 4-15 SmartDocumentRecognizerTableBlock

参数	参数类型	描述
location	Array<Array<Integer>>	当前表格的位置信息，列表形式，分别表示文字块4个顶点的x, y坐标；坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。 **说明** 输入数据格式是PDF时，返回的表格坐标仅用于参考，表示表格的相对位置关系。
words_block_count	Integer	表格中所包含的单元格数量。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
words_block_list	Array of **SmartDocumentRecognizerTableWordsBlock** objects	单元格识别结果列表。
excel	String	表格识别结果的base64编码，仅当return_excel为True时返回该字段。对返回的excel编码可用base64.b64decode解码并保存为.xlsx文件。

表 4-16 SmartDocumentRecognizerTableWordsBlock

参数	参数类型	描述
words	String	单元格内的文字识别结果。
rows	Array of integers	文字块占用的行信息，编号从0开始，列表形式，数据类型为Integer。
columns	Array of integers	文字块占用的列信息，编号从0开始，列表形式，数据类型为Integer。

表 4-17 SmartDocumentRecognizerFormulaResult

参数	参数类型	描述
formula_count	Integer	数学公式数量。
formula_list	Array of **SmartDocumentRecognizerFormulaBlock** objects	数学公式识别结果列表。

表 4-18 SmartDocumentRecognizerFormulaBlock

参数	参数类型	描述
formula	String	数学公式识别结果，以latex字符串表示。
location	Array<Array<Integer>>	数学公式位置信息，列表形式，分别表示4个顶点的x, y坐标；坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。 **说明** 输入数据格式是PDF时，返回的数学公式坐标仅用于参考，表示数学公式的相对位置关系。

版权所有 © 华为云计算技术有限公司

状态码： 400

表 4-19 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。
  例如，智能文档解析部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/smart-document-recognizer”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 传入文档图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/smart-document-recognizer

{
  "data" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入文档图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/smart-document-recognizer

{
  "url" : "https://BucketName.obs.myhuaweicloud.com/ObjectName"
}
```

响应示例

状态码： 200

成功响应示例

```
{
  "result" : [ {
    "formula_result" : {
      "formula_count" : 1,
      "formula_list" : [ {
        "formula" : "\\\\int _ { L } \\\\left ( 2 x y ^ { 3 } - y ^ { 2 } \\\\cos x \\\\right ) \\\\mathrm { d } x + \\\\left ( 1 - 2 y \\\\sin x + 3 x ^ { 2 } y ^ { 2 } \\\\right ) \\\\mathrm { d } y",
        "location" : [ [ 171, 919 ], [ 950, 919 ], [ 950, 967 ], [ 171, 967 ] ]
      } ]
    }
  }, {
    "layout_result" : {
      "layout_block_count" : 19,
      "layout_block_list" : [ {
        "location" : [ [ 1165, 368 ], [ 2031, 368 ], [ 2031, 465 ], [ 1165, 465 ] ],
        "type" : "title",
        "text" : "黑龙江增值税专用发票",
        "words_ids" : [ 0 ]
```


版权所有 © 华为云计算技术有限公司

```
    }, {
      "location" : [ [ 15, 19 ], [ 1078, 19 ], [ 1078, 637 ], [ 15, 637 ] ],
      "type" : "form",
      "text" : "xxxx",
      "words_ids" : [ 2, 3, 4 ],
      "form_id" : 0
    }, {
      "location" : [ [ 18, 180 ], [ 1077, 180 ], [ 1077, 636 ], [ 18, 636 ] ],
      "type" : "table",
      "text" : "xxxx",
      "words_ids" : [ 0, 1, 2 ],
      "table_id" : 0
    } ]
  }
}, {
  "form_result" : {
    "form_count" : 1,
    "form_list" : [ {
      "location" : [ [ 15, 19 ], [ 1074, 19 ], [ 1074, 636 ], [ 15, 636 ] ],
      "words_block_count" : 24,
      "words_block_list" : [ {
        "words" : "xxx",
        "rows" : [ 0 ],
        "columns" : [ 0, 1, 2 ]
      }, {
        "words" : "xxxx",
        "rows" : [ 1 ],
        "columns" : [ 0, 1, 2 ]
      } ],
      "excel" : "UEsDBBQAAAAIAAAAIQBhXUk6TwEAAI8EAAATAAAAW0NvbnRlbnRfVHlwZX..."
    } ]
  }
}, {
  "table_result" : {
    "table_count" : 1,
    "table_list" : [ {
      "words_block_count" : 24,
      "words_block_list" : [ {
        "words" : "货物或应税劳务、服务名称",
        "rows" : [ 0 ],
        "columns" : [ 0 ]
      }, {
        "words" : "规格型号",
        "rows" : [ 0 ],
        "columns" : [ 1 ]
      } ],
      "excel" : "xxxx",
      "location" : [ [ 275, 967 ], [ 2919, 967 ], [ 2919, 1177 ], [ 275, 1177 ] ]
    } ]
  }
}, {
  "kv_result" : {
    "kv_block_count" : 25,
    "kv_block_list" : [ {
      "key" : "开票日期",
      "value" : "2017年08月31日",
      "words_block_count" : 2,
      "words_block_list" : [ {
        "words" : "开票日期",
        "location" : [ [ 2241, 589 ], [ 2480, 592 ], [ 2480, 646 ], [ 2241, 643 ] ],
        "type" : "key"
      }, {
        "words" : "2017年08月31日",
        "location" : [ [ 2479, 591 ], [ 2850, 595 ], [ 2850, 649 ], [ 2479, 645 ] ],
        "type" : "value"
      } ]
    } ]
  }
}, {
```


版权所有 © 华为云计算技术有限公司

```
  "ocr_result" : {
    "direction" : 0.4767,
    "words_block_count" : 67,
    "words_block_list" : [ {
      "words" : "黑龙江增值税专用发票",
      "location" : [ [ 430, 100 ], [ 874, 99 ], [ 874, 139 ], [ 430, 141 ] ],
      "confidence" : 0.9552
    } ]
  }
 } ]
}
```

状态码： 400

失败响应示例

```
{
 "error_code" : "AIS.0103",
 "error_msg" : "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入文档图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeSmartDocumentRecognizerSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeSmartDocumentRecognizerRequest request = new RecognizeSmartDocumentRecognizerRequest();
        SmartDocumentRecognizerRequestBody body = new SmartDocumentRecognizerRequestBody();
```

 版权所有 © 华为云计算技术有限公司

```
        body.withData("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeSmartDocumentRecognizerResponse response =
client.recognizeSmartDocumentRecognizer(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入文档图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeSmartDocumentRecognizerSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeSmartDocumentRecognizerRequest request = new
RecognizeSmartDocumentRecognizerRequest();
        SmartDocumentRecognizerRequestBody body = new SmartDocumentRecognizerRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeSmartDocumentRecognizerResponse response =
client.recognizeSmartDocumentRecognizer(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
```

 版权所有 © 华为云计算技术有限公司

```
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入文档图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeSmartDocumentRecognizerRequest()
        request.body = SmartDocumentRecognizerRequestBody(
            data="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_smart_document_recognizer(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入文档图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
```

 版权所有 © 华为云计算技术有限公司

```
        .build()

    try:
        request = RecognizeSmartDocumentRecognizerRequest()
        request.body = SmartDocumentRecognizerRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_smart_document_recognizer(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入文档图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeSmartDocumentRecognizerRequest{}
    dataSmartDocumentRecognizerRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.SmartDocumentRecognizerRequestBody{
        Data: &dataSmartDocumentRecognizerRequestBody,
    }
    response, err := client.RecognizeSmartDocumentRecognizer(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入文档图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
```

```
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeSmartDocumentRecognizerRequest{}
    urlSmartDocumentRecognizerRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.SmartDocumentRecognizerRequestBody{
        Url: &urlSmartDocumentRecognizerRequestBody,
    }
    response, err := client.RecognizeSmartDocumentRecognizer(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

版权所有 © 华为云计算技术有限公司

4.2 通用表格识别

功能介绍

识别表格图片中的文字内容，并将识别的结果以JSON格式返回给用户。返回结果将图片区域（words_region）划分为两类：纯文本区（text）和表格区（table），并返回表格结构（row, column）和文本信息。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-1 通用表格示例图

门诊检验报告单			
**血常规（5 分类）			
标本状态：正常	临床诊断：1.慢性扁桃体炎		
检验项目	结果	参考范围	单位
中性细胞百分率（NEL%）	77.1	40-75	%
淋巴细胞百分率（LYM%）	8.8	20-50	%
单核细胞百分率（MONO%）	7.1	3.0-10.0	%
红细胞计数（RBC）	6.66	4.3-5.8	%
送检医生：	检验者：	审核者：	

约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中识别区域有效占比超过80%，保证整张表格及其边缘包含在图像内。
- 支持图像任意角度的水平旋转。
- 目前不支持复杂背景（如户外自然场景、防伪水印等）和表格线扭曲图像的文字识别。
- 支持中英文以及部分繁体字。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

版权所有 © 华为云计算技术有限公司

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/general-table

表 4-20 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-21 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

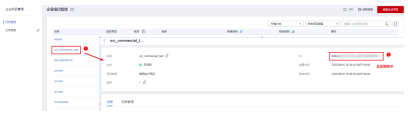
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-22 请求 Body 参数

参数	是否必选	类型	说明
image	否	String	该参数与url二选一。 单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	类型	说明
url	否	String	该参数与image二选一。 url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url例如https://support.huaweicloud.com/api-ocr/zh-cn_image_0282767866.png • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
return_text_location	否	Boolean	返回文本块坐标及单元格坐标信息，可选值如下所示。 • true：返回文本块和单元格坐标 • false：不返回 如果未传入该参数时默认为false，即不返回。
return_char_location	否	Boolean	返回单字符的坐标信息，可选值包括： • true：返回单字符的坐标 • false：不返回 未传入该参数时默认为false，即不返回。如果此参数为true时，return_text_location必须为true
return_confidence	否	Boolean	是否返回置信度的开关，可选值如下所示。 • true：返回置信度 • false：不返回置信度 如果未传入该参数，系统默认为“false”，即不返回置信度。
return_excel	否	Boolean	是否返回表格转换Microsoft Excel的Base64编码字段。可选值包括： • true: 返回'excel'字段，表示xlsx格式的表格识别结果的Base64编码 • false: 不返回。默认为false 对返回的Excel编码，可用Python函数base64.b64decode解码后保存为.xlsx文件。

版权所有 © 华为云计算技术有限公司

参数	是否必选	类型	说明
return_rectification_matrix	否	Boolean	可选值包括： • true：返回透视变换矩阵 • false：不返回 未传入该参数时默认为false，即不返回透视变换矩阵。
with_borders	否	Boolean	可选值包括： • true：输入图像仅包含有线表格，仅进行有线表格识别。 • false: 输入图像可能包含无线表格，同时进行有线表格与无线表格识别。 未传入该参数时默认为false，即同时进行有线表格与无线表格识别。当确认输入仅包含有线表格时，该参数设为true可达到更优识别效果。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-23 响应 Body 参数

参数	参数类型	描述
result	**GeneralTableResult** object	调用成功时表示调用结果。 调用失败时无此字段。

表 4-24 GeneralTableResult

参数	参数类型	描述
words_region_count	Integer	文字区域数目。
words_region_list	Array of **WordsRegionList** objects	文字区域识别结果列表，输出顺序从左到右，先上后下。
excel	String	表格图像转换为excel的base64编码，图像中的文字和表格按位置写入excel。对返回的excel编码可用base64.b64decode解码并保存为.xlsx文件。

版权所有 © 华为云计算技术有限公司

表 4-25 WordsRegionList

参数	参数类型	描述
type	String	文字识别区域类型。 ● text：文本识别区域; ● table：表格识别区域。
words_block_count	Integer	子区域识别文字块数目。
words_block_list	Array of **GeneralTableWordsBlockList** objects	子区域识别文字块列表，输出顺序从左到右，先上后下。

表 4-26 GeneralTableWordsBlockList

参数	参数类型	描述
words	String	文字块识别结果。
confidence	Float	字段的平均置信度，置信度越大，表示本次识别的对应字段的可靠性越高，在统计意义上，置信度越大，准确率越高。置信度由算法给出，不直接等价于对应字段的准确率。
location	Array<Array<Integer>>	文字块位置信息，列表形式，分别表示文字块4个顶点的x, y坐标;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。
words_list	Array of **WordsListItem** objects	单元格内文字段列表。输出顺序从左到右，从上到下。仅当入参"return_text_location"为true时存在。
rows	Array of integers	文字块占用的行信息，编号从0开始，列表形式，数据类型为Integer。仅在表格区域内有效，即type字段为"table"时该字段有效。
columns	Array of integers	文字块占用的列信息，编号从0开始，列表形式，数据类型为Integer。仅在表格区域内有效，即type字段为"table"时该字段有效。
cell_location	Array<Array<Integer>>	单元格位置信息，列表形式，分别表示单元格4个顶点的x, y坐标;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。

表 4-27 WordsListItem

参数	参数类型	描述
words	String	文字块识别结果。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
confidence	Float	字段的平均置信度，置信度越大，表示本次识别的对应字段的可靠性越高，在统计意义上，置信度越大，准确率越高。置信度由算法给出，不直接等价于对应字段的准确率。
location	Array<Array<Integer>>	文字块位置信息，列表形式，分别表示文字块4个顶点的x, y坐标;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。
char_list	Array of **CharListItem** objects	单元格内文字段列表。输出顺序从左到右，从上到下。仅当入参"return_text_location"和"return_char_location"同时为true时存在。

表 4-28 CharListItem

参数	参数类型	描述
char	String	单字符识别结果。
char_confidence	Float	单字符的置信度，置信度越大，表示本次识别的文字的可靠性越高，在统计意义上，置信度越大，准确率越高。置信度由算法给出，不直接等价于对应字段的准确率。
char_location	Array<Array<Integer>>	单字符的位置信息，列表形式，分别表示文字块4个顶点的x, y坐标;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。

状态码： 400

表 4-29 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

版权所有 © 华为云计算技术有限公司

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，通用表格识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/general-table”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入表格图片的base64编码文字识别，并不返回置信度信息

```
POST https://{endpoint}/v2/{project_id}/ocr/general-table
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
   "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAg...",
   "return_confidence":false
 }
```

- 传入表格图片的url进行文字识别，并不返回置信度信息

```
POST https://{endpoint}/v2/{project_id}/ocr/general-table
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
    "url":"https://BucketName.obs.xxxx.com/ObjectName",
    "return_confidence":false
 }
```

响应示例

状态码： 200

成功响应示例

```
{
  "result" : {
    "words_region_count" : 2,
    "words_region_list" : [ {
      "type" : "text",
      "words_block_count" : 1,
      "words_block_list" : [ {
        "words" : "文字区域识别文字块1",
        "confidence" : 0.9991
      } ]
    }, {
      "type" : "table",
      "words_block_count" : 2,
      "words_block_list" : [ {
        "words" : "表格区域识别文字块1",
        "confidence" : 0.9942,
        "rows" : [ 0 ],
        "columns" : [ 0 ]
      }, {
        "words" : "表格区域识别文字块2",
        "confidence" : 0.914,
        "rows" : [ 0 ],
        "columns" : [ 1, 2 ]
      } ]
```


版权所有 © 华为云计算技术有限公司

```
  } ]
 }
}
```

状态码： 400

失败响应示例

```
{
  "result" : {
    "error_code" : "AIS.0103",
    "error_msg" : "The image size does not meet the requirements."
  }
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入表格图片的base64编码进行文字识别，并不返回置信度信息

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeGeneralTableSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeGeneralTableRequest request = new RecognizeGeneralTableRequest();
        GeneralTableRequestBody body = new GeneralTableRequestBody();
        body.withReturnConfidence(false);
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeGeneralTableResponse response = client.recognizeGeneralTable(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```


版权所有 © 华为云计算技术有限公司

```
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入表格图片的url进行文字识别，并不返回置信度信息

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeGeneralTableSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeGeneralTableRequest request = new RecognizeGeneralTableRequest();
        GeneralTableRequestBody body = new GeneralTableRequestBody();
        body.withReturnConfidence(false);
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeGeneralTableResponse response = client.recognizeGeneralTable(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

 版权所有 © 华为云计算技术有限公司

Python

- 传入表格图片的base64编码进行文字识别，并不返回置信度信息

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeGeneralTableRequest()
        request.body = GeneralTableRequestBody(
            return_confidence=False,
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_general_table(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入表格图片的url进行文字识别，并不返回置信度信息

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeGeneralTableRequest()
        request.body = GeneralTableRequestBody(
            return_confidence=False,
```

 版权所有 © 华为云计算技术有限公司

```
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_general_table(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入表格图片的base64编码进行文字识别，并不返回置信度信息

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeGeneralTableRequest{}
    returnConfidenceGeneralTableRequestBody:= false
    imageGeneralTableRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.GeneralTableRequestBody{
        ReturnConfidence: &returnConfidenceGeneralTableRequestBody,
        Image: &imageGeneralTableRequestBody,
    }
    response, err := client.RecognizeGeneralTable(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入表格图片的url进行文字识别，并不返回置信度信息

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
```

 版权所有 © 华为云计算技术有限公司

```
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeGeneralTableRequest{}
    returnConfidenceGeneralTableRequestBody:= false
    urlGeneralTableRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.GeneralTableRequestBody{
        ReturnConfidence: &returnConfidenceGeneralTableRequestBody,
        Url: &urlGeneralTableRequestBody,
    }
    response, err := client.RecognizeGeneralTable(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

版权所有 © 华为云计算技术有限公司

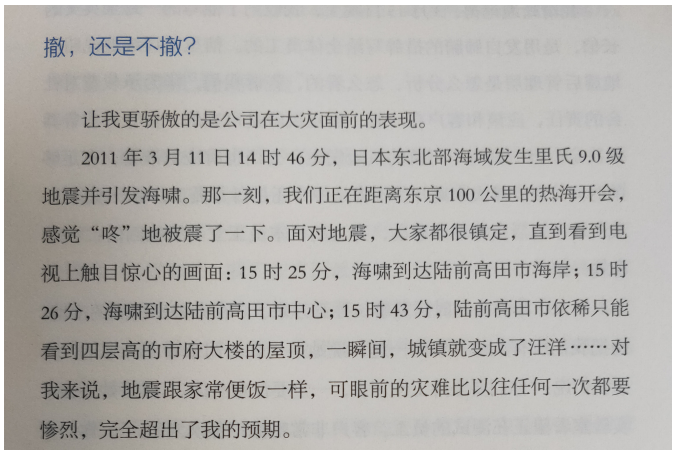
4.3 通用文字识别

功能介绍

识别图片上的文字信息，以JSON格式返回识别的文字和坐标。支持扫描文件、电子文档、书籍、票据和表单等多种场景的文字识别。

支持中英文以及部分繁体字。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-2 通用文字示例图



约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、GIF、TIFF、WEBP、PCX、ICO、PSD、PDF格式图片。
- 图像各边的像素大小在15px到30000px之间，图像高*宽的总像素数不能大于1.6亿。单个图片、PDF文件其对应的Base64编码不超过10MB。
- 图像中识别区域有效占比超过80%，保证所有文字及其边缘包含在图像内。
- 支持图像任意角度的水平旋转。
- 支持自动过滤浅色文字水印。
- 目前不支持复杂背景（如户外自然场景等）和文字扭曲图像的文字识别。
- 支持中英文以及部分繁体字、马来语、乌克兰语、印地语、俄语、越南语、印尼语、泰语、阿拉伯语、德语、拉丁语、法语、意大利语、西班牙语、葡萄牙语、罗马尼亚语、波兰语、阿姆哈拉语、日语、韩语、土耳其语、挪威语、丹麦语、瑞典语、柬埔寨语、希伯来语识别。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

版权所有 © 华为云计算技术有限公司

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/general-text

表 4-30 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-31 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

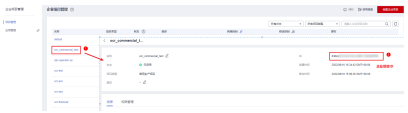
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-32 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 Base64编码，要求单个图片、PDF文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图片最短边不小于15px，最长边不超过30000px。支持JPEG、JPG、PNG、BMP、GIF、TIFF、WEBP、PCX、ICO、PSD、PDF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
url	否	String	该参数与image二选一。 单个图片、PDF文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图片的url路径目前支持： • 公网http/https url例如https://support.huaweicloud.com/api-ocr/zh-cn_image_0288038182.png • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
detect_direction	否	Boolean	是否校正图片的倾斜角度，可选值如下。 • true：校正图片的倾斜角度 • false：不校正图片的倾斜角度 支持任意角度的校正，未传入该参数时默认为“false”。 待识别图片如果存在倾斜，建议将此参数设置为“true”。
quick_mode	否	Boolean	快速模式开关，针对单行文字图片（要求图片只包含一行文字，且文字区域占比超过50%），打开时可以更快返回识别内容。可选值如下所示。 • true：打开快速模式 • false：关闭快速模式 未传入该参数时默认为false，即关闭快速模式。
character_mode	否	Boolean	单字符模式开关。可选值包括： • true：打开单字符模式 • false：关闭单字符模式 未传入该参数时默认为false，即不返回单个文本行的单字符信息。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
language	否	String	语种选择，未传入该参数时默认为中英文识别模式。 ● auto：自动语种分类 ● ms：马来语 ● uk：乌克兰语 ● hi：印地语 ● ru：俄语 ● vi：越南语 ● id：印尼语 ● th：泰语 ● zh：中英文 ● ar：阿拉伯语 ● de：德语 ● la：拉丁语 ● fr：法语 ● it：意大利语 ● es：西班牙语 ● pt：葡萄牙语 ● ro：罗马尼亚语 ● pl：波兰语 ● am：阿姆哈拉语 ● ja：日语 ● ko：韩语 ● tr：土耳其语 ● no：挪威语 ● da：丹麦语 ● sv：瑞典语 ● km：柬埔寨语 ● he：希伯来语
single_orientation_mode	否	Boolean	单朝向模式开关。可选值包括： ● true：打开单朝向模式 ● false：关闭单朝向模式 未传入该参数时默认为false，即默认图片中的字段为多朝向。
pdf_page_number	否	Integer	指定PDF页码识别。传入该参数时，则识别指定页码的内容。如果不传该参数，则默认识别第1页。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
return_markdown_result	否	Boolean	返回文字块拼接结果开关。可选值包括： • true：打开返回文字块拼接结果开关； • false：关闭返回文字块拼接结果开关。 未传入该参数时默认为false，既默认关闭返回文字块拼接结果开关。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-33 响应 Body 参数

参数	参数类型	描述
result	**GeneralTextResult** object	识别结果。 调用失败时不返回此字段。

表 4-34 GeneralTextResult

参数	参数类型	描述
direction	Float	图片朝向。 • 当“detect_direction”为“true”时，该字段有效。返回图片逆时针旋转角度，值区间为0~359。 • 当“detect_direction”为“false”时，该字段值为 -1。
words_block_count	Integer	检测到的文字块数目。
words_block_list	Array of **GeneralTextWordsBlockList** objects	识别文字块列表。输出顺序从左到右，先上后下。

版权所有 © 华为云计算技术有限公司

表 4-35 GeneralTextWordsBlockList

参数	参数类型	描述
words	String	文字块识别结果。
location	Array<Array<Integer>>	文字块的区域位置信息，列表形式，包含文字区域四个顶点的二维坐标（x,y）;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。 **说明** 输入数据格式是PDF时，返回的字段坐标仅用于参考，表示字段间的相对位置关系。
confidence	Float	文字块识别结果的置信度。
char_list	Array of **GeneralTextCharList** objects	文字块对应的单字符识别列表，输出顺序从左到右，先上后下。

表 4-36 GeneralTextCharList

参数	参数类型	描述
char	String	单字符识别结果。
char_location	Array<Array<Integer>>	单字符的区域位置信息，列表形式，包含字符区域四个顶点的二维坐标（x,y）;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。
char_confidence	Float	单字符识别结果的置信度。
markdown_result	String	所有文字块拼接的识别结果，同一行的文字块使用“\t”拼接，不同行的文字块使用“\n”拼接。 当return_markdown_result为true时，返回该字段值，否则，不返回该字段。

状态码： 400

表 4-37 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

版权所有 © 华为云计算技术有限公司

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。
  例如，通用文字识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/general-text”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入图片的base64编码进行文字识别，识别过程不校验图片倾斜角度，并关闭快速模式

```
POST https://{endpoint}/v2/{project_id}/ocr/general-text
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
    "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...",
    "detect_direction":false,
    "quick_mode":false
 }
```

- 传入图片的url进行文字识别，识别过程不校验图片倾斜角度，并关闭快速模式

```
POST https://{endpoint}/v2/{project_id}/ocr/general-text
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
    "url":"https://BucketName.obs.xxxx.com/ObjectName",
    "detect_direction":false,
    "quick_mode":false
 }
```

响应示例

状态码：200

成功响应示例

```
{
  "result" : {
    "direction" : 67.6506,
    "words_block_count" : 1,
    "words_block_list" : [ {
      "words" : "文字",
      "confidence" : 0.9999,
      "location" : [ [ 517, 447 ], [ 540, 504 ], [ 505, 518 ], [ 482, 461 ] ],
      "char_list" : [ {
        "char" : "文",
        "char_location" : [ [ 517, 447 ], [ 530, 479 ], [ 495, 493 ], [ 482, 461 ] ],
        "char_confidence" : 0.9999
      }, {
        "char" : "字",
        "char_location" : [ [ 530, 479 ], [ 540, 504 ], [ 505, 518 ], [ 495, 493 ] ],
        "char_confidence" : 0.9999
      } ]
    } ]
  }
}
```

状态码：400

版权所有 © 华为云计算技术有限公司

失败响应示例

```
{
    "error_code": "AIS.0103",
    "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入图片的base64编码进行文字识别，识别过程不校验图片倾斜角度，并关闭快速模式

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeGeneralTextSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeGeneralTextRequest request = new RecognizeGeneralTextRequest();
        GeneralTextRequestBody body = new GeneralTextRequestBody();
        body.withQuickMode(false);
        body.withDetectDirection(false);
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeGeneralTextResponse response = client.recognizeGeneralText(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
```


版权所有 © 华为云计算技术有限公司

```
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入图片的url进行文字识别，识别过程不校验图片倾斜角度，并关闭快速模式

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeGeneralTextSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeGeneralTextRequest request = new RecognizeGeneralTextRequest();
        GeneralTextRequestBody body = new GeneralTextRequestBody();
        body.withQuickMode(false);
        body.withDetectDirection(false);
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeGeneralTextResponse response = client.recognizeGeneralText(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入图片的base64编码进行文字识别，识别过程不校验图片倾斜角度，并关闭快速模式

```
# coding: utf-8
```

 版权所有 © 华为云计算技术有限公司

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeGeneralTextRequest()
        request.body = GeneralTextRequestBody(
            quick_mode=False,
            detect_direction=False,
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_general_text(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入图片的url进行文字识别，识别过程不校验图片倾斜角度，并关闭快速模式

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeGeneralTextRequest()
        request.body = GeneralTextRequestBody(
            quick_mode=False,
            detect_direction=False,
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_general_text(request)
```

 版权所有 © 华为云计算技术有限公司

```
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入图片的base64编码进行文字识别，识别过程不校验图片倾斜角度，并关闭快速模式

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeGeneralTextRequest{}
    quickModeGeneralTextRequestBody:= false
    detectDirectionGeneralTextRequestBody:= false
    imageGeneralTextRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.GeneralTextRequestBody{
        QuickMode: &quickModeGeneralTextRequestBody,
        DetectDirection: &detectDirectionGeneralTextRequestBody,
        Image: &imageGeneralTextRequestBody,
    }
    response, err := client.RecognizeGeneralText(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入图片的url进行文字识别，识别过程不校验图片倾斜角度，并关闭快速模式

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
```

 版权所有 © 华为云计算技术有限公司

```
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeGeneralTextRequest{}
    quickModeGeneralTextRequestBody:= false
    detectDirectionGeneralTextRequestBody:= false
    urlGeneralTextRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.GeneralTextRequestBody{
        QuickMode: &quickModeGeneralTextRequestBody,
        DetectDirection: &detectDirectionGeneralTextRequestBody,
        Url: &urlGeneralTextRequestBody,
    }
    response, err := client.RecognizeGeneralText(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

版权所有 © 华为云计算技术有限公司

4.4 网络图片识别

功能介绍

识别网络图片中的文字内容，并以JSON格式返回识别的结构化结果。支持横向、竖向、艺术字识别，支持字体分类和长图检测。

该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-3 网络图片示例图



约束与限制

- 支持中英文及部分中文繁体字。
- 只支持识别JPG、JPEG、PNG、BMP、TIFF、TGA、WEBP、ICO、PCX、GIF格式图片。
- 支持常见网络图片：手机截图、电脑截图、电商产品图及广告设计图等互联网图片。
- 图像各边的像素大小在15px到30000px之间。单个图片对应的Base64编码不超过10MB。
- 图像中有效文字图片占比超过60%，避免有效文字图片占比过小。
- 支持图像中有效文字图片的任意角度的水平旋转（需开启方向检测）。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/web-image

版权所有 © 华为云计算技术有限公司

表 4-38 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-39 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

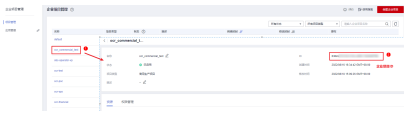
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** **该功能仅支持开通在华北-北京四、华东-上海一区域的API。** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-40 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最短边不小于15px，最长边不超过30000px，支持JPG、JPEG、PNG、BMP、TIFF、TGA、WEBP、ICO、PCX、GIF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url例如https://support.huaweicloud.com/api-ocr/zh-cn_image_0288038400.png • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
detect_direction	否	Boolean	是否校正图片的倾斜角度，可选值如下。 • true：校正图片的倾斜角度 • false：不校正图片的倾斜角度 支持任意角度的校正，未传入该参数时默认为“false”。 待识别图片如果存在倾斜，建议将此参数设置为“true”。
extract_type	否	Array of strings	结构化数据提取参数列表，目前只支持联系人信息、图像宽高，其入参值分别为“contact_info”，“image_size”。 若不填写该参数或删除该参数，默认不提取该参数值。
detect_font	否	Boolean	为Boolean类型，若不传该字段，默认不检测切片字体，为True时，将检测切片的字体类型，并返回最相似的5种字体名称。
detect_text_direction	否	Boolean	为Boolean类型，若不传该字段，默认为True，即检测每个字段的文字方向。为False时，则不检测文字方向。若图片中所有文字方向均是水平朝上时，建议将该值设为False，即不检测文字方向。

版权所有 © 华为云计算技术有限公司

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-41 响应 Body 参数

参数	参数类型	描述
result	**WebImageResult** object	调用成功时表示调用结果。 调用失败时无此字段。

表 4-42 WebImageResult

参数	参数类型	描述
words_block_count	Integer	代表检测识别出来的文字块数目。
words_block_list	Array of **WebImageWordsBlockList** objects	识别文字块列表，输出顺序从左到右，从上到下。 • 支持竖行识别。 • 支持多朝向文字识别。 • 支持字体识别。
extracted_data	**WebImageExtractedData** object	提取出的结构化JSON结果，该字典内的key值与入参列表extract_type的值一致，目前仅支持联系人、图像高宽信息提取，亦即key值为"contact_info"的字段、"image_size"字段。 若入参extract_type为空列表或该字段缺失时，不进行提取，此字段为空。

表 4-43 WebImageWordsBlockList

参数	参数类型	描述
words	String	文字块识别结果。
confidence	Float	相关字段的置信度信息，置信度越大，表示本次识别的对应字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于对应字段的准确率。
location	Array<Array<Integer>>	文字块的区域位置信息，列表形式，包含文字区域四个顶点的二维坐标（ x,y ）;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
font_list	Array of strings	文字块所属字体类型，列表形式，表示与文字块的文字最接近的字体类型。
font_scores	Array of numbers	文字块所属字体类型的概率，列表形式，与font_list一一对应，表示文字块的文字属于某种字体类型的概率。

表 4-44 WebImageExtractedData

参数	参数类型	描述
contact_info	**WebImageContactInfo** object	该字段表示提取的联系人信息，包括：姓名、联系电话、省市区以及详细地址。 若入参extract_type列表中无该字段，则此字段不存在。
image_size	**WebImageImageSize** object	该字段表示返回图片宽高信息。 如入参extract_type列表中无该字段，则此字段不存在。

表 4-45 WebImageContactInfo

参数	参数类型	描述
name	String	传入contact_info时的返回，为姓名。
phone	String	传入contact_info时的返回，联系电话。
province	String	传入contact_info时的返回，省。
city	String	传入contact_info时的返回，市。
district	String	传入contact_info时的返回，县区。
detail_address	String	传入contact_info时的返回，详细地址（不含省市区）。

表 4-46 WebImageImageSize

参数	参数类型	描述
height	Integer	传入image_size时的返回，为图像高度。
width	Integer	传入image_size时的返回，为图像宽度。

状态码： 400

版权所有 © 华为云计算技术有限公司

表 4-47 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。
  例如，网络图片识别服务部署在“华南-广州”区域的“endpoint”为“ocr.cn-south-1.myhuaweicloud.com”或“ocr.cn-south-1.myhuaweicloud.cn”，请求URL为“https://ocr.cn-south-1.myhuaweicloud.com/v2/{project_id}/ocr/web-image”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入网络图片的base64编码进行文字识别，识别过程校验图片倾斜角度，判断待识别字体类型，并校验图片是否包含联系人信息

```
POST https://{endpoint}/v2/{project_id}/ocr/web-image
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...

Request Body:
{
   "image":"/9j/4AAQSkZJRgABAgEASABIAAD/...",
   "extract_type" : [ "contact_info", "image_size" ],
   "detect_direction" : true,
   "detect_font" : true
}
```

- 传入网络图片的url进行文字识别，识别过程校验图片倾斜角度，判断待识别字体类型，并校验图片是否包含联系人信息

```
POST https://{endpoint}/v2/{project_id}/ocr/web-image
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "url":"https://BucketName.obs.xxxx.com/ObjectName",
    "extract_type" : [ "contact_info", "image_size" ],
    "detect_direction" : true,
    "detect_font" : true
}
```

响应示例

状态码：200

成功响应示例

```
{
  "result" : {
    "extracted_data" : {
```


版权所有 © 华为云计算技术有限公司

```
    "contact_info" : {
      "name" : "张三",
      "phone" : "13112345678",
      "province" : "广东省",
      "city" : "深圳市",
      "district" : "龙岗区",
      "detail_address" : "五和大道华为D区"
    },
    "image_size" : {
      "height" : 480,
      "width" : 640
    }
  },
  "words_block_count" : 2,
  "words_block_list" : [ {
    "words" : "收件人：张三，联系电话：13112345678，收件地址：广东省深圳市龙岗区五和大道华为D区",
    "confidence" : 0.9188,
    "location" : [ [ 13, 476 ], [ 91, 332 ], [ 125, 351 ], [ 48, 494 ] ],
    "font_list" : [ "黑体", "Arial", "华文中宋体", "微软正黑粗体", "文泉驿等宽正黑体" ],
    "font_scores" : [ 0.36, 0.2761, 0.2117, 0.0616, 0.0606 ]
  }, {
    "words" : "方正字体",
    "confidence" : 0.8631,
    "location" : [ [ 13, 476 ], [ 91, 332 ], [ 125, 351 ], [ 48, 494 ] ],
    "font_list" : [ "方正铭石体", "方正黑简体", "文泉驿等宽正黑体", "腾祥孔淼石头简体", "微软正黑粗体" ],
    "font_scores" : [ 0.9999, 0.0, 0.0, 0.0, 0.0 ]
  } ]
 }
}
```


状态码：400

失败响应示例


```
{
   "error_code": "AIS.0103",
   "error_msg": "The image size does not meet the requirements."
}
```


SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入网络图片的base64编码进行文字识别，识别过程校验图片倾斜角度，判断待识别字体类型，并校验图片是否包含联系人信息

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class RecognizeWebImageSolution {
```

```
    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeWebImageRequest request = new RecognizeWebImageRequest();
        WebImageRequestBody body = new WebImageRequestBody();
        List<String> listbodyExtractType = new ArrayList<>();
        listbodyExtractType.add("contact_info");
        listbodyExtractType.add("image_size");
        body.withDetectFont(true);
        body.withExtractType(listbodyExtractType);
        body.withDetectDirection(true);
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeWebImageResponse response = client.recognizeWebImage(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入网络图片的url进行文字识别，识别过程校验图片倾斜角度，判断待识别字体类型，并校验图片是否包含联系人信息

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class RecognizeWebImageSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```

 版权所有 © 华为云计算技术有限公司

```
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeWebImageRequest request = new RecognizeWebImageRequest();
        WebImageRequestBody body = new WebImageRequestBody();
        List<String> listbodyExtractType = new ArrayList<>();
        listbodyExtractType.add("contact_info");
        listbodyExtractType.add("image_size");
        body.withDetectFont(true);
        body.withExtractType(listbodyExtractType);
        body.withDetectDirection(true);
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeWebImageResponse response = client.recognizeWebImage(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入网络图片的base64编码进行文字识别，识别过程校验图片倾斜角度，判断待识别字体类型，并校验图片是否包含联系人信息

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeWebImageRequest()
        listExtractTypebody = [
```

 版权所有 © 华为云计算技术有限公司

```
            "contact_info",
            "image_size"
        ]
        request.body = WebImageRequestBody(
            detect_font=True,
            extract_type=listExtractTypebody,
            detect_direction=True,
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_web_image(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入网络图片的url进行文字识别，识别过程校验图片倾斜角度，判断待识别字体类型，并校验图片是否包含联系人信息

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeWebImageRequest()
        listExtractTypebody = [
            "contact_info",
            "image_size"
        ]
        request.body = WebImageRequestBody(
            detect_font=True,
            extract_type=listExtractTypebody,
            detect_direction=True,
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_web_image(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入网络图片的base64编码进行文字识别，识别过程校验图片倾斜角度，判断待识别字体类型，并校验图片是否包含联系人信息

```
package main
```

 版权所有 © 华为云计算技术有限公司

```
import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeWebImageRequest{}
    var listExtractTypebody = []string{
        "contact_info",
        "image_size",
    }
    detectFontWebImageRequestBody:= true
    detectDirectionWebImageRequestBody:= true
    imageWebImageRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.WebImageRequestBody{
        DetectFont: &detectFontWebImageRequestBody,
        ExtractType: &listExtractTypebody,
        DetectDirection: &detectDirectionWebImageRequestBody,
        Image: &imageWebImageRequestBody,
    }
    response, err := client.RecognizeWebImage(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入网络图片的url进行文字识别，识别过程校验图片倾斜角度，判断待识别字体类型，并校验图片是否包含联系人信息

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```

 版权所有 © 华为云计算技术有限公司

```
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeWebImageRequest{}
    var listExtractTypebody = []string{
        "contact_info",
        "image_size",
    }
    detectFontWebImageRequestBody:= true
    detectDirectionWebImageRequestBody:= true
    urlWebImageRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.WebImageRequestBody{
        DetectFont: &detectFontWebImageRequestBody,
        ExtractType: &listExtractTypebody,
        DetectDirection: &detectDirectionWebImageRequestBody,
        Url: &urlWebImageRequestBody,
    }
    response, err := client.RecognizeWebImage(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

版权所有 © 华为云计算技术有限公司

4.5 智能分类识别

功能介绍

自动分类识别17+种票证，支持指定票证的结构化识别和印章检测，并以JSON格式返回识别的结构化结果。

接口以列表形式返回图片上要识别票证的位置坐标、结构化识别的内容以及对应的类别。该接口支持的票证类型请见**表4-48**，该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-4 智能分类示例图



表 4-48 支持票证类型

票证类型	
增值税发票（支持识别电子版）	增值税发票（销货清单）
通用机打发票	定额发票
出租车票	火车票（支持识别电子版）
飞机行程单（支持识别电子版）	车辆通行费发票
机动车销售发票	车船票

版权所有 © 华为云计算技术有限公司

票证类型	
网约车行程单	身份证
护照	驾驶证
行驶证（支持识别电子版）	道路运输证
银行卡	-

说明

智能分类服务只对识别成功的票证进行计费，识别失败的票证不计费。例如，图片中包含三张票证，有两张识别成功，一张识别失败，此时接口计费两次。

约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF、PDF、OFD格式数据，OFD格式数据仅支持增值税发票服务调用。
- OFD格式数据仅支持识别第一页；PDF格式数据默认识别第一页，或者您可以指定要识别的页码。
- 图像各边的像素大小在15px到8000px之间。单个图片对应的Base64编码不超过10MB。
- 支持图像中票证任意角度的水平旋转。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/auto-classification

版权所有 © 华为云计算技术有限公司

表 4-49 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-50 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

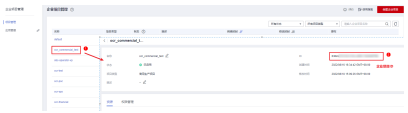
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-51 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，要求 单个图片、PDF文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图片最短边不小于15px，最长边不超过8000px，支持JPEG、JPG、PNG、BMP、TIFF、PDF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
url	否	String	该参数与image二选一。 单个图片、PDF文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图片的url路径目前支持： • 公网http/https url例如https://support.huaweicloud.com/api-ocr/zh-cn_image_0000001222608606.png • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
type_list	否	List	输入图片中，待识别票证的类型，指定后不出现在type_list参数中的票证不识别。不指定时默认返回所有支持类别票证的识别信息。 当前版本支持的票证类型请参见**表4-52**。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明

extended_parameters	否	Object	可指定需要识别票证的传入参数，具体参数可参考各票证API文档。若不指定则默认传入image 。当前版本支持票证类型如下：
			● vat_invoice：增值税发票（含全电发票）
			● quota_invoice：定额发票
			● taxi_invoice：出租车票
			● train_ticket：火车票
			● flight_itinerary：飞机行程单
			● toll_invoice：车辆通行费发票
			● mvs_invoice：机动车销售发票
			● id_card：身份证
			● passport：护照
			● driver_license：驾驶证
			● vehicle_license：行驶证
			● transportation_license：道路运输证
			● bankcard：银行卡
			● bus_ship_ticket：车船票
			● ride_hailing_itinerary：网约车行程单
			● machine_printed_invoice：通用机打发票
			增值税发票传参示例：
			"extended_parameters" : {
			"vat_invoice" : {
			"advanced_mode" : true,
			"return_text_location" : true
			}
			}
			车船票传参示例：
			"extended_parameters" : {
			"bus_ship_ticket":{
			"return_text_location" : true //开启返回识别到的文字块的区域位置信息。
			}
			}
			说明
			若指定票证参数填写错误会导致该票证识别错误，会提示"AIS.0101":"The input parameter is invalid."报错。

detect_seal	否	Boolean	检测印章开关，开启时则返回单张票证是否含有印章，否则不返回是否含有印章。可选值包括： ● true：开启检测票证是否含有印章功能。 ● false：关闭检测票证是否含有印章功能。 **说明** 该功能仅检测判断有无印章，不返回印章具体内容。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
pdf_page_number	否	Integer	指定PDF页码识别。传入该参数时，则识别指定页码的内容。如果不传该参数，则默认识别第1页。 **说明** 如果需要指定PDF页码识别，请通过该参数传入页码。目前不支持通过extended_parameters参数指定票证PDF页码。

表 4-52 支持票证类型

归属服务名称	参数值	类型描述
身份证识别	id_card_portrait_side	身份证头像面
	id_card_emblem_side	身份证国徽面
行驶证识别	vehicle_license_front	行驶证主页
	vehicle_license_back	行驶证副页
驾驶证识别	driver_license_front	驾驶证主页
	driver_license_back	驾驶证副页
护照识别	china_passport或 foreign_passport	护照
银行卡识别	bankcard	银行卡
道路运输证识别	transportation_license	道路运输证
增值税发票识别	vat_invoice	增值税专用发票 增值税普通发票 增值税电子普通发票（含区块链发票） 增值税电子专用发票 增值税电子普通发票（通行费） 全电发票（不含行业类发票）
	vat_roll_invoice	增值税普通发票（卷票）
	vat_invoice_sales_list	增值税发票（销货清单）
机动车销售统一发票识别	mvs_invoice	机动车销售统一发票
出租车发票识别	taxi_invoice	出租车发票

版权所有 © 华为云计算技术有限公司

归属服务名称	参数值	类型描述
火车票识别	train_ticket	火车票
定额发票识别	quota_invoice	定额发票
车辆通行费发票识别	toll_invoice	车辆通行费发票
飞机行程单识别	flight_itinerary	飞机行程单
智能分类识别	bus_ship_ticket	车船票
	ride_hailing_itinerary	网约车行程单
	machine_printed_invoice	通用机打发票

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-53 响应 Body 参数

参数	参数类型	描述
result	Array of **AutoClassificationResult** objects	识别结果。 调用失败时不返回此字段。

表 4-54 AutoClassificationResult

参数	参数类型	描述
status	**status** object	指示各对应票证的识别状态。
content	Object	对应票证具体结构化识别的结果。具体每个票证类别的响应参数请参考各API接口文档。 **说明** • **车船票识别响应参数** • **通用机打发票识别响应参数** • **网约车行程单识别响应参数**
type	String	对应票证的类别。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
location	Array<Array<Integer>>	文字块的区域位置信息，列表形式，包含文字区域四个顶点的二维坐标（x,y）;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。 **说明** 输入数据格式是PDF时，返回的字段坐标仅用于参考，表示字段间的相对位置关系。
seal_mark	Boolean	对应票证中是否含有印章。可选值包括： ● true：该票证中含有印章。 ● false：该票证中不含有印章。

表 4-55 BusShipTicketResult 车船票识别

参数	参数类型	描述
title	String	标题。
code	String	发票代码。
number	String	发票号码。
name	String	乘车人姓名。
id_number	String	乘车人身份证号。
departure_date	String	出发日期。
departure_time	String	出发时间。
departure_station	String	出发车站。
destination_station	String	到达车站。
total_amount	String	金额。
province	String	省份。
city	String	城市。
currency	String	币种（“默认RMB”）。
confidence	Map<String,Number>	相关字段的置信度信息，置信度越大，表示本次识别的对应字段的可靠性越高，在统计意义上，置信度越大，准确率越高。注：置信度由算法给出，不直接等价于对应字段的准确率。

版权所有 © 华为云计算技术有限公司

表 4-56 MachinePrintedInvoiceResult 通用机打发票识别

参数	参数类型	描述
title	String	发票名称。
code	String	发票代码。
number	String	发票号码。
date	String	开票日期。
time	String	时间。
check_code	String	校验码。
industry	String	行业分类。
category	String	种类。
province	String	省。
city	String	市。
buyer_name	String	购买方名称。
buyer_id	String	购买方纳税人识别号。
buyer_address_tel	String	购买方地址及电话。
buyer_bank_account	String	购买方银行及账号。
payment_info	String	经办人支付信息。
ticket_pickup_user	String	经办人取票用户。
merchant_number	String	经办人商户号。
order_number	String	经办人订单号。
item_list	Array of **ItemList** Objects	项目列表。
subtotal_tax	String	合计税额。
subtotal_amount	String	合计金额。
total_in_words	String	价税合计（大写）。
total	String	价税合计（小写）。
seller_name	String	销售方名称。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
seller_id	String	销售方纳税人识别号。
seller_address_tel	String	销售方地址及电话。
seller_bank_account	String	销售方银行及账号。
receiptor	String	收款人。
reviewer	String	复核人。
issuer	String	开票人。
company_seal_mark	Boolean	是否含有公司印章，取值为true或false： ● 含有公司印章：true ● 不含有公司印章：false
electronic_mark	Boolean	是否为浙江/广东通用机打电子发票，取值为true或false： ● 是浙江/广东通用机打电子发票：true ● 不是浙江/广东通用机打电子发票：false

表 4-57 MachinePrintedInvoiceItemList 通用机打发票识别

参数	参数类型	描述
name	String	项目名称。
specification	String	规格型号。
unit	String	单位。
quantity	String	数量。
unit_price	String	单价。
amount	String	金额。
tax_rate	String	税率。
tax	String	税额。

表 4-58 RideHailingItinerary 网约车行程单识别

参数	参数类型	描述
title	String	标题。
date	String	申请日期。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
date_start	String	行程开始时间。
date_end	String	行程结束时间。
phone_number	String	行程人手机号。
total	String	总计金额。
itineraries	String	行程数。
items_list	Array of **RideItineraryResult** object	行程信息列表。

表 4-59 RideItineraryResult 网约车行程单识别

参数	参数类型	描述
serial_number	String	序号。
vehicle_type	String	车型。
pickup_time	String	上车时间。
dropoff_time	String	下车时间。
city	String	城市。
departure	String	起点。
destination	String	终点。
mileage	String	里程。
amount	String	金额。
provider	String	服务商。
time_order	String	下单时间。
remarks	String	备注。

表 4-60 status

参数	参数类型	描述
error_code	String	指示各对应票证的状态码。
error_msg	String	指示各对应票证的状态信息。

版权所有 © 华为云计算技术有限公司

状态码： 400

表 4-61 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。
  例如，智能分类识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/auto-classification”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入图片的base64编码，进行图片分类和内容识别

```
POST https://{endpoint}/v1.0/ocr/auto-classification
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
   "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入图片的url，进行图片分类和内容识别

```
POST https://{endpoint}/v2/{project_id}/ocr/auto-classification
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
   "url":"https://BucketName.obs.xxxx.com/ObjectName"
}
```

响应示例

状态码：200

成功响应示例

```
{
  "result" : [ {
    "status" : {
      "error_code" : "AIS.0000",
      "error_msg" : "Succeeded."
    },
    "type" : "bus_ship_ticket",
    "location" : [ [ 10, 11 ], [ 607, 9 ], [ 608, 244 ], [ 9, 247 ] ],
```


版权所有 © 华为云计算技术有限公司

```
  "content" : {
    "title" : "台州市xxx经营有限公司汽车客运机1打310",
    "code" : "133101982501",
    "number" : "00462506",
    "departure_date" : "2020年12月02日",
    "departure_time" : "20:00",
    "total_amount" : "9.00",
    "departure_station" : "黄岩站",
    "destination_station" : "温岭火车",
    "name" : "鲁xx",
    "id_number" : "331021xxxxxx3018",
    "province" : "浙江省",
    "city" : "台州市",
    "currency" : "RMB",
    "confidence" : {
      "title" : 0.322,
      "code" : 0.9344,
      "number" : 0.9864,
      "departure_date" : 0.9992,
      "departure_time" : 0.964,
      "total_amount" : 0.9992,
      "departure_station" : 0.9984,
      "destination_station" : 0.9232,
      "name" : 0.9024,
      "id_number" : 0.9752,
      "province" : 0.9999,
      "city" : 0.9999,
      "currency" : 0.9999
    }
  },
  "seal_mark" : true
}, {
  "status" : {
    "error_code" : "AIS.0000",
    "error_msg" : "Succeeded."
  },
  "type" : "ride_hailing_itinerary",
  "location" : [ [ 70, 330 ], [ 560, 330 ], [ 559, 558 ], [ 70, 557 ] ],
  "content" : {
    "title" : "xx出行-行程单",
    "date" : "2020-06-28",
    "date_start" : "2020-06-19",
    "date_end" : "2020-06-24",
    "phone_number" : "17xxxx99344",
    "total" : "351.40",
    "itineraries" : "2",
    "items_list" : [ {
      "serial_number" : "1",
      "vehicle_type" : "快车",
      "pickup_time" : "06-19 21:37 周五",
      "dropoff_time" : "",
      "city" : "北京市",
      "departure" : "xxxx有限公司-西门",
      "destination" : "北家路一区",
      "mileage" : "45",
      "amount" : "166.90",
      "provider" : "",
      "time_order" : "",
      "remarks" : ""
    }, {
      "serial_number" : "2",
      "vehicle_type" : "快车",
      "pickup_time" : "06-24 21:46 周三",
      "dropoff_time" : "",
      "city" : "北京市",
      "departure" : "xxxx有限公司-西门",
      "destination" : "北家路一区",
      "mileage" : "34.48",
      "amount" : "144.10",
```


 版权所有 © 华为云计算技术有限公司

```
      "provider" : "",
      "time_order" : "",
      "remarks" : ""
    } ]
  },
  "seal_mark" : true
 }, {
  "status" : {
    "error_code" : "AIS.0104",
    "error_msg" : "The image is not supported or the image quality is poor."
  },
  "type" : "quota_invoice",
  "location" : [ [ 816, 103 ], [ 2125, 103 ], [ 2125, 978 ], [ 816, 978 ] ],
  "content" : { },
  "seal_mark" : false
 } ]
}
```

状态码：400

失败响应示例

```
{
   "error_code": "AIS.0103",
   "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入图片的base64编码，进行图片分类和内容识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class RecognizeAutoClassificationSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);
```

 版权所有 © 华为云计算技术有限公司

```
        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeAutoClassificationRequest request = new RecognizeAutoClassificationRequest();
        AutoClassificationRequestBody body = new AutoClassificationRequestBody();
        List<String> listbodyTypeList = new ArrayList<>();
        listbodyTypeList.add("vat_invoice");
        listbodyTypeList.add("train_ticket");
        listbodyTypeList.add("taxi_invoice");
        body.withExtendedParameters("{\"toll_invoice\":{},\"mvs_invoice\":{},\"taxi_invoice\":
{\"return_text_location\":true},\"id_card\":{\"return_verification\":true,\"detect_copy
\":true,\"return_text_location\":true,\"detect_reproduce\":true},\"vehicle_license\":
{\"return_issuing_authority\":true,\"return_text_location\":true},\"quota_invoice\":{},\"vat_invoice\":
{\"advanced_mode\":true,\"return_text_location\":true},\"flight_itinerary\":{},\"transportation_license\":
{},\"passport\":{},\"train_ticket\":{\"return_text_location\":true},\"bankcard\":{\"return_text_location
\":true},\"driver_license\":{\"return_issuing_authority\":true,\"return_text_location\":true}}");
        body.withTypeList(listbodyTypeList);
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeAutoClassificationResponse response = client.recognizeAutoClassification(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入图片的url，进行图片分类和内容识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeAutoClassificationSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
```

 版权所有 © 华为云计算技术有限公司

```
        RecognizeAutoClassificationRequest request = new RecognizeAutoClassificationRequest();
        AutoClassificationRequestBody body = new AutoClassificationRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeAutoClassificationResponse response = client.recognizeAutoClassification(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入图片的base64编码，进行图片分类和内容识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeAutoClassificationRequest()
        listTypeListbody = [
            "vat_invoice",
            "train_ticket",
            "taxi_invoice"
        ]
        request.body = AutoClassificationRequestBody(
            extended_parameters="{\"toll_invoice\":{},\"mvs_invoice\":{},\"taxi_invoice\":{\"return_text_location\":true},\"id_card\":{\"return_verification\":true,\"detect_copy\":true,\"return_text_location\":true,\"detect_reproduce\":true},\"vehicle_license\":{\"return_issuing_authority\":true,\"return_text_location\":true},\"quota_invoice\":{},\"vat_invoice\":{\"advanced_mode\":true,\"return_text_location\":true},\"flight_itinerary\":{},\"transportation_license\":{},\"passport\":{},\"train_ticket\":{\"return_text_location\":true},\"bankcard\":{\"return_text_location\":true},\"driver_license\":{\"return_issuing_authority\":true,\"return_text_location\":true}}",
            type_list=listTypeListbody,
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_auto_classification(request)
        print(response)
    except exceptions.ClientRequestException as e:
```

 版权所有 © 华为云计算技术有限公司

```
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入图片的url，进行图片分类和内容识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeAutoClassificationRequest()
        request.body = AutoClassificationRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_auto_classification(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入图片的base64编码，进行图片分类和内容识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
```

 版权所有 © 华为云计算技术有限公司

```
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeAutoClassificationRequest{}
    var listTypeListbody = []string{
        "vat_invoice",
        "train_ticket",
        "taxi_invoice",
    }
    var extendedParametersAutoClassificationRequestBody interface{} = "{\"toll_invoice\":
{},\"mvs_invoice\":{},\"taxi_invoice\":{\"return_text_location\":true},\"id_card\":{\"return_verification
\":true,\"detect_copy\":true,\"return_text_location\":true,\"detect_reproduce\":true},\"vehicle_license\":
{\"return_issuing_authority\":true,\"return_text_location\":true},\"quota_invoice\":{},\"vat_invoice\":
{\"advanced_mode\":true,\"return_text_location\":true},\"flight_itinerary\":{},\"transportation_license\":
{},\"passport\":{},\"train_ticket\":{\"return_text_location\":true},\"bankcard\":{\"return_text_location
\":true},\"driver_license\":{\"return_issuing_authority\":true,\"return_text_location\":true}}"
    imageAutoClassificationRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/
4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.AutoClassificationRequestBody{
        ExtendedParameters: &extendedParametersAutoClassificationRequestBody,
        TypeList: &listTypeListbody,
        Image: &imageAutoClassificationRequestBody,
    }
    response, err := client.RecognizeAutoClassification(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入图片的url，进行图片分类和内容识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeAutoClassificationRequest{}
    urlAutoClassificationRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
```

 版权所有 © 华为云计算技术有限公司

```
    request.Body = &model.AutoClassificationRequestBody{
        Url: &urlAutoClassificationRequestBody,
    }
    response, err := client.RecognizeAutoClassification(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

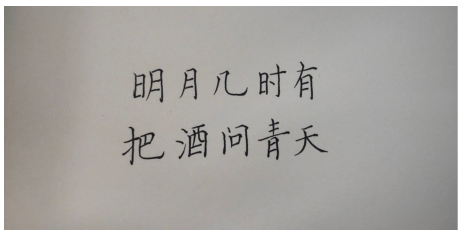
错误码请参见**错误码**。

4.6 手写文字识别

功能介绍

识别文档中的手写文字、印刷文字信息，并将识别的结构化结果以JSON格式返回给用户。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-5 手写文字示例图



约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。

版权所有 © 华为云计算技术有限公司

- 图像中识别区域有效占比超过80%，保证所有文字及其边缘包含在图像内。
- 支持图像任意角度的水平旋转（需开启方向检测）。
- 目前不支持复杂背景（如户外自然场景、防伪水印等）和表格线扭曲图像的文字识别。
- 文字书写越工整，识别率越高。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/handwriting

表 4-62 路径参数

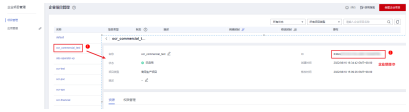
参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-63 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-64 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url例如https://support.huaweicloud.com/api-ocr/zh-cn_image_0288038984.png • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
quick_mode	否	Boolean	快速模式开关，针对单行文字图片（要求图片只包含一行文字，且文字区域占比超过50%），打开时可以更快返回识别内容。可选值如下所示。 • true：打开快速模式 • false：关闭快速模式 未传入该参数时默认为false，即关闭快速模式。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
char_set	否	String	字符集设置，用户可以根据实际需要限定输出字符集范围。可选值如下所示。 • digit：数字模式 • letter：大小写字母模式 • digit_letter：数字+字母模式 • general：数字+字母+中文模式 未传入该参数时，默认为“general”模式。
detect_direction	否	Boolean	是否校正图片的倾斜角度，可选值如下。 • true：校正图片的倾斜角度 • false：不校正图片的倾斜角度 支持任意角度的校正，未传入该参数时默认为“false”。 待识别图片如果存在倾斜，建议将此参数设置为“true”。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（status code）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-65 响应 Body 参数

参数	参数类型	描述
result	HandwritingResult object	识别结果。 调用失败时不返回此字段。

表 4-66 HandwritingResult

参数	参数类型	描述
words_block_count	Integer	检测到的文字块数目。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
words_block_list	Array of **HandwritingItemsResponse** objects	识别文字块列表。输出顺序从左到右，从上到下。

表 4-67 HandwritingItemsResponse

参数	参数类型	描述
words	String	文字块识别结果。
type	String	说明该识别结果所属类型，返回值为“text”。
confidence	Float	字段的置信度，取值范围0~1。 置信度越大，本次识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于字段的准确率。
location	Array<Array<Integer>>	文字块“words”的区域位置信息，列表形式，分别表示文字块顶点的（x,y）坐标；采用图像坐标系，坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。

状态码： 400

表 4-68 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

版权所有 © 华为云计算技术有限公司

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，手写文字识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/handwriting”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入手写文字图片的base64编码进行内容识别，识别范围包括数字、字母、中文，识别过程关闭快速模式并不校验图片倾斜角度

```
POST https://{endpoint}/v2/{project_id}/ocr/handwriting
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
   "image": "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...",
   "quick_mode": false,
   "char_set": "general",
   "detect_direction": false
 }
```

- 传入手写文字图片的url进行内容识别，识别范围包括数字、字母、中文，识别过程关闭快速模式并不校验图片倾斜角度

```
POST https://{endpoint}/v2/{project_id}/ocr/handwriting
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
    "url":"https://BucketName.obs.xxxx.com/ObjectName",
    "quick_mode":false,
    "char_set": "general",
    "detect_direction": false
 }
```

响应示例

状态码：200

成功响应示例

```
{
   "result": {
      "words_block_count": 2,
      "words_block_list": [
         {
            "words": "大江东去",
            "type": "text",
            "confidence": 0.98,
            "location": [
               [282, 45],
               [461, 47],
               [460, 77],
               [280, 76]
            ]
         },
         {
            "words": "浪淘尽",
            "type": "text",
```


版权所有 © 华为云计算技术有限公司

```
            "confidence": 0.99,
            "location": [
                [949, 52],
                [1095, 53],
                [1100, 87],
                [953, 86]
            ]
        }
    ]

}
}
```

状态码：400

失败响应示例

```
{
   "error_code": "AIS.0103",
   "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入手写文字图片的base64编码进行内容识别，识别范围包括数字、字母、中文，识别过程关闭快速模式并不校验图片倾斜角度。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeHandwritingSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeHandwritingRequest request = new RecognizeHandwritingRequest();
```

 版权所有 © 华为云计算技术有限公司

```
        HandwritingRequestBody body = new HandwritingRequestBody();
        body.withDetectDirection(false);
        body.withCharSet("general");
        body.withQuickMode(false);
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeHandwritingResponse response = client.recognizeHandwriting(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入手写文字图片的url进行内容识别，识别范围包括数字、字母、中文，识别过程关闭快速模式并不校验图片倾斜角度

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeHandwritingSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeHandwritingRequest request = new RecognizeHandwritingRequest();
        HandwritingRequestBody body = new HandwritingRequestBody();
        body.withDetectDirection(false);
        body.withCharSet("general");
        body.withQuickMode(false);
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeHandwritingResponse response = client.recognizeHandwriting(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入手写文字图片的base64编码进行内容识别，识别范围包括数字、字母、中文，识别过程关闭快速模式并不校验图片倾斜角度。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeHandwritingRequest()
        request.body = HandwritingRequestBody(
            detect_direction=False,
            char_set="general",
            quick_mode=False,
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_handwriting(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入手写文字图片的url进行内容识别，识别范围包括数字、字母、中文，识别过程关闭快速模式并不校验图片倾斜角度

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
```

 版权所有 © 华为云计算技术有限公司

```
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeHandwritingRequest()
        request.body = HandwritingRequestBody(
            detect_direction=False,
            char_set="general",
            quick_mode=False,
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_handwriting(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入手写文字图片的base64编码进行内容识别，识别范围包括数字、字母、中文，识别过程关闭快速模式并不校验图片倾斜角度。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeHandwritingRequest{}
    detectDirectionHandwritingRequestBody:= false
    charSetHandwritingRequestBody:= "general"
    quickModeHandwritingRequestBody:= false
    imageHandwritingRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/
4RFZRXhpZgAATU0AKgAAAA..."
```

 版权所有 © 华为云计算技术有限公司

```
    request.Body = &model.HandwritingRequestBody{
        DetectDirection: &detectDirectionHandwritingRequestBody,
        CharSet: &charSetHandwritingRequestBody,
        QuickMode: &quickModeHandwritingRequestBody,
        Image: &imageHandwritingRequestBody,
    }
    response, err := client.RecognizeHandwriting(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入手写文字图片的url进行内容识别，识别范围包括数字、字母、中文，识别过程关闭快速模式并不校验图片倾斜角度

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeHandwritingRequest{}
    detectDirectionHandwritingRequestBody:= false
    charSetHandwritingRequestBody:= "general"
    quickModeHandwritingRequestBody:= false
    urlHandwritingRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.HandwritingRequestBody{
        DetectDirection: &detectDirectionHandwritingRequestBody,
        CharSet: &charSetHandwritingRequestBody,
        QuickMode: &quickModeHandwritingRequestBody,
        Url: &urlHandwritingRequestBody,
    }
    response, err := client.RecognizeHandwriting(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.7 印章识别

功能介绍

检测和识别合同文件或常用票据中的印章，并可擦除和提取图片中的印章，通过JSON格式返回印章检测、识别、擦除和提取的结果。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

约束与限制

- 支持JPEG、JPG、PNG、BMP、TIFF、PDF格式，多页PDF默认识别第一页，或者您可以指定要识别的页码。
- 图像各边的像素大小在15px到8192px之间。单个图片、PDF文件其对应的Base64编码不超过10MB。
- 支持单图多印章识别。
- 支持图像中印章任意角度的水平旋转。
- 支持圆形章、椭圆章、方形章、三角章和菱形章的检测和识别。
- 能处理反光、暗光等干扰的图片但影响识别精度。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

版权所有 © 华为云计算技术有限公司

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/seal

表 4-69 路径参数

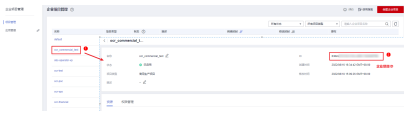
参数	是否必选	描述
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-70 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-71 请求 Body 参数

参数	是否必选	参数类型	描述
data	否	String	该参数与url二选一。 图片的Base64编码，要求 单个图片、PDF文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图片最短边不小于15px，最长边不超过8192px。支持JPEG、JPG、PNG、BMP、GIF、TIFF、WEBP、PCX、ICO、PSD、PDF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	该参数与data二选一。 单个图片、PDF文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图片的url路径目前支持： • 公网http/https url例如https://support.huaweicloud.com/api-ocr/zh-cn_image_0000001079447456.png • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
return_seal_image	否	Boolean	是否返回提取的印章base64编码图片，可选值如下所示： • true：返回提取的印章图片base64编码 • false：不返回提取的印章图片base64编码 如果无该参数，系统默认不返回印章擦除后的图片。如果输入参数不是Boolean类型，则会报非法参数错误。
return_erased_seal_image	否	Boolean	是否返回印章擦除后的base64编码图片，可选值包括： • true：返回印章擦除后的图片base64编码 • false：不返回印章擦除后的图片base64编码 如果无该参数，系统默认不返回印章擦除后的图片。如果输入参数不是Boolean类型，则会报非法参数错误。
pdf_page_number	否	Integer	指定PDF页码识别。传入该参数时，则识别指定页码的内容。如果不传该参数，则默认识别第1页。

响应参数

状态码： 200

表 4-72 响应 Body 参数

参数	参数类型	描述
result	**SealResult** object	调用成功时表示调用结果。 调用失败时无此字段。

表 4-73 SealResult

参数	参数类型	描述
seal_list	Array of **SealList** objects	印章信息列表。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
erased_seal_image	String	在输入图片基础上进行印章擦除后的base64编码图片。

表 4-74 SealList

参数	参数类型	描述
type	String	印章类型，当前支持circle（圆形章）、ellipse（椭圆章）、rectangle（方形章）、triangle（三角章）、rhombus（菱形章）五种。
seal_image	String	提取的单个印章base64编码图片。
confidence	Float	印章位置的置信度。
location	Array<Array<Integer>>	印章位置，列表形式，包含印章区域四个顶点的二维坐标（x,y），坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。
words_block_list	Array of **SealWordsBlockList** objects	印章文本块列表。

表 4-75 SealWordsBlockList

参数	参数类型	描述
words	String	印章文本块。
words_confidence	Float	印章文本块的置信度。

状态码： 400

表 4-76 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

版权所有 © 华为云计算技术有限公司

请求示例

- 传入印章图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/seal

{
  "data" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入印章证图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/seal

{
  "url" : "https://BucketName.obs.myhuaweicloud.com/ObjectName"
}
```

响应示例

状态码： 200

成功响应示例

```
{
  "result" : {
    "seal_list" : [ {
      "words_block_list" : [ {
        "words" : "1100000256848",
        "words_confidence" : 0.9269
      }, {
        "words" : "中国建材集团有限公司",
        "words_confidence" : 0.9999
      } ],
      "confidence" : 0.9834,
      "location" : [ [ 262, 321 ], [ 447, 321 ], [ 447, 507 ], [ 262, 507 ] ],
      "type" : "circle",
      "seal_image" : "/9j/4AAQSkZJRgABAQAAAQABAAD..."
    } ],
    "erased_seal_image" : "/9j/4AAQSkZJRgABAQAAAQABAA36K..."
  }
}
```

状态码： 400

失败响应示例

```
{
  "result" : {
    "error_code" : "AIS.0103",
    "error_msg" : "The image size does not meet the requirements."
  }
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入印章图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;
```

 版权所有 © 华为云计算技术有限公司

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeSealSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeSealRequest request = new RecognizeSealRequest();
        SealRequestBody body = new SealRequestBody();
        body.withData("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeSealResponse response = client.recognizeSeal(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入印章证图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeSealSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
```

```
        // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeSealRequest request = new RecognizeSealRequest();
        SealRequestBody body = new SealRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeSealResponse response = client.recognizeSeal(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入印章图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeSealRequest()
        request.body = SealRequestBody(
            data="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_seal(request)
        print(response)
```

 版权所有 © 华为云计算技术有限公司

```
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入印章证图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeSealRequest()
        request.body = SealRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_seal(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入印章图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
```

 版权所有 © 华为云计算技术有限公司

```
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeSealRequest{}
    dataSealRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.SealRequestBody{
        Data: &dataSealRequestBody,
    }
    response, err := client.RecognizeSeal(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入印章证图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeSealRequest{}
    urlSealRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.SealRequestBody{
        Url: &urlSealRequestBody,
    }
    response, err := client.RecognizeSeal(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.8 身份证识别

功能介绍

识别身份证图片中的文字内容，并将识别的结果以JSON格式返回给用户。支持身份证翻拍、PS、模糊、反光、边框完整性、边框内异物遮挡、复印件和临时身份证告警功能。

该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

身份证信息核验请使用**人证核身服务**。

图 4-6 身份证示例图



说明

身份证识别支持中华人民共和国居民身份证识别。

如果图片中包含多张卡证票据，请调用**智能分类识别**服务。

约束与限制

- 支持中华人民共和国居民身份证的识别。
- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。

版权所有 © 华为云计算技术有限公司

- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中身份证区域有效占比超过25%，保证整张身份证内容及其边缘包含在图像内。
- 支持图像中身份证任意角度的水平旋转。
- 支持少量扭曲，扭曲后图像中的身份证长宽比与实际身份证相差不超过10%。
- 能处理反光、暗光等干扰的图片但影响识别精度。
- 目前支持识别单张身份证的正面或者反面。
- 支持居民身份证的正反面同时识别，不支持存在两张及以上同面身份证的图片识别。
- 告警检测结果仅作为参考依据，实际图片情况可能与检测结果存在差异，可结合其他校验方式，如字段是否完整等共同判断。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/id-card

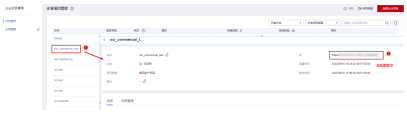
表 4-77 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

版权所有 © 华为云计算技术有限公司

请求参数

表 4-78 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-79 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片最短边不小于15px，最长边不超过8192px。支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url例如https://support.huaweicloud.com/api-ocr/zh-cn_image_0000001079447456.png • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
side	否	String	检测身份证的人像面或国徽面。 • front：身份证人像面 • back：身份证国徽面 • double_side：身份证双面信息 如果参数值为空或无该参数，系统自动识别，建议填写，准确率更高。
return_verification	否	Boolean	返回校验身份证号等信息的开关，默认false，可选值如下所示： • true：返回校验信息 • false：不返回校验信息

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
return_text_location	否	Boolean	识别到的文字块的区域位置信息。可选值包括： ● true：返回各个文字块区域 ● false：不返回各个文字块区域 如果无该参数，系统默认不返回文字块区域。如果输入参数不是Boolean类型，则会报非法参数错误。
return_portrait_image	否	Boolean	返回头像图片信息（base64码）的开关，默认false，可选值如下所示： ● true： 开启头像图片信息（base64码）的功能 ● false : 关闭头像图片信息（base64码）的功能
return_adjusted_image	否	Boolean	返回身份证卡面（base64码）的开关，默认false，可选值如下所示： ● true： 开启身份证卡面（base64码）的功能 ● false : 关闭身份证卡面（base64码）的功能
return_portrait_location	否	Boolean	返回头像位置信息的开关，默认false，可选值如下所示： ● true： 开启返回头像位置信息的功能 ● false : 关闭返回头像位置信息的功能
detect_reproduce	否	Boolean	返回判断身份证图像是否经过翻拍的开关，默认false，可选值如下所示： ● true： 开启判断身份证图像是否经过翻拍功能 ● false：关闭判断身份证图像是否经过翻拍功能
detect_copy	否	Boolean	返回判断身份证图像是否是黑白复印件的开关，默认false，可选值如下所示： ● true： 开启判断身份证图像是否是复印件功能 ● false : 关闭身份证图像是否是复印件功能

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
detect_tampering	否	Boolean	身份证图像PS告警功能开关，默认false，可选值如下： ● true： 开启身份证图像PS告警功能 ● false : 关闭身份证图像告警功能 **说明** ● 身份证PS告警功能不支持精细化P图。
detect_border_integrity	否	Boolean	身份证图像边框完整性告警功能开关，默认false，可选值如下： ● true： 打开身份证图像边框完整性告警功能 ● false : 关闭身份证图像边框完整性告警功能
detect_blocking_within_border	否	Boolean	身份证图像边框内部是否有异物遮挡的告警功能开关，默认false，可选值如下： ● true： 开启身份证边框内部异物遮挡告警功能 ● false : 关闭身份证边框内部异物遮挡告警功能
detect_blur	否	Boolean	身份证图像模糊告警功能的开关，默认false，可选值如下： ● true： 开启身份证图像模糊告警功能 ● false : 关闭身份证图像模糊告警功能
detect_interim	否	Boolean	临时身份证告警功能开关，默认false，可选值如下： ● true： 开启临时身份证告警功能 ● false : 关闭临时身份证告警功能
detect_glare	否	Boolean	身份证反光告警功能开关，默认false，可选值如下： ● true： 开启身份证反光告警功能 ● false : 关闭身份证反光告警功能

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

版权所有 © 华为云计算技术有限公司

表 4-80 响应 Body 参数

参数	参数类型	描述
result	**IDCardResult** object	识别结果。 调用失败时不返回此字段。

表 4-81 IdCardResult

参数	参数类型	描述
name	String	姓名。
sex	String	性别。
birth	String	出生日期。
ethnicity	String	民族。
address	String	地址。
number	String	身份证号。
issue	String	发证机关。
valid_from	String	有效起始日期。
valid_to	String	有效结束日期。
verification_result	**IdcardVerificationResult** object	校验信息。仅在输入参数return_verification为true时，返回该字段，该字段包含valid_number、valid_birth、valid_sex、valid_validity_period和valid_date的校验信息。
text_location	Object	对应所有在原图上识别到的字段位置信息，包含所有文字区域四个顶点的二维坐标（x,y）。采用图像坐标系，坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。
portrait_image	String	身份证头像图片信息的base64码结果。 **说明** • 仅在输入参数return_portrait_image为true时，返回该字段。 • 当输入为身份证背面时返回为空字符串。
adjusted_image	String	身份证卡面图片信息的base64码结果。 **说明** • 仅在输入参数return_adjusted_image为true时，返回该字段。
portrait_location	Array<Array<Integer>>	身份证头像位置信息的结果，仅在输入参数return_portrait_location为true时，返回该字段，当输入身份证背面时返回为空列表。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
detect_reproduce_result	Boolean	身份证图像是否翻拍告警结果。 ● true：表示身份证图片经过翻拍。 ● false：表示身份证图片未经过翻拍。 仅在输入参数detect_reproduce为true时，返回该字段。
detect_copy_result	Boolean	身份证图像是否黑白复印件告警结果。 ● true：表示身份证图片是复印件。 ● false：表示身份证图片是原件。 仅在输入参数detect_copy为true时，返回该字段。
detect_tampering_result	Boolean	身份证图片是否PS告警结果。 ● true：表示身份证经过PS。 ● false：表示未经过PS。 仅在传入参数detect_tampering为true时，返回该字段。
detect_border_integrity_result	Boolean	身份证图片边框完整性告警结果。 ● true：表示边框不完整 ● false：表示边框完整。 仅在输入参数detect_border_integrity为true时，返回该字段。
detect_blocking_within_border_result	Boolean	身份证图像框内是否存在遮挡的告警结果。 ● true：表示边框内部存在遮挡。 ● false：表示边框内部不存在遮挡。 仅在输入参数detect_blocking_within_border为true时，返回该字段。
detect_blur_result	Boolean	身份证模糊告警结果。 ● true：表示身份证图片较模糊。 ● false：表示身份证清晰。 仅在输入参数detect_blur为true时，返回该字段。
detect_interim_result	Boolean	临时身份证告警结果。 ● true：表示是临时身份证。 ● false：表示非临时身份证。 仅在输入参数detect_interim为true时，返回该字段。
detect_glare_result	Boolean	身份证反光告警结果。 ● true：表示身份证图片存在反光。 ● false：表示是身份证不存在反光。 仅在输入参数detect_glare为true时，返回该字段。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
score_info	**IdcardScoreInfoResult** object	告警分数，包含字段：tampering_score、interim_score、reproduce_score、copy_score、border_integrity_score、blur_score、glare_score、blocking_within_border_score，以上字段取值范围为[0, 99]。
front	**IdcardFrontResult** object	身份证人像面识别结果。 仅在输入参数side为double_side时，返回该字典，若未识别到人像面或人像面图片质量太差则该字典为空。
back	**IdcardBackResult** object	身份证国徽面识别结果。 仅在输入参数side为double_side时，返回该字典，若未识别到国徽面或国徽面图片质量太差则该字典为空。

表 4-82 IdcardVerificationResult

参数	参数类型	描述
valid_number	Boolean	身份证号规则校验是否通过。 ● true：表示身份证号规则校验通过。 ● false：表示身份证号规则校验不通过。 当识别结果为单页，身份证图片是国徽面时，默认是false。输入参数side为double_side时，该字典仅在front字段中存在。 仅在输入参数return_verification为true时，返回该字段。
valid_birth	Boolean	出生日期与身份证号所表示的出生日期是否一致。 ● true：表示一致。 ● false：表示不一致。 当识别结果为单页，身份证图片是国徽面，或者身份证号规则校验不通过时，默认是false。输入参数side为double_side时，该字段仅在front字典中存在。 仅在输入参数return_verification为true时，返回该字段。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
valid_sex	Boolean	性别与身份证号所表示的性别信息是否一致。 ● true：表示一致。 ● false：表示不一致。 当识别结果为单页，身份证图片是国徽面，或者身份证号规则校验不通过时，默认是false。输入参数side为double_side时，该字段仅在front字典中存在。 仅在输入参数return_verification为true时，返回该字段。
valid_date	Boolean	当前日期是否在有效期内。 ● true：表示当前日期在有效期内。 ● false：表示当前日期不在有效期内。 当识别结果为单页，身份证图片是人像面时，默认是false。输入参数side为double_side时，该字段仅在back字典中存在。 仅在输入参数return_verification为true时，返回该字段。
valid_validity_period	Boolean	身份证有效日期是否合法。 ● true：表示身份证的有效日期合法 ● false：表示身份证有效日期非法 当识别结果为单页，身份证图片是人像面时，默认是false。输入参数side为double_side时，该字段仅在back字典中存在。 仅在输入参数return_verification为true时，返回该字段。

表 4-83 IdcardFrontResult

参数	参数类型	描述
name	String	姓名。
sex	String	性别。
birth	String	出生日期。
ethnicity	String	民族。
address	String	地址。
number	String	身份证号。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
portrait_image	String	身份证头像图片信息的base64码结果。 **说明** • 仅在输入参数return_portrait_image为true时，返回该字段。 • 当输入为身份证背面时返回为空字符串。
adjusted_image	String	身份证卡面图片信息的base64码结果。 **说明** • 仅在输入参数return_adjusted_image为true时，返回该字段。
verification_result	**IdcardFrontVerificationResult** object	校验信息。 仅在输入参数return_verification为true时，返回该字段，该字段包含valid_number、valid_birth和valid_sex的校验信息。
text_location	Object	文本框在原图位置。输出左上、右上、右下、左下四个点坐标。 仅return_text_location设置为true时才返回。
portrait_location	Array<Array<Integer>>	身份证头像位置信息的结果。 仅在输入参数return_portrait_location为true时，返回该字段。
detect_reproduce_result	Boolean	身份证图像是否翻拍告警结果。 • true：表示身份证图片经过翻拍。 • false：表示身份证图片未经过翻拍。 仅在输入参数detect_reproduce为true时，返回该字段。
detect_copy_result	Boolean	身份证图像是否黑白复印件告警结果。 • true：表示身份证图片是复印件。 • false”表示身份证图片是原件。 仅在输入参数detect_copy为true时，返回该字段。
detect_tampering_result	Boolean	身份证图片是否PS告警结果。 • true：表示身份证经过PS。 • false：表示未经过PS。 仅在传入参数detect_tampering为true时，返回该字段。
detect_border_integrity_result	Boolean	身份证图片边框完整性告警结果。 • true：表示边框不完整 • false：表示边框完整。 仅在输入参数detect_border_integrity为true时，返回该字段。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
detect_blocking_within_border_result	Boolean	身份证图像框内是否存在遮挡的告警结果。 ● true：表示边框内部存在遮挡。 ● false：表示边框内部不存在遮挡。 仅在输入参数detect_blocking_within_border为true时，返回该字段。
detect_blur_result	Boolean	身份证模糊告警结果。 ● true：表示身份证图片较模糊。 ● false：表示身份证清晰。 仅在输入参数detect_blur为true时，返回该字段。
detect_interim_result	Boolean	临时身份证告警结果。 ● true：表示是临时身份证。 ● false：表示非临时身份证。 仅在输入参数detect_interim为true时，返回该字段。
detect_glare_result	Boolean	身份证反光告警结果。 ● true：表示身份证图片存在反光。 ● false：表示是身份证不存在反光。 仅在输入参数detect_glare为true时，返回该字段。
score_info	**IdcardScoreInfoResult** object	告警分数，包含字段：tampering_score、interim_score、reproduce_score、copy_score、border_integrity_score、blur_score、glare_score、blocking_within_border_score，以上字段取值范围为[0, 99]。

表 4-84 IdcardFrontVerificationResult

参数	参数类型	描述
valid_number	Boolean	身份证号规则校验是否通过。 ● true：表示身份证号规则校验通过。 ● false：表示身份证号规则校验不通过。 当识别结果为单页，身份证图片是国徽面时，默认是false。输入参数side为double_side时，该字典仅在front字段中存在。 仅在输入参数return_verification为true时，返回该字段。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
valid_birth	Boolean	出生日期与身份证号所表示的出生日期是否一致。 ● true：出生日期与身份证号所表示的出生日期一致。 ● false：出生日期与身份证号所表示的出生日期不一致。 当识别结果为单页，身份证图片是国徽面，或者身份证号规则校验不通过时，默认是false。输入参数side为double_side时，该字段仅在front字典中存在。 仅在输入参数return_verification为true时，返回该字段。
valid_sex	Boolean	性别与身份证号所表示的性别信息是否一致。 - true：性别与身份证号所表示的性别信息一致 - false：性别与身份证号所表示的性别信息不一致。 当识别结果为单页，身份证图片是国徽面，或者身份证号规则校验不通过时，默认是false。输入参数side为double_side时，该字段仅在front字典中存在。 仅在输入参数return_verification为true时，返回该字段。

表 4-85 IdcardBackResult

参数	参数类型	描述
issue	String	发证机关。
valid_from	String	有效起始日期。
valid_to	String	有效结束日期。
adjusted_image	String	身份证卡面图片信息的base64码结果。 **说明** ● 仅在输入参数return_adjusted_image为true时，返回该字段。
verification_result	**IdcardBackVerificationResult** object	校验信息。 仅在输入参数return_verification为true时，返回该字段，该字段包含valid_date和valid_validity_period的校验信息。
text_location	Object	文本框在原图位置。输出左上、右上、右下、左下四个点坐标。 仅return_text_location设置为true时才返回。
detect_reproduce_result	Boolean	身份证图像是否翻拍告警结果。 ● true：表示身份证图片经过翻拍。 ● false：表示身份证图片未经过翻拍。 仅在输入参数detect_reproduce为true时，返回该字段。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
detect_copy_result	Boolean	身份证图像是否黑白复印件告警结果。 ● true：表示身份证图片是复印件。 ● false”表示身份证图片是原件。 仅在输入参数detect_copy为true时，返回该字段。
detect_tampering_result	Boolean	身份证图片是否PS告警结果。 ● true：表示身份证经过PS。 ● false：表示未经过PS。 仅在传入参数detect_tampering为true时，返回该字段。
detect_border_integrity_result	Boolean	身份证图片边框完整性告警结果。 ● true：表示边框不完整 ● false：表示边框完整。 仅在输入参数detect_border_integrity为true时，返回该字段。
detect_blocking_within_border_result	Boolean	身份证图像框内是否存在遮挡的告警结果。 ● true：表示边框内部存在遮挡。 ● false：表示边框内部不存在遮挡。 仅在输入参数detect_blocking_within_border为true时，返回该字段。
detect_blur_result	Boolean	身份证模糊告警结果。 ● true：表示身份证图片较模糊。 ● false：表示身份证清晰。 仅在输入参数detect_blur为true时，返回该字段。
detect_interim_result	Boolean	临时身份证告警结果。 ● true：表示是临时身份证。 ● false：表示非临时身份证。 仅在输入参数detect_interim为true时，返回该字段。
detect_glare_result	Boolean	身份证反光告警结果。 ● true：表示身份证图片存在反光。 ● false：表示是身份证不存在反光。 仅在输入参数detect_glare为true时，返回该字段。
score_info	**IdcardScoreInfoResult** object	告警分数，包含字段：tampering_score、interim_score、reproduce_score、copy_score、border_integrity_score、blur_score、glare_score、blocking_within_border_score，以上字段取值范围为[0, 99]。

版权所有 © 华为云计算技术有限公司

表 4-86 IdcardBackVerificationResult

参数	参数类型	描述
valid_date	Boolean	当前日期是否在有效期内。 ● true：表示当前日期在有效期内。 ● false：表示当前日期不在有效期内。 当识别结果为单页，身份证图片是人像面时，默认是false。输入参数side为double_side时，该字段仅在back字典中存在。 仅在输入参数return_verification为true时，返回该字段。
valid_validity_period	Boolean	身份证有效日期是否合法。 ● true：表示身份证的有效日期合法 ● false：表示身份证有效日期非法 当识别结果为单页，身份证图片是人像面时，默认是false。输入参数side为double_side时，该字段仅在back字典中存在。 仅在输入参数return_verification为true时，返回该字段。

表 4-87 IdcardScoreInfoResult

参数	参数类型	描述
tampering_score	Integer	身份证PS告警分数，分数越高，PS的可能性越高。仅在传入参数detect_tampering为true时，返回该字段。
interim_score	Integer	临时身份证告警分数，分数越高，临时身份证的可能性越高。仅在传入参数detect_interim为true时，返回该字段。
reproduce_score	Integer	身份证翻拍告警分数，分数越高，身份证图像经过翻拍的可能性越高。仅在传入参数detect_reproduce为true时，返回该字段。
copy_score	Integer	身份证复印告警分数, 分数越高，身份证图像是复印件的可能性越高。仅在传入参数detect_copy为true时，返回该字段。
border_integrity_score	Integer	身份证边缘完整性告警的分数，分数越高，身份证图像边缘不完整的可能性越高。仅在传入参数detect_border_integrity为true时，返回该字段。
blur_score	Integer	身份证模糊告警分数，分数越高，身份证图像模糊的可能性越高。仅在传入参数detect_blur为true时，返回该字段。
glare_score	Integer	身份证反光告警分数，分数越高，身份证图像存在反光的可能性越高。 仅在传入参数detect_glare为true时，返回该字段。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
blocking_within_border_score	Integer	身份证图像框内是否有遮挡的告警分数，分数越高，身份证图像框内存在异物遮挡的可能性越高。仅在传入参数detect_blocking_within_border为true时，返回该字段。

状态码： 400

表 4-88 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，身份证识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/id-card”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入身份证图片的base64编码进行文字识别，并对身份证号进行规则校验，同时判断身份证是否为翻拍、复印件

```
POST https://{endpoint}/v2/{project_id}/ocr/id-card
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
{
  "image" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...",
  "side" : "front",
  "return_text_location" : true,
  "return_verification" : true,
  "detect_reproduce" : true,
  "detect_copy" : true
}
```

- 传入身份证图片的url进行文字识别，并对身份证号进行规则校验，同时判断身份证是否为翻拍、复印件

```
POST https://{endpoint}/v2/{project_id}/ocr/id-card
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
```


版权所有 © 华为云计算技术有限公司

```
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
{
  "url" : "https://BucketName.obs.myhuaweicloud.com/ObjectName",
  "side" : "front",
  "return_text_location" : true,
  "return_verification" : true,
  "detect_reproduce" : true,
  "detect_copy" : true
}
```

响应示例

状态码：200

成功响应示例（人像面）

```
{
  "result" : {
    "name" : "张三",
    "sex" : "男",
    "ethnicity" : "汉",
    "birth" : "2000-03-06",
    "address" : "XX省XX市XX区XX街道XX号",
    "number" : "610XXXXXXXXXXXXXX",
    "adjusted_image" : "/9j/4AAQSkZJRgABAQAAAQAB....",
    "portrait_image" : "/9j/4AAQSkZJRgABAQAAAQABAAD....",
    "detect_reproduce_result" : true,
    "detect_copy_result" : true,
    "verification_result" : {
      "valid_number" : true,
      "valid_birth" : true,
      "valid_sex" : true,
      "valid_date" : false
    }
  }
}
```

成功响应示例（国徽面）

```
{
  "result" : {
    "issue" : "XX省XX市XX区XX派出所",
    "valid_from" : "2004-08-26",
    "valid_to" : "2034-08-26",
    "adjusted_image" : "/9j/4AAQSkZJRgABAQAAAQAB....",
    "portrait_image" : "",
    "verification_result" : {
      "valid_number" : false,
      "valid_birth" : false,
      "valid_sex" : false,
      "valid_date" : true
    },
    "detect_reproduce_result" : true,
    "detect_copy_result" : true
  }
}
```

成功响应示例(双面)

```
{
  "result" : {
    "front" : {
      "name" : "张三",
      "sex" : "男",
      "ethnicity" : "汉",
      "birth" : "1984-27-28",
      "address" : "杭州市xx区xx街xx路xx号",
      "number" : "372901xxxxxxxx4358",
      "adjusted_image" : "/9j/4AAQSkZJRgABAQAAAQAB....",
```

 版权所有 © 华为云计算技术有限公司

```
    "portrait_image" : "/9j/4AAQSkZJRgABAQAAAQABAAD....",
    "verification_result" : {
      "valid_number" : true,
      "valid_birth" : false,
      "valid_sex" : true
    },
    "detect_reproduce_result" : true,
    "detect_copy_result" : false,
    "score_info" : {
      "reproduce_score" : 49,
      "copy_score" : 0
    }
  },
  "back" : {
    "issue" : "xx县公安局",
    "valid_from" : "2006-08-30",
    "valid_to" : "2016-08-30",
    "adjusted_image" : "/9j/4AAQSkZJRgABAQAAAQABAAD....",
    "verification_result" : {
      "valid_date" : true,
      "valid_validity_period" : true
    },
    "detect_reproduce_result" : false,
    "detect_copy_result" : false,
    "score_info" : {
      "reproduce_score" : 23,
      "copy_score" : 0
    }
  }
 }
}
```

状态码：400

失败响应示例

```
{
   "error_code": "AIS.0103",
   "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入身份证图片的base64编码进行文字识别，并对身份证号进行规则校验，同时判断身份证是否为翻拍、复印件以及返回文字块的区域位置信息

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeIdCardSolution {
```

```
    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeIdCardRequest request = new RecognizeIdCardRequest();
        IdCardRequestBody body = new IdCardRequestBody();
        body.withDetectCopy(true);
        body.withDetectReproduce(true);
        body.withReturnTextLocation(true);
        body.withReturnVerification(true);
        body.withSide("front");
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeIdCardResponse response = client.recognizeIdCard(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入身份证图片的url进行文字识别，并对身份证号进行规则校验，同时判断身份证是否为翻拍、复印件以及返回文字块的区域位置信息

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeIdCardSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
```

 版权所有 © 华为云计算技术有限公司

```
        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeIdCardRequest request = new RecognizeIdCardRequest();
        IdCardRequestBody body = new IdCardRequestBody();
        body.withDetectCopy(true);
        body.withDetectReproduce(true);
        body.withReturnTextLocation(true);
        body.withReturnVerification(true);
        body.withSide("front");
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeIdCardResponse response = client.recognizeIdCard(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入身份证图片的base64编码进行文字识别，并对身份证号进行规则校验，同时判断身份证是否为翻拍、复印件以及返回文字块的区域位置信息

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeIdCardRequest()
        request.body = IdCardRequestBody(
            detect_copy=True,
            detect_reproduce=True,
            return_text_location=True,
            return_verification=True,
```

 版权所有 © 华为云计算技术有限公司

```
            side="front",
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_id_card(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入身份证图片的url进行文字识别，并对身份证号进行规则校验，同时判断身份证是否为翻拍、复印件以及返回文字块的区域位置信息

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeIdCardRequest()
        request.body = IdCardRequestBody(
            detect_copy=True,
            detect_reproduce=True,
            return_text_location=True,
            return_verification=True,
            side="front",
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_id_card(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入身份证图片的base64编码进行文字识别，并对身份证号进行规则校验，同时判断身份证是否为翻拍、复印件以及返回文字块的区域位置信息

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)
```

 版权所有 © 华为云计算技术有限公司

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeIdCardRequest{}
    detectCopyIdCardRequestBody:= true
    detectReproduceIdCardRequestBody:= true
    returnTextLocationIdCardRequestBody:= true
    returnVerificationIdCardRequestBody:= true
    sideIdCardRequestBody:= "front"
    imageIdCardRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.IdCardRequestBody{
        DetectCopy: &detectCopyIdCardRequestBody,
        DetectReproduce: &detectReproduceIdCardRequestBody,
        ReturnTextLocation: &returnTextLocationIdCardRequestBody,
        ReturnVerification: &returnVerificationIdCardRequestBody,
        Side: &sideIdCardRequestBody,
        Image: &imageIdCardRequestBody,
    }
    response, err := client.RecognizeIdCard(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入身份证图片的url进行文字识别，并对身份证号进行规则校验，同时判断身份证是否为翻拍、复印件以及返回文字块的区域位置信息

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()
```

 版权所有 © 华为云计算技术有限公司

```
    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeIdCardRequest{}
    detectCopyIdCardRequestBody:= true
    detectReproduceIdCardRequestBody:= true
    returnTextLocationIdCardRequestBody:= true
    returnVerificationIdCardRequestBody:= true
    sideIdCardRequestBody:= "front"
    urlIdCardRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.IdCardRequestBody{
        DetectCopy: &detectCopyIdCardRequestBody,
        DetectReproduce: &detectReproduceIdCardRequestBody,
        ReturnTextLocation: &returnTextLocationIdCardRequestBody,
        ReturnVerification: &returnVerificationIdCardRequestBody,
        Side: &sideIdCardRequestBody,
        Url: &urlIdCardRequestBody,
    }
    response, err := client.RecognizeIdCard(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.9 户口本识别

功能介绍

识别户口本中的文字信息，并返回识别的结构化结果。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

版权所有 © 华为云计算技术有限公司

约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式的图片。
- 图像各边的像素在15到8192px之间。单个图片对应的Base64编码不超过10MB。
- 支持多页同时识别。
- 能处理反光、暗光、水印等干扰的图片但影响识别精度。
- 文字识别服务属于公有云服务，线上用户资源共享，如果需要多并发请求，请提前**联系我们**。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/household-register

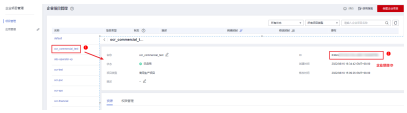
表 4-89 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

版权所有 © 华为云计算技术有限公司

请求参数

表 4-90 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-91 请求 Body 参数

参数	是否必选	参数类型	描述
image	否	String	与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最小边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

版权所有 © 华为云计算技术有限公司

表 4-92 响应 Body 参数

参数	参数类型	描述
result	Array of **HouseholdRegisterResult** objects	调用成功时表示调用结果。 调用失败时无此字段。

表 4-93 HouseholdRegisterResult

参数	参数类型	描述
type	String	类型。参数为“首页”或“登记页”。
location	Array<Array<Integer>>	户口本证件位置信息，列表形式，包含证件位置四个顶点的二维坐标（x,y）;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。
content	**HouseholdRegisterContent** object	户口本具体信息内容。
confidence	Map<String,Number>	content中各个字段的置信度，取值范围0~1。置信度越大，本次识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。置信度由算法给出，不直接等价于字段的准确率。

表 4-94 HouseholdRegisterContent

参数	参数类型	描述
name	String	姓名。
householder_relationship	String	户主或与户主关系。
former_name	String	曾用名。
sex	String	性别。
birthplace	String	出生地。
ethnicity	String	民族。
origin_place	String	籍贯。
birth_date	String	出生日期。
other_address	String	本市(县)其他住址。
religious_belief	String	宗教信仰。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
id_card_number	String	公民身份证件编号。
height	String	身高。
blood_type	String	血型。
education	String	文化程度。
marital_status	String	婚姻状况。
military_service_status	String	兵役情况。
work_place	String	服务处所。
occupation	String	职业。
migrated_to_city	String	何时由何地迁来本市(县)。
migrated_to_address	String	何时由何地迁来本址。
registrar_signature_seal	String	承办人签章。
registration_date	String	登记日期。
household_type	String	户别。
household_number	String	户号。
householder_name	String	户主姓名。当type参数为“首页”时，返回此参数。
community	String	社区。当type参数为“首页”时，返回此参数。
address	String	住址。当type参数为“首页”时，返回此参数。
issue_date	String	签发日期。当type参数为“首页”时，返回此参数。
police_station	String	户口登记机关。当type参数为“首页”时，返回此参数。

状态码： 400

版权所有 © 华为云计算技术有限公司

表 4-95 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见错误码。 调用成功时无此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，户口本识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/household-register”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入户口本图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/household-register

{
  "image" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入户口本图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/household-register

{
  "url" : "https://BucketName.obs.myhuaweicloud.com/ObjectName"
}
```

响应示例

状态码： 200

成功响应示例

```
{
  "result" : [ {
    "type" : "登记页",
    "location" : [ [ 24, 372 ], [ 528, 372 ], [ 528, 738 ], [ 24, 738 ] ],
    "content" : {
      "name" : "张三",
      "householder_relationship" : "子",
      "former_name" : "xxx",
      "sex" : "男",
      "birthplace" : "湖南省衡山县",
      "ethnicity" : "汉",
      "origin_place" : "湖南省衡山县",
      "birth_date" : "19xx年8月16日",
      "other_address" : "湖南省衡山县",
      "religious_belief" : "xxx",
      "id_card_number" : "xxxx",
      "height" : "xxx",
      "blood_type" : "xxx",
      "education" : "初中",
      "marital_status" : "未婚",
      "military_service_status" : "xxx",
```


版权所有 © 华为云计算技术有限公司

```
    "work_place" : "江东村",
    "occupation" : "种情业牛产人员",
    "migrated_to_city" : "xxx",
    "migrated_to_address" : "xxx",
    "registration_date" : "2010年08月30日",
    "registrar_signature_seal" : "xxx",
    "household_type" : "居民家庭户口",
    "household_number" : "xxx"
   }
  }, {
   "type" : "首页",
   "location" : [ [ 2, 4 ], [ 502, 4 ], [ 502, 367 ], [ 2, 367 ] ],
   "content" : {
    "household_type" : "居民家庭户口",
    "householder_name" : "张二",
    "community" : "xx社区居委会",
    "household_number" : "xxxx32467",
    "address" : "湖南省衡山县",
    "police_station" : "xxx",
    "issue_date" : "2010年08月30日",
    "registrar_signature_seal" : "xxx"
   }
  } ]
}
```

状态码： 400

失败响应示例

```
{
  "result" : {
    "error_code" : "AIS.0103",
    "error_msg" : "The image size does not meet the requirements."
  }
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入户口本图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeHouseholdRegisterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
```

```
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeHouseholdRegisterRequest request = new RecognizeHouseholdRegisterRequest();
        HouseholdRegisterRequestBody body = new HouseholdRegisterRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeHouseholdRegisterResponse response = client.recognizeHouseholdRegister(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入户口本图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeHouseholdRegisterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeHouseholdRegisterRequest request = new RecognizeHouseholdRegisterRequest();
        HouseholdRegisterRequestBody body = new HouseholdRegisterRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
```

 版权所有 © 华为云计算技术有限公司

```
        try {
            RecognizeHouseholdRegisterResponse response = client.recognizeHouseholdRegister(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入户口本图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeHouseholdRegisterRequest()
        request.body = HouseholdRegisterRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_household_register(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入户口本图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
```

 版权所有 © 华为云计算技术有限公司

```
    # In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeHouseholdRegisterRequest()
        request.body = HouseholdRegisterRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_household_register(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入户口本图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeHouseholdRegisterRequest{}
    imageHouseholdRegisterRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/
4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.HouseholdRegisterRequestBody{
        Image: &imageHouseholdRegisterRequestBody,
    }
    response, err := client.RecognizeHouseholdRegister(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
```

 版权所有 © 华为云计算技术有限公司

```
    } else {
        fmt.Println(err)
    }
}
```

- 传入户口本图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeHouseholdRegisterRequest{}
    urlHouseholdRegisterRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.HouseholdRegisterRequestBody{
        Url: &urlHouseholdRegisterRequestBody,
    }
    response, err := client.RecognizeHouseholdRegister(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

 版权所有 © 华为云计算技术有限公司

错误码

错误码请参见**错误码**。

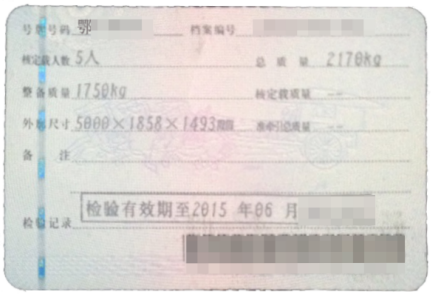
4.10 行驶证识别

功能介绍

识别行驶证图片中主页和副页的文字内容，并将识别的结果以JSON格式返回给用户。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-7 行驶证示例图







说明

如果图片中包含多张卡证票据，请调用**智能分类识别**服务。

版权所有 © 华为云计算技术有限公司

约束与限制

- 只支持中国大陆行驶证的识别。
- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 图像各边的像素大小在100px到8000px之间。单个图片对应的Base64编码不超过10MB。
- 图像中行驶证区域有效占比超过50%，保证整张行驶证内容及其边缘包含在图像内。
- 支持图像中行驶证任意角度的水平旋转。
- 支持少量扭曲，扭曲后图像中的行驶证长宽比与实际行驶证相差不超过10%。
- 能处理反光、暗光、防伪标识等干扰的图片但影响识别精度。
- 目前只支持识别2008年版的行驶证和电子行驶证。
- 仅支持纸质行驶证的框内遮挡、边框完整性告警检测。
- 告警检测结果仅作为参考依据，实际图片情况可能与检测结果存在差异，可结合其他校验方式，如字段是否完整等共同判断。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/vehicle-license

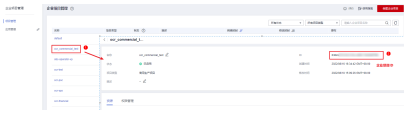
表 4-96 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

版权所有 © 华为云计算技术有限公司

请求参数

表 4-97 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-98 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最短边不小于100px，最长边不超过8000px。支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
side	否	String	检测行驶证的主页或副页。 • front: 行驶证主页 • back: 行驶证副页 • double_side：行驶证双页信息 如果参数值为空或无该参数，系统默认识别主页，建议填写，准确率更高。
return_issuing_authority	否	Boolean	是否返回发证机关的开关，可选值包括： • true：返回发证机关 • false：不返回发证机关 如果无该参数，系统默认不返回发证机关。如果输入参数不是Boolean类型，则会报非法参数错误。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
return_text_location	否	Boolean	识别到的文字块的区域位置信息。可选值包括： true：返回各个文字块区域 false：不返回各个文字块区域 如果无该参数，系统默认不返回文字块区域。如果输入参数不是Boolean类型，则会报非法参数错误。
recognize_electronic_license	否	Boolean	是否支持识别电子行驶证，可选值包括： • true：支持识别电子行驶证 • false：不支持识别电子行驶证 如果无该参数，系统默认不支持识别电子行驶证。如果输入参数不是Boolean类型，则会报非法参数错误
alarm	否	Boolean	是否返回纸质行驶证图像的告警信息，可选值包括： • true：返回纸质行驶证图像的告警信息 • false：不返回纸质行驶证图像的告警信息 如果无该参数，系统默认不返回告警信息。如果输入参数不是Boolean类型，则会报非法参数错误。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-99 响应 Body 参数

参数	参数类型	描述
result	VehicleLicenseResult object	识别结果。 调用失败时不返回此字段。

版权所有 © 华为云计算技术有限公司

表 4-100 VehicleLicenseResult

参数	参数类型	描述
type	String	行驶证类型。normal: 纸质行驶证 electronic: 电子行驶证
number	String	号牌号码。
vehicle_type	String	车辆类型。
name	String	所有人。
address	String	住址。
use_character	String	使用性质。
model	String	品牌型号。
engine_no	String	发动机号码。
vin	String	车辆识别代号。
register_date	String	注册日期。
issue_date	String	发证日期。
issuing_authority	String	发证机关。
file_no	String	档案编码。纸质行驶证独有字段。
approved_passengers	String	核定载人数。
gross_mass	String	总质量。
unladen_mass	String	整备质量。
approved_load	String	核定载质量。
dimension	String	外廓尺寸。
traction_mass	String	准牵引总质量。
remarks	String	备注。
inspection_record	String	检验记录。
code_number	String	条码号。
text_location	Object	对应所有在原图上识别到的字段位置信息，包含所有文字区域四个顶点的二维坐标（x,y）。采用图像坐标系，坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。
energy_type	String	能源类型。
color	String	车身颜色，电子行驶证独有字段。
mandatory_scrapping_date	String	强制报废日期，电子行驶证独有字段。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
status	Array of strings	状态，电子行驶证独有字段。
front	表4-101 object	行驶证主页。当请求参数"side"设置为"double_side"时才返回。纸质行驶证返回方式。
back	表4-102 object	行驶证副页。当请求参数"side"设置为"double_side"时才返回。纸质行驶证返回方式。
alarm_result	MyanmarIdcardAlarmResult object	图像告警检测结果。 仅在输入参数alarm为true时，返回该字段，包含字段：detect_blocking_within_border_result、detect_border_integrity_result
alarm_confidence	MyanmarIdcardAlarmConfidence object	图像告警检测结果分数。 仅在输入参数alarm为true时，返回该字段，包含字段： blocking_within_border_score、border_integrity_score，以上字段取值范围为[0,99]。

表 4-101 VehicleLicenseFront

参数	参数类型	描述
number	String	号牌号码。
vehicle_type	String	车辆类型。
name	String	所有人。
address	String	住址。
use_character	String	使用性质。
model	String	品牌型号。
vin	String	车辆识别代号。
engine_no	String	发动机号码。
register_date	String	注册日期。
issue_date	String	发证日期。
issuing_authority	String	发证机关。
text_location	Object	文本框在原图位置。输出左上、右上、右下、左下四个点坐标。 当“return_text_location”设置为“true”时才返回。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
alarm_result	MyanmarIdcardAlarmResult object	图像告警检测结果。 仅在输入参数alarm为true时，返回该字段，包含字段：detect_blocking_within_border_result、detect_border_integrity_result。
alarm_confidence	MyanmarIdcardAlarmConfidence object	图像告警检测结果分数。 仅在输入参数alarm为true时，返回该字段，包含字段：blocking_within_border_score、border_integrity_score，以上字段取值范围为[0, 99]。

表 4-102 VehicleLicenseback

参数	参数类型	描述
number	String	号牌号码。
file_no	String	档案编码。
approved_passengers	String	核定载人数。
gross_mass	String	总质量。
unladen_mass	String	整备质量。
approved_load	String	核定载质量。
dimension	String	外廓尺寸。
traction_mass	String	准牵引总质量。
remarks	String	备注。
inspection_record	String	检验记录。
code_number	String	条码号。
energy_type	String	能源类型。
text_location	Object	文本框在原图位置。输出左上、右上、右下、左下四个点坐标。 当“return_text_location”设置为“true”时才返回。
alarm_result	MyanmarIdcardAlarmResult object	图像告警检测结果。 仅在输入参数alarm为true时，返回该字段，包含字段：detect_blocking_within_border_result、detect_border_integrity_result

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
alarm_confidence	MyanmarIdcardAlarmConfidence object	图像告警检测结果分数。 仅在输入参数alarm为true时，返回该字段，包含字段：blocking_within_border_score、border_integrity_score，以上字段取值范围为[0, 99]。

表 4-103 MyanmarIdcardAlarmResult

参数	参数类型	描述
detect_blocking_within_border_result	Boolean	证件图像框内遮挡告警结果。 true：表示证件图片内部有被遮挡。 false：表示证件图片内部未被遮挡。
detect_border_integrity_result	Boolean	证件图像边框完整性告警结果。 true：表示边框不完整。 false：表示边框完整。

表 4-104 MyanmarIdcardAlarmConfidence

参数	参数类型	描述
blocking_within_border_score	Integer	证件图像框内遮挡告警分数，分数越高，证件图像框内遮挡的可能性越高。
border_integrity_score	Integer	证件图像边框完整性告警分数，分数越高，证件图像边框不完整的可能性越高。

状态码： 400

表 4-105 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

版权所有 © 华为云计算技术有限公司

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，行驶证识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/vehicle-license”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入行驶证主页图片的base64编码进行文字识别，并识别发证机关信息

```
POST https://{endpoint}/v2/{project_id}/ocr/vehicle-license
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
   "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAg...",
   "side":"front",
   "return_issuing_authority":true
 }
```

- 传入行驶证副页图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/vehicle-license
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
    "url":"https://BucketName.obs.xxxx.com/ObjectName",
    "side":"back"
 }
```

- 传入电子行驶证图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/vehicle-license
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
    "url":"https://BucketName.obs.xxxx.com/ObjectName",
    "recognize_electronic_license":true
 }
```

- 传入行驶证图片的url进行文字识别，并返回告警信息

```
POST https://{endpoint}/v2/{project_id}/ocr/vehicle-license
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
    "url":"https://BucketName.obs.xxxx.com/ObjectName",
    "alarm":true
 }
```

- 使用Python3语言读取本地行驶证图片，并进行文字识别

```
# encoding:utf-8

import requests
import base64

url = "https://{endpoint}/v2/{project_id}/ocr/vehicle-license"
```


版权所有 © 华为云计算技术有限公司

```
token = "用户获取得到的实际token值"
headers = {'Content-Type': 'application/json', 'X-Auth-Token': token}

imagepath = r'./data/vehicle-license-demo.png' # 读取本地图片
with open(imagepath, "rb") as bin_data:
    image_data = bin_data.read()
image_base64 = base64.b64encode(image_data).decode("utf-8")  # 使用图片的Base64编码
payload = {"image": image_base64}
response = requests.post(url, headers=headers, json=payload)
print(response.text)
```

响应示例

状态码： 200

成功响应示例（纸质行驶证双页）

```
{
  "result" : {
    "front" : {
      "number" : "粤B0xxx2",
      "vehicle_type" : "小型轿车",
      "name" : "张三",
      "address" : "xx省xx市xx区xxx街道号",
      "engine_no" : "601xx99",
      "vin" : "LPAA3xxxxxxxxx709",
      "model" : "xx牌CAPxxxxxC3",
      "register_date" : "2015-07-07",
      "issue_date" : "2015-07-08",
      "use_character" : "非营运"
    },
    "back" : {
      "number" : "粤B8888",
      "file_no" : "44xxxxx57436",
      "approved_passengers" : "5人",
      "gross_mass" : "1960kg",
      "unladen_mass" : "1525kg",
      "approved_load" : "",
      "dimension" : "4553×1858×1588mm",
      "traction_mass" : "",
      "remarks" : "",
      "inspection_record" : "检验有效期至20xx年xx月粤B(12)汽油",
      "code_number" : "*44xxx24xxxx49*",
      "energy_type" : "汽油"
      },
    "type" : "normal"
  }
}
```

成功响应示例（电子行驶证双页）

```
{
   "result" : {
     "number" : "粤B0xxx2",
     "vehicle_type" : "小型轿车",
     "name" : "张三",
     "status" : ["正常"],
     "model" : "xx牌CAPxxxxxC3",
     "engine_no" : "601xx99",
     "vin" : "LPAA3xxxxxxxxx709",
     "register_date" : "2015-07-07",
     "issue_date" : "2015-07-08",
     "use_character" : "非营运",
     "approved_passengers" : "5人",
     "gross_mass" : "1960kg",
     "unladen_mass" : "1525kg",
     "approved_load" : "",
     "dimension" : "4553×1858×1588mm",
```

 版权所有 © 华为云计算技术有限公司

```
    "traction_mass" : "",
    "remarks" : "",
    "inspection_record" : "检验有效期至20xx年xx月",
    "code_number" : "*44xxx24xxxx49*",
    "energy_type" : "汽油",
    "address" : "xx省xx市xx区xxx街道号",
    "mandatory_scrapping_date" : "",
    "color" : "白",
    "type" : "electronic"
  }
}
```

状态码： 400

失败响应示例

```
{
  "result" : {
    "error_code" : "AIS.0103",
    "error_msg" : "The image size does not meet the requirements."
  }
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入行驶证主页图片的base64编码进行文字识别，并识别发证机关信息

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeVehicleLicenseSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeVehicleLicenseRequest request = new RecognizeVehicleLicenseRequest();
```

 版权所有 © 华为云计算技术有限公司

```
        VehicleLicenseRequestBody body = new VehicleLicenseRequestBody();
        body.withReturnIssuingAuthority(true);
        body.withSide("front");
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAg...");
        request.withBody(body);
        try {
            RecognizeVehicleLicenseResponse response = client.recognizeVehicleLicense(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入行驶证副页图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeVehicleLicenseSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeVehicleLicenseRequest request = new RecognizeVehicleLicenseRequest();
        VehicleLicenseRequestBody body = new VehicleLicenseRequestBody();
        body.withSide("back");
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeVehicleLicenseResponse response = client.recognizeVehicleLicense(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
```

 版权所有 © 华为云计算技术有限公司

```
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入行驶证主页图片的base64编码进行文字识别，并识别发证机关信息

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeVehicleLicenseRequest()
        request.body = VehicleLicenseRequestBody(
            return_issuing_authority=True,
            side="front",
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAg..."
        )
        response = client.recognize_vehicle_license(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入行驶证副页图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \
```

 版权所有 © 华为云计算技术有限公司

```
    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeVehicleLicenseRequest()
        request.body = VehicleLicenseRequestBody(
            side="back",
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_vehicle_license(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入行驶证主页图片的base64编码进行文字识别，并识别发证机关信息

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeVehicleLicenseRequest{}
    returnIssuingAuthorityVehicleLicenseRequestBody:= true
    sideVehicleLicenseRequestBody:= "front"
    imageVehicleLicenseRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAg..."
    request.Body = &model.VehicleLicenseRequestBody{
        ReturnIssuingAuthority: &returnIssuingAuthorityVehicleLicenseRequestBody,
        Side: &sideVehicleLicenseRequestBody,
        Image: &imageVehicleLicenseRequestBody,
    }
    response, err := client.RecognizeVehicleLicense(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
```

 版权所有 © 华为云计算技术有限公司

```
    }
}
```

- 传入行驶证副页图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeVehicleLicenseRequest{}
    sideVehicleLicenseRequestBody:= "back"
    urlVehicleLicenseRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.VehicleLicenseRequestBody{
        Side: &sideVehicleLicenseRequestBody,
        Url: &urlVehicleLicenseRequestBody,
    }
    response, err := client.RecognizeVehicleLicense(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

版权所有 © 华为云计算技术有限公司

错误码

错误码请参见**错误码**。

4.11 驾驶证识别

功能介绍

识别驾驶证图片中主页与副页的文字内容，并将识别的结果以JSON格式返回给用户。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-8 驾驶证示例图



图 4-9 电子驾驶证示例图



版权所有 © 华为云计算技术有限公司

说明

如果图片中包含多张卡证票据，请调用**智能分类识别**服务。

约束与限制

- 只支持中国大陆驾驶证纸质版本及交管12123提供的电子驾驶证版本识别。
- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 图像各边的像素大小在100px到8000px之间。单个图片对应的Base64编码不超过10MB。
- 图像中驾驶证区域有效占比超过50%，保证整张驾驶证内容及其边缘包含在图像内。
- 支持图像中驾驶证任意角度的水平旋转。
- 支持少量扭曲，扭曲后图像中的驾驶证长宽比与实际驾驶证相差不超过10%。
- 能处理反光、暗光、防伪标识等干扰的图片但影响识别精度。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/driver-license

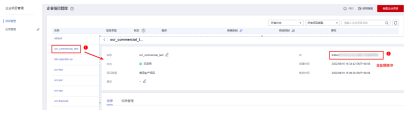
表 4-106 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

版权所有 © 华为云计算技术有限公司

请求参数

表 4-107 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-108 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最短边不小于100px，最长边不超过8000px。支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
side	否	String	检测驾驶证的主页、副页。 • front: 驾驶证主页 • back: 驾驶证副页 • double_side：驾驶证双页信息 如果参数值为空或无该参数，系统默认识别主页，建议填写，准确率更高。
return_issuing_authority	否	Boolean	是否返回发证机关的开关，可选值包括： true：返回发证机关 false：不返回发证机关 如果无该参数，系统默认不返回发证机关。如果输入参数不是Boolean类型，则会报非法参数错误AIS.0101。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
return_text_location	否	Boolean	识别到的文字块的区域位置信息。可选值包括： true：返回各个文字块区域 false：不返回各个文字块区域 如果无该参数，系统默认不返回文字块区域。如果输入参数不是Boolean类型，则会报非法参数错误。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（status code）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-109 响应 Body 参数

参数	参数类型	描述
result	DriverLicenseResult object	识别结果。 调用失败时不返回此字段。

表 4-110 DriverLicenseResult

参数	参数类型	描述
type	String	驾驶证类型。 normal：纸质驾驶证 electronic：电子驾驶证
number	String	驾驶证号。
name	String	姓名。
sex	String	性别。
nationality	String	国籍。
address	String	住址。
birth	String	出生日期。
issue_date	String	初次领证日期。
class	String	准驾类型。
valid_from	String	有效起始日期。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
valid_to	String	有效结束日期。
issuing_authority	String	发证机关。
file_number	String	档案编号。
record	String	记录。
accumulated_scores	String	累积记分。
status	Array of strings	状态。
generation_date	String	生成时间。
current_time	String	当前时间。
text_location	Object	对应所有在原图上识别到的字段位置信息，包含所有文字区域四个顶点的二维坐标（x,y）。采用图像坐标系，坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。
front	**DriverLicenseFront** object	驾驶证主页。 当请求参数"side"设置为"double_side"时才返回。
back	**DriverLicenseBack** object	驾驶证副页。 当请求参数"side"设置为"double_side"时才返回。

表 4-111 DriverLicenseFront

参数	参数类型	描述
type	String	驾驶证类型。 normal：纸质驾驶证 electronic：电子驾驶证
number	String	驾驶证号。
name	String	姓名。
sex	String	性别。
nationality	String	国籍。
address	String	住址。
birth	String	出生日期。
issue_date	String	初次领证日期。
class	String	准驾类型。
valid_from	String	有效起始日期。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
valid_to	String	有效结束日期。
issuing_authority	String	发证机关。
accumulated_scores	String	累积记分。
status	Array of strings	状态。
generation_date	String	生成时间。
current_time	String	当前时间。
file_number	String	档案编号。
text_location	Object	对应所有在原图上识别到的字段位置信息，包含所有文字区域四个顶点的二维坐标（x,y）。采用图像坐标系，坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。

表 4-112 DriverLicenseBack

参数	参数类型	描述
type	String	驾驶证类型。 normal：纸质驾驶证 electronic：电子驾驶证
number	String	驾驶证号。
name	String	姓名。
issuing_authority	String	发证机关。
address	String	住址。
file_number	String	档案编号。
record	String	记录。
text_location	Object	对应所有在原图上识别到的字段位置信息，包含所有文字区域四个顶点的二维坐标（x,y）。采用图像坐标系，坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。

状态码： 400

版权所有 © 华为云计算技术有限公司

表 4-113 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，驾驶证识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/driver-license”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入驾驶证主页图片的base64编码进行文字识别，并识别发证机关信息

```
POST https://{endpoint}/v2/{project_id}/ocr/driver-license
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
   "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...",
   "side":"front",
   "return_issuing_authority": true
 }
```

- 传入驾驶证副页图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/driver-license
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
    "url":"https://BucketName.obs.xxxx.com/ObjectName",
    "side":"back"
 }
```

响应示例

状态码：200

成功响应示例（纸质驾驶证主页）

```
{
  "result" : {
    "type" : "normal",
    "number" : "362XXXXXXXXXXXXXXX",
    "name" : "张三",
    "sex" : "男",
    "nationality" : "中国",
    "address" : "XX省XX市XX区XX街道XX号",
```


版权所有 © 华为云计算技术有限公司

```
    "birth" : "2000-03-06",
    "issue_date" : "2000-03-06",
    "class" : "C1",
    "valid_from" : "2004-08-26",
    "valid_to" : "2034-08-26"
  }
}
```

成功响应示例（纸质驾驶证副页）

```
{
  "result" : {
    "type" : "normal",
    "number" : "362XXXXXXXXXXXXXXX",
    "name" : "张三",
    "file_number" : "880008886666",
    "record" : "实习期至2005年08月26日。"
  }
}
```

状态码：400

失败响应示例

```
{
  "error_code" : "AIS.0103",
  "error_msg" : "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入驾驶证主页图片的base64编码进行文字识别，并识别发证机关信息

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeDriverLicenseSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);
```


版权所有 © 华为云计算技术有限公司

```
        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeDriverLicenseRequest request = new RecognizeDriverLicenseRequest();
        DriverLicenseRequestBody body = new DriverLicenseRequestBody();
        body.withReturnIssuingAuthority(true);
        body.withSide("front");
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAg...");
        request.withBody(body);
        try {
            RecognizeDriverLicenseResponse response = client.recognizeDriverLicense(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入驾驶证副页图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeDriverLicenseSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeDriverLicenseRequest request = new RecognizeDriverLicenseRequest();
        DriverLicenseRequestBody body = new DriverLicenseRequestBody();
        body.withSide("back");
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeDriverLicenseResponse response = client.recognizeDriverLicense(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入驾驶证主页图片的base64编码进行文字识别，并识别发证机关信息

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeDriverLicenseRequest()
        request.body = DriverLicenseRequestBody(
            return_issuing_authority=True,
            side="front",
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAg..."
        )
        response = client.recognize_driver_license(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入驾驶证副页图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
```

 版权所有 © 华为云计算技术有限公司

```
environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeDriverLicenseRequest()
        request.body = DriverLicenseRequestBody(
            side="back",
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_driver_license(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入驾驶证主页图片的base64编码进行文字识别，并识别发证机关信息

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeDriverLicenseRequest{}
    returnIssuingAuthorityDriverLicenseRequestBody:= true
    sideDriverLicenseRequestBody:= "front"
    imageDriverLicenseRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAg..."
    request.Body = &model.DriverLicenseRequestBody{
        ReturnIssuingAuthority: &returnIssuingAuthorityDriverLicenseRequestBody,
        Side: &sideDriverLicenseRequestBody,
        Image: &imageDriverLicenseRequestBody,
    }
```

 版权所有 © 华为云计算技术有限公司

```
    response, err := client.RecognizeDriverLicense(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入驾驶证副页图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeDriverLicenseRequest{}
    sideDriverLicenseRequestBody:= "back"
    urlDriverLicenseRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.DriverLicenseRequestBody{
        Side: &sideDriverLicenseRequestBody,
        Url: &urlDriverLicenseRequestBody,
    }
    response, err := client.RecognizeDriverLicense(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例

版权所有 © 华为云计算技术有限公司

状态码	描述
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.12 护照识别

功能介绍

识别护照首页图片中的文字信息，并以JSON格式返回识别的结构化结果。

当前版本支持2012年及以后发行的中国普通电子护照的全字段识别。中国-港澳台地区、外国护照支持护照下方两行国际标准化的机读码识别，并可从中提取7个关键字段信息。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-10 护照示例图



说明

如果图片中包含多张卡证票据，请调用**智能分类识别**服务。

约束与限制

- 支持中国大陆护照的全字段识别。
- 支持含有完整机读码的中国-港澳台地区及外国护照识别。
- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中护照首页区域有效占比超过25%，保证护照首页内容及其边缘包含在图像内。
- 支持图像中护照任意角度的水平旋转。

版权所有 © 华为云计算技术有限公司

- 支持少量扭曲，扭曲后图像中的护照长宽比与实际护照相差不超过10%。
- 能处理反光、暗光等干扰的图片但影响识别精度。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**开通服务**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/passport

表 4-114 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-115 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

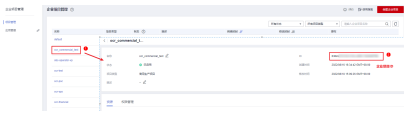
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-116 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
country_code	否	String	输入值为护照颁发国的国家码，根据国家码选择对应的护照识别服务。 • 若输入中未选此字段，服务会根据自己识别判断的护照类型匹配相应的护照识别服务。 • 若选择值为“GENERAL”，则选择护照机器码识别。 • 若选择值为“CHN”，则选择中国护照全字段识别。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（status code）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-117 响应 Body 参数

参数	参数类型	描述
result	**PassportResult** object	识别结果。 调用失败时不返回此字段。 该结果中包含：13个主要字段，由英文表达；extra_info由本地官方语言表达；主要字段的置信度。置信度越大，字段的值越准确。

版权所有 © 华为云计算技术有限公司

表 4-118 PassportResult

参数	参数类型	描述
passport_type	String	护照类型。 ● P:普通因私护照 ● W:外交护照 ● G:公务护照 **说明** 该字段仅限中国大陆护照。
country_code	String	护照签发国的国家码。
passport_number	String	护照号码。
nationality	String	护照持有人国籍。 **说明** 该字段仅限中国大陆护照。
surname	String	姓。
given_name	String	名字。
sex	String	性别。
date_of_birth	String	出生日期。返回值例如，1990-12-12。
date_of_expiry	String	护照有效期。返回值例如，2020-07-08。
date_of_issue	String	护照签发日期。返回值例如，2010-07-09。 **说明** 该字段仅限中国大陆护照。
place_of_birth	String	出生地。 **说明** 该字段仅限中国大陆护照。
place_of_issue	String	签发地。 **说明** 该字段仅限中国大陆护照。
issuing_authority	String	签发机关。 因为各个领事馆签发机关简写未统一，所以服务统一对中国的英文简写处理为“P.R.China”。例如：有的签发机关是“P.R.C”，识别结果均显示“P.R.China”。 **说明** 该字段仅限中国大陆护照。
confidence	Object	字段的置信度，取值范围0~1。 置信度越大，表示识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于字段的准确率。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
extra_info	Object	参数值默认为空。对于中国护照，“extra_info”的值会包含护照上由汉字描述的字段信息，如姓名、出生地等信息。 **说明** 该字段仅限中国大陆护照。

状态码： 400

表 4-119 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，护照识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com或ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/passport”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 读取护照图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/passport
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...",
    "country_code": "GENERAL"
}
```

- 读取护照图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/passport
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "url":"https://BucketName.obs.xxxx.com/ObjectName",
    "country_code": "GENERAL"
}
```


版权所有 © 华为云计算技术有限公司

响应示例

状态码：200

中国护照


```
{
   "result": {
      "passport_type": "P",
      "country_code": "CHN",
      "passport_number": "ED999XXXX",
      "nationality": "CHINESE",
      "surname": "ZHANG",
      "given_name": "SAN",
      "sex": "F",
      "date_of_birth": "1990-12-12",
      "date_of_expiry": "2020-07-08",
      "date_of_issue": "2010-07-09",
      "place_of_birth": "HUNAN",
      "place_of_issue": "GUANGDONG",
      "issuing_authority": "MPS Exit & Entry Administration",
      "extra_info": {
         "local_language": {
            "name": "张三",
            "sex": "女",
            "place_of_birth": "湖南",
            "place_of_issue": "广东",
            "issuing_authority": "xxx出入境管理局",
            "nationality": "中国"
         }
      },
      "confidence": {
         "passport_type": 0.9987,
         "country_code": 0.9897,
         "passport_number": 0.9997,
         "nationality": 0.9977,
         "surname": 0.9729,
         "given_name": 0.9729,
         "sex": 0.9897,
         "date_of_birth": 0.9998,
         "date_of_expiry": 0.9995,
         "date_of_issue": 0.9969,
         "place_of_birth": 0.9937,
         "place_of_issue": 0.9993,
         "issuing_authority": 0.9985
      }
   }
}
```


外国护照


```
{
   "result": {
      "country_code": "ETF",
      "surname": "HUZHAO",
      "given_name": "ZHAOMIN DESALEGN ",
      "passport_number": "EP435XXXX",
      "date_of_birth": "1985-09-18",
      "sex": "M",
      "date_of_expiry": "2022-01-15",
      "machine_code": "P<ETFHUZHAO<< ZHAOMIN <DESALEGN<<<<<<<<<<<<<<<",
      "machine_code2": "EP435XXXX7ETF8509185M2201155<<<<<<<<<<<<<<08",
      "extra_info": {},
      "confidence": {
         "country_code": 0.9727,
         "surname": 0.9727,
         "given_name": 0.9727,
         "passport_number": 0.9558,
         "date_of_birth": 0.9558,
```


版权所有 © 华为云计算技术有限公司

```
            "sex": 0.9558,
            "date_of_expiry": 0.9558
        }
    }
}
```

状态码：400

失败响应示例

```
{
    "error_code": "AIS.0103",
    "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 读取护照图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizePassportSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizePassportRequest request = new RecognizePassportRequest();
        PassportRequestBody body = new PassportRequestBody();
        body.withCountryCode("CHN");
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAg...");
        request.withBody(body);
        try {
            RecognizePassportResponse response = client.recognizePassport(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 读取护照图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizePassportSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizePassportRequest request = new RecognizePassportRequest();
        PassportRequestBody body = new PassportRequestBody();
        body.withCountryCode("CHN");
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizePassportResponse response = client.recognizePassport(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

 版权所有 © 华为云计算技术有限公司

Python

- 读取护照图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizePassportRequest()
        request.body = PassportRequestBody(
            country_code="CHN",
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAg..."
        )
        response = client.recognize_passport(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 读取护照图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizePassportRequest()
        request.body = PassportRequestBody(
            country_code="CHN",
```


版权所有 © 华为云计算技术有限公司

```
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_passport(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 读取护照图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizePassportRequest{}
    countryCodePassportRequestBody:= "CHN"
    imagePassportRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAg..."
    request.Body = &model.PassportRequestBody{
        CountryCode: &countryCodePassportRequestBody,
        Image: &imagePassportRequestBody,
    }
    response, err := client.RecognizePassport(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 读取护照图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)
```

 版权所有 © 华为云计算技术有限公司

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizePassportRequest{}
    countryCodePassportRequestBody:= "CHN"
    urlPassportRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.PassportRequestBody{
        CountryCode: &countryCodePassportRequestBody,
        Url: &urlPassportRequestBody,
    }
    response, err := client.RecognizePassport(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

版权所有 © 华为云计算技术有限公司

4.13 银行卡识别

功能介绍

识别银行卡上的关键文字信息，并以JSON格式返回识别的结构化结果。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-11 银行卡示例图



说明

如果图片中包含多张卡证票据，请调用**智能分类识别**服务。

约束与限制

- 只支持识别JPG、JPEG、PNG、BMP、TIFF格式图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 只支持识别银行卡正面，不支持识别背面。
- 只支持识别85.60×53.98 mm常规大小的银行卡，不支持迷你卡和形状不规则的异形卡。
- 支持图像中银行卡任意角度的水平旋转。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/bankcard

版权所有 © 华为云计算技术有限公司

表 4-120 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-121 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

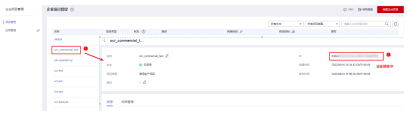
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-122 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
return_text_location	否	Boolean	识别到的文字块的区域位置信息。可选值包括： true：返回各个文字块区域 false：不返回各个文字块区域 如果无该参数，系统默认不返回文字块区域。如果输入参数不是Boolean类型，则会报非法参数错误。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-123 响应 Body 参数

参数	参数类型	描述
result	**BankcardResult** object	识别结果。 调用失败时不返回此字段。

版权所有 © 华为云计算技术有限公司

表 4-124 BankcardResult

参数	参数类型	描述
bank_name	String	发卡行。
card_number	String	银行卡号。
issue_date	String	有效期开始日期。
expiry_date	String	有效期截止日期。
type	String	银行卡类别，如：借记卡，信用卡，准贷记卡，预付费卡。 如果参数返回为空，表示未识别到银行卡类别。
confidence	Object	字段的置信度，取值范围0~1。 置信度越大，表示识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于字段的准确率。
text_location	Object	对应所有在原图上识别到的字段位置信息，包含所有文字区域四个顶点的二维坐标（x,y）。采用图像坐标系，坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。

状态码： 400

表 4-125 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，银行卡识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/bankcard”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

版权所有 © 华为云计算技术有限公司

- 传入银行卡图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/bankcard
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入银行卡图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/bankcard
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "url":"https://BucketName.obs.xxxx.com/ObjectName"
}
```

响应示例

状态码：200

成功响应示例

```
{
    "result": {
        "bank_name": "XX银行",
        "card_number": "XXXX XXXX XXXX XXXX",
        "issue_date": "12/10",
        "expiry_date": "12/20",
        "type": "",
        "confidence": {
            "bank_name": 0.9431,
            "card_number": 0.8562,
            "issue_date": 0.9962,
            "expiry_date": 0.9256,
            "type": 0
        }
    }
}
```

状态码：400

失败响应示例

```
{
    "error_code": "AIS.0103",
    "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入银行卡图片的base64编码进行文字识别

 版权所有 © 华为云计算技术有限公司

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeBankcardSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeBankcardRequest request = new RecognizeBankcardRequest();
        BankcardRequestBody body = new BankcardRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeBankcardResponse response = client.recognizeBankcard(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入银行卡图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeBankcardSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

```
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeBankcardRequest request = new RecognizeBankcardRequest();
        BankcardRequestBody body = new BankcardRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeBankcardResponse response = client.recognizeBankcard(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入银行卡图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeBankcardRequest()
        request.body = BankcardRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
```

 版权所有 © 华为云计算技术有限公司

```
        response = client.recognize_bankcard(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入银行卡图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeBankcardRequest()
        request.body = BankcardRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_bankcard(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入银行卡图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
```

 版权所有 © 华为云计算技术有限公司

```
    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeBankcardRequest{}
    imageBankcardRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.BankcardRequestBody{
        Image: &imageBankcardRequestBody,
    }
    response, err := client.RecognizeBankcard(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入银行卡图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeBankcardRequest{}
    urlBankcardRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.BankcardRequestBody{
        Url: &urlBankcardRequestBody,
    }
    response, err := client.RecognizeBankcard(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.14 营业执照识别

功能介绍

识别营业执照首页图片中的文字信息，并以JSON格式返回识别的结构化结果。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-12 营业执照示例图



图 4-13 营业执照示例图-横版



版权所有 © 华为云计算技术有限公司

约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片及PDF。
- 图像各边的像素在15px到8192px之间。单个图片、PDF文件其对应的Base64编码不超过10MB。
- 图像中营业执照区域有效占比超过70%，保证整张营业执照及其边缘包含在图像内。
- 支持图像中营业执照旋转、支持少量扭曲。
- 能处理暗光等干扰的图片但影响识别精度。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/business-license

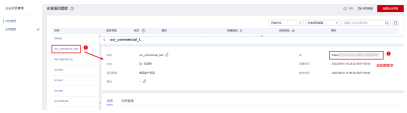
表 4-126 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

版权所有 © 华为云计算技术有限公司

请求参数

表 4-127 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-128 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 单个图片、PDF文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF、PDF格式。多页PDF仅识别第一页。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	该参数与image二选一。 单个图片、PDF文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

版权所有 © 华为云计算技术有限公司

表 4-129 响应 Body 参数

参数	参数类型	描述
result	**BusinessLicenseResult** object	识别结果。 调用失败时不返回此字段。

表 4-130 BusinessLicenseResult

参数	参数类型	描述
registration_number	String	• 老版本营业执照对应注册号。 • 新三证合一版本营业执照对应社会保障号。
name	String	企业名称。
type	String	公司/企业类型/主体类型/类型。
address	String	住所/营业场所/企业住所/主要经营场所/经营场所。
legal_representative	String	法定代表人/负责人/执行事务合伙人/投资人/经营者。
registered_capital	String	注册资本/出资额。
organization_form	String	组成形式。
found_date	String	成立日期/注册日期。
business_term	String	营业期限。 **说明** 若无截止日期，默认返回长期。
business_scope	String	经营范围。
issue_date	String	发照日期。
confidence	Object	字段的置信度，取值范围0~1。 置信度越大，本次识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于字段的准确率。

状态码： 400

版权所有 © 华为云计算技术有限公司

表 4-131 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，营业执照识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/business-license”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 传入营业执照图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/business-license
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入营业执照图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/business-license
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "url":"https://BucketName.obs.xxxx.com/ObjectName"
}
```

响应示例

状态码：200

成功响应示例

```
{
  "result" : {
    "registration_number" : "1234567890A",
    "name" : "ABC技术有限公司",
    "type" : "有限责任公司",
    "address" : "广东省深圳市珞喻路999号",
    "legal_representative" : "张三",
    "registered_capital" : "人民币壹佰万元整",
    "found_date" : "2011年10月18日",
    "organization_form" : "",
    "business_term" : "2011年10月18日至2021年10月17日",
```


版权所有 © 华为云计算技术有限公司

```
    "business_scope" : "计算机软硬件开发；电脑图文设计(依法须经批准的项目，经相关部门批准后方可开展经营活动)。",
    "issue_date" : "2011年10月18日",
    "confidence" : {
      "registration_number" : "0.9562",
      "name" : "0.9999",
      "type" : "0.9631",
      "address" : "0.9996",
      "legal_representative" : "0.9256",
      "registered_capital" : "0.9862",
      "found_date" : "0.9796",
      "organization_form" : 0.0,
      "business_term" : "0.9636",
      "business_scope" : "0.9584",
      "issue_date" : "0.9344"
    }
  }
}
```

状态码：400

失败响应示例

```
{
    "error_code": "AIS.0103",
    "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入营业执照图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeBusinessLicenseSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);
```

 版权所有 © 华为云计算技术有限公司

```
        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeBusinessLicenseRequest request = new RecognizeBusinessLicenseRequest();
        BusinessLicenseRequestBody body = new BusinessLicenseRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeBusinessLicenseResponse response = client.recognizeBusinessLicense(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入营业执照图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeBusinessLicenseSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeBusinessLicenseRequest request = new RecognizeBusinessLicenseRequest();
        BusinessLicenseRequestBody body = new BusinessLicenseRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeBusinessLicenseResponse response = client.recognizeBusinessLicense(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入营业执照图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeBusinessLicenseRequest()
        request.body = BusinessLicenseRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_business_license(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入营业执照图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \
```

 版权所有 © 华为云计算技术有限公司

```
    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeBusinessLicenseRequest()
        request.body = BusinessLicenseRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_business_license(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入营业执照图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeBusinessLicenseRequest{}
    imageBusinessLicenseRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.BusinessLicenseRequestBody{
        Image: &imageBusinessLicenseRequestBody,
    }
    response, err := client.RecognizeBusinessLicense(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入营业执照图片的url进行文字识别

```
package main
```

 版权所有 © 华为云计算技术有限公司

```
import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeBusinessLicenseRequest{}
    urlBusinessLicenseRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.BusinessLicenseRequestBody{
        Url: &urlBusinessLicenseRequestBody,
    }
    response, err := client.RecognizeBusinessLicense(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

 版权所有 © 华为云计算技术有限公司

4.15 道路运输证识别

功能介绍

识别道路运输证首页中的文字信息，并以JSON格式返回识别的结构化结果。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-14 道路运输证示例图







说明

如果图片中包含多张卡证票据，请调用**智能分类识别**服务。

约束与限制

- 只支持中国大陆道路运输证的识别。
- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 支持图像中道路运输证任意角度的水平旋转。
- 能处理反光、暗光、防伪标识等干扰的图片但影响识别精度。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

版权所有 © 华为云计算技术有限公司

URI

POST /v2/{project_id}/ocr/transportation-license

表 4-132 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-133 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

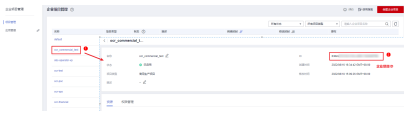
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-134 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
return_image_location	否	Boolean	返回道路运输证在原图中的坐标位置的开关，默认false，取值范围包括： • true：开启返回坐标位置的功能 • false：关闭返回坐标位置的功能
return_adjusted_image	否	Boolean	返回道路运输证（base64）的开关，默认false，取值范围包括： • true：开启道路运输证（base64）的功能 • false：关闭道路运输证（base64）的功能

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（status code）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-135 响应 Body 参数

参数	参数类型	描述
result	**TransportationLicenseResult** object	识别结果。 调用失败时不返回此字段。 Result中包含：6个主要字段和主字段外的其余可提取字段。

版权所有 © 华为云计算技术有限公司

表 4-136 TransportationLicenseResult

参数	参数类型	描述
owner_name	String	业户名称。
license_number	String	道路运输证号。
vehicle_number	String	车辆号牌。
vehicle_type	String	车辆类型。
maximum_capacity	String	吨(座)位。
vehicle_size	String	车辆尺寸。
issuing_authority	String	核发机关。
issue_date	String	发证日期。
owner_address	String	业户地址。
economic_type	String	经济类型。
business_certificate	String	经营许可证号。
business_scope	String	经营范围。
expiry_date	String	有效期。
review_expiry_date	String	审验有效期。
assessed_technical_level	String	技术等级评定。
image_location	Array<Array<Integer>>	道路运输证在原图中的坐标位置，输出左上、右上、右下、左下四个点坐标。仅在return_image_location设置为true时返回该字段。
adjusted_image	String	道路运输证图片的base64编码结果。仅在return_adjusted_image设置为true时返回该字段。
confidence	Object	字段的置信度，取值范围0~1。 置信度越大，本次识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于字段的准确率。 **说明** • 置信度中的相关字段均与返回值中的相关字段一一对应。

状态码： 400

版权所有 © 华为云计算技术有限公司

表 4-137 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。
  例如，道路运输证识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/transportation-license”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 传入道路运输证图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/transportation-license

Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入道路运输证图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/transportation-license

Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "url":"https://BucketName.obs.xxxx.com/ObjectName"
}
```

响应示例

状态码：200

成功响应示例

```
{
  "result" : {
    "owner_name" : "临猗县晋通货运有限公司",
    "license_number" : "晋交运管货字00000000号",
    "vehicle_number" : "晋M000000（黄）",
    "vehicle_type" : "重型半挂牵引车-解放牌CA000000000000",
    "maximum_capacity" : "0吨",
    "vehicle_size" : "6475*2000*3000毫米",
```


版权所有 © 华为云计算技术有限公司

```
  "issuing_authority" : "临猗县道路运输管理所",
  "issue_date" : "2014-11-16",
  "expiry_date" : "2024年08月xx日",
  "review_expiry_date" : "20xx年08月",
  "assessed_technical_level" : "二级 20xx年08月",
  "confidence" : {
    "owner_name" : 0.9963,
    "license_number" : 0.9988,
    "vehicle_number" : 0.9999,
    "vehicle_type" : 0.918,
    "maximum_capacity" : 0.9793,
    "vehicle_size" : 0.9671,
    "issuing_authority" : 0.9983,
    "issue_date" : 0.8969,
    "expiry_date" : 0.8526,
    "review_expiry_date" : 0.9112,
    "assessed_technical_level" : 0.8959
  }
 }
}
```

状态码：400

失败响应示例

```
{
   "error_code": "AIS.0103",
   "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入道路运输证图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeTransportationLicenseSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
```

 版权所有 © 华为云计算技术有限公司

```
            .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeTransportationLicenseRequest request = new RecognizeTransportationLicenseRequest();
        TransportationLicenseRequestBody body = new TransportationLicenseRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeTransportationLicenseResponse response =
client.recognizeTransportationLicense(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入道路运输证图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeTransportationLicenseSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeTransportationLicenseRequest request = new RecognizeTransportationLicenseRequest();
        TransportationLicenseRequestBody body = new TransportationLicenseRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeTransportationLicenseResponse response =
client.recognizeTransportationLicense(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入道路运输证图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeTransportationLicenseRequest()
        request.body = TransportationLicenseRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_transportation_license(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入道路运输证图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
```

 版权所有 © 华为云计算技术有限公司

```
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeTransportationLicenseRequest()
        request.body = TransportationLicenseRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_transportation_license(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入道路运输证图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeTransportationLicenseRequest{}
    imageTransportationLicenseRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.TransportationLicenseRequestBody{
        Image: &imageTransportationLicenseRequestBody,
    }
    response, err := client.RecognizeTransportationLicense(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

- 传入道路运输证图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeTransportationLicenseRequest{}
    urlTransportationLicenseRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.TransportationLicenseRequestBody{
        Url: &urlTransportationLicenseRequestBody,
    }
    response, err := client.RecognizeTransportationLicense(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

版权所有 © 华为云计算技术有限公司

错误码

错误码请参见**错误码**。

4.16 不动产证识别

功能介绍

识别不动产证中的文字信息，并返回识别的结构化结果。

说明

如果图片中包含多张卡证票据，请调用**智能分类识别**服务。

约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式的图片。
- 图像各边的像素大小在15到8192px之间。单个图片对应的Base64编码不超过10MB。
- 支持图像中不动产证任意角度的水平旋转。
- 能处理反光、暗光等干扰的图片但影响识别精度。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/real-estate-certificate

表 4-138 路径参数

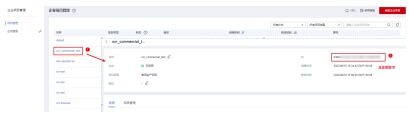
参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。

版权所有 © 华为云计算技术有限公司

参数	是否必选	说明
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-139 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-140 请求 Body 参数

参数	是否必选	参数类型	描述
image	否	String	与url二选一。 图像数据，base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片最小边不小于15像素，最长边不超过8192像素。支持JPG/PNG/BMP/TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	与image二选一。 url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的URL路径目前仅支持华为云上OBS提供的匿名公开授权访问的URL以及公网URL。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。

响应参数

状态码： 200

表 4-141 响应 Body 参数

参数	参数类型	描述
result	**RealEstateCertificateResult** object	调用成功时表示调用结果。 调用失败时无此字段。

版权所有 © 华为云计算技术有限公司

表 4-142 RealEstateCertificateResult

参数	参数类型	描述
issuer	String	填发单位。
issue_date	String	填发日期。
real_estate_certificate_no	String	不动产证编号。
mortgage_seals	Integer	抵押印章个数。
canceled_mortgage_seals	Integer	注销的抵押印章个数。
estate_location	String	房屋坐落。
total_floors	String	总楼层数。
floor	String	所在层。
year_built	String	建成年份。
structure	String	结构。
area	String	建筑面积。
revenue_stamps	Integer	印花税票个数。
ownership_certificate_no	String	产权证号。
estate_holder	String	房屋所有权人。
obligee	String	权利人。
ownership	String	共有情况。
property_unit_no	String	不动产单元号。
right_type	String	权利类型。
right_nature	String	权利性质。
usage	String	使用用途。
intended_usage	String	设计、规划用途。
confidence	Object	各个字段的置信度。

状态码： 400

版权所有 © 华为云计算技术有限公司

表 4-143 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

请求示例

说明

- Endpoint即调用API的请求地址，不同服务不同区域的Endpoint不同，具体请参见**终端节点**。 例如，不动产证识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/real-estate-certificate”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入不动产证图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/real-estate-certificate

{
  "image" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入不动产证图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/real-estate-certificate

{
  "url" : "https://BucketName.obs.myhuaweicloud.com/ObjectName"
}
```

响应示例

状态码： 200

成功响应示例

```
{
  "result" : {
    "estate_holder" : "xx",
    "ownership_certificate_no" : "黔 （ 2019 ） 白云区 不动产权第xxx号",
    "obligee" : "抵押预告登记",
    "ownership" : "xxx",
    "estate_location" : "xxx银行股份有限公司xxx分行",
    "right_type" : "x",
    "right_nature" : "x",
    "usage" : "x",
    "intended_usage" : "x",
    "area" : "白云区云峰大xx号B、C栋（ B ）x单元x层x号",
    "total_floors" : "x",
    "floor" : "x",
    "issuer" : "xxx规划和国土资源管理委员会",
    "issue_date" : "2019-06-2x",
    "real_estate_certificate_no" : "xx201016xxx",
    "mortgage_seals" : 0,
    "canceled_mortgage_seals" : 0,
    "year_built" : "x",
```


版权所有 © 华为云计算技术有限公司

```
    "structure" : "x",
    "revenue_stamps" : 0,
    "property_unit_no" : "x",
    "confidence" : {
      "estate_holder" : 0,
      "ownership_certificate_no" : 0.8931,
      "obligee" : 0.8879,
      "ownership" : 0.2806,
      "estate_location" : 0.8145,
      "right_type" : 0,
      "right_nature" : 0,
      "usage" : 0,
      "intended_usage" : 0,
      "area" : 0.6588,
      "total_floors" : 0,
      "floor" : 0,
      "issuer" : 0.974,
      "issue_date" : 0.9198,
      "real_estate_certificate_no" : 0.9475,
      "mortgage_seals" : 0,
      "canceled_mortgage_seals" : 0,
      "year_built" : 0,
      "structure" : 0,
      "revenue_stamps" : 0,
      "property_unit_no" : 0
    }
  }
}
```

状态码： 400

失败响应示例

```
{
  "result" : {
    "error_code" : "AIS.0103",
    "error_msg" : "The image size does not meet the requirements."
  }
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入不动产证图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeRealEstateCertificateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

```
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeRealEstateCertificateRequest request = new RecognizeRealEstateCertificateRequest();
        RealEstateCertificateRequestBody body = new RealEstateCertificateRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeRealEstateCertificateResponse response = client.recognizeRealEstateCertificate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入不动产证图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeRealEstateCertificateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
```

 版权所有 © 华为云计算技术有限公司

```
                .build();
        RecognizeRealEstateCertificateRequest request = new RecognizeRealEstateCertificateRequest();
        RealEstateCertificateRequestBody body = new RealEstateCertificateRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeRealEstateCertificateResponse response =
client.recognizeRealEstateCertificate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入不动产证图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeRealEstateCertificateRequest()
        request.body = RealEstateCertificateRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_real_estate_certificate(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入不动产证图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
```

 版权所有 © 华为云计算技术有限公司

```
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeRealEstateCertificateRequest()
        request.body = RealEstateCertificateRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_real_estate_certificate(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入不动产证图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeRealEstateCertificateRequest{}
    imageRealEstateCertificateRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
```

 版权所有 © 华为云计算技术有限公司

```
    request.Body = &model.RealEstateCertificateRequestBody{
        Image: &imageRealEstateCertificateRequestBody,
    }
    response, err := client.RecognizeRealEstateCertificate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入不动产证图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeRealEstateCertificateRequest{}
    urlRealEstateCertificateRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.RealEstateCertificateRequestBody{
        Url: &urlRealEstateCertificateRequestBody,
    }
    response, err := client.RecognizeRealEstateCertificate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例

 版权所有 © 华为云计算技术有限公司

状态码	描述
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.17 车辆合格证识别

功能介绍

识别车辆合格证中的文字信息，并返回识别的结构化结果。

约束与限制

- 只支持中国大陆车辆合格证的识别。
- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式的图片。
- 图像各边的像素大小在15到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中识别区域有效占比超过80%，保证整张车辆合格证内容及其边缘包含在图像内。
- 支持图像中车辆合格证任意角度的水平旋转。
- 支持少量扭曲，扭曲后图像中的车辆合格证长宽比与实际驾驶证相差不超过10%。
- 能处理暗光等干扰的图片但影响识别精度。
- 文字识别服务属于公有云服务，线上用户资源共享，如果需要多并发请求，请提前**联系我们**。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

版权所有 © 华为云计算技术有限公司

URI

POST /v2/{project_id}/ocr/vehicle-certificate

表 4-144 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-145 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。

版权所有 © 华为云计算技术有限公司

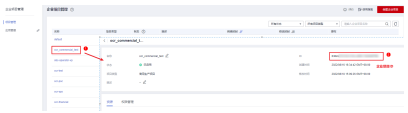
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-146 请求 Body 参数

参数	是否必选	参数类型	描述
image	否	String	与url二选一。 图片base64编码，url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最小边不小于15像素，最长边不超过8192像素。支持JPG/PNG/BMP/TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
url	否	String	与image二选一。 url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的URL路径目前仅支持华为云上OBS提供的匿名公开授权访问的URL以及公网URL。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。
return_text_location	否	Boolean	识别到的文字块的区域位置信息。可选值包括： • true：返回各个文字块区域 • false：不返回各个文字块区域 如果无该参数，系统默认不返回文字块区域。如果输入参数不是Boolean类型，则会报非法参数错误。

响应参数

状态码： 200

表 4-147 响应 Body 参数

参数	参数类型	描述
result	**VehicleCertificateResult** object	调用成功时表示调用结果。 调用失败时无此字段。

表 4-148 VehicleCertificateResult

参数	参数类型	描述
certificate_number	String	合格证编号。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
issue_date	String	发证日期。
manufacture_name	String	车辆制造企业名称。
vehicle_brand	String	车辆品牌。
vehicle_name	String	车辆名称。
vehicle_model	String	车辆型号。
vin	String	车架号。
vehicle_color	String	车身颜色。
chassis_model	String	底盘型号。
chassis_id	String	底盘ID。
chassis_certificate_number	String	底盘合格证编号。
engine_model	String	发动机型号。
engine_number	String	发动机号。
fuel_type	String	燃料种类。
displacement	String	排量。
power	String	功率。
emission_standard	String	排放标准。
fuel_consumption	String	油耗。
overall_dimension_length	String	外廓尺寸-长。
overall_dimension_width	String	外廓尺寸-宽。
overall_dimension_height	String	外廓尺寸-高。
container_dimension_length	String	货厢内部尺寸-长。
container_dimension_width	String	货厢内部尺寸-宽。
container_dimension_height	String	货厢内部尺寸-高。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
spring_quantity	String	钢板弹簧片数。
tire_quantity	String	轮胎数。
tire_size	String	轮胎规格。
front_wheel_track	String	轮距-前。
rear_wheel_track	String	轮距-后。
wheelbase	String	轴距。
axle_load	String	轴荷。
axle_quantity	String	轴数。
steering_form	String	转向形式。
total_weight	String	总质量。
equipment_weight	String	整备质量。
maximum_laden_mass	String	额定载质量。
mass_utilization_coefficient	String	载质量利用系数。
traction_weight	String	准牵引总质量。
maximum_load_mass	String	半挂车鞍座最大允许总质量。
cab_passenger_capacity	String	驾驶室准乘人数。
passenger_capacity	String	额定载客。
max_design_speed	String	最高设计车速。
manufacture_date	String	车辆制造日期。
confidence	Object	字段的置信度，取值范围0~1。 置信度越大，本次识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于字段的准确率。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
text_location	Object	对应所有在原图上识别到的字段位置信息，包含所有文字区域四个顶点的二维坐标（x,y）。采用图像坐标系，坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。

状态码： 400

表 4-149 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

请求示例

说明

- Endpoint即调用API的请求地址，不同服务不同区域的Endpoint不同，具体请参见**终端节点**。 例如，车辆合格证识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/vehicle-certificate”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入车辆合格证图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/vehicle-certificate

{
  "image" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入车辆合格证图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/vehicle-certificate

{
  "url" : "https://BucketName.obs.myhuaweicloud.com/ObjectName"
}
```

响应示例

状态码： 200

成功响应示例

```
{
  "result" : {
    "certificate_number" : "WAS1xxxx2399503",
    "issue_date" : "2022年10月23日",
    "manufacture_name" : "xxxx汽车有限公司",
```


版权所有 © 华为云计算技术有限公司

```
"vehicle_brand" : "xxx牌/xxx",
"vehicle_name" : "轿车",
"vehicle_model" : "TV7xxxxx",
"vin" : "LFMAYxxxxxxx92063",
"vehicle_color" : "白",
"chassis_model" : "x",
"chassis_id" : "x",
"chassis_certificate_number" : "x",
"engine_model" : "9NR",
"engine_number" : "A2853xx",
"fuel_type" : "汽油",
"displacement" : "x197",
"power" : "xx",
"emission_standard" : "GBXXXXX-XXXX",
"fuel_consumption" : "xx",
"overall_dimension_length" : "xx35",
"overall_dimension_width" : "17xx",
"overall_dimension_height" : "14xx",
"container_dimension_length" : "x",
"container_dimension_width" : "x",
"container_dimension_height" : "x",
"spring_quantity" : "xx",
"tire_quantity" : "x",
"tire_size" : "XXXXX.X",
"front_wheel_track" : "xx27",
"rear_wheel_track" : "xx26",
"wheelbase" : "27xx",
"axle_load" : "xxx/830",
"axle_quantity" : "x",
"steering_form" : "方向盘",
"total_weight" : "xx70",
"equipment_weight" : "xx40",
"maximum_laden_mass" : "x",
"mass_utilization_coefficient" : "x",
"traction_weight" : "x",
"maximum_load_mass" : "x",
"cab_passenger_capacity" : "x",
"passenger_capacity" : "x",
"max_design_speed" : "xxx",
"manufacture_date" : "2022年10月xxx日",
"confidence" : {
  "certificate_number" : 0.9856,
  "issue_date" : 0.9931,
  "manufacture_name" : 0.9485,
  "vehicle_brand" : 0.9612,
  "vehicle_name" : 0.9562,
  "vehicle_model" : 0.9741,
  "vin" : 0.9963,
  "vehicle_color" : 0.9645,
  "chassis_model" : 0,
  "chassis_id" : 0,
  "chassis_certificate_number" : 0,
  "engine_model" : 0.9745,
  "engine_number" : 0.9384,
  "fuel_type" : 0.9845,
  "displacement" : 0.9954,
  "power" : 0.9861,
  "emission_standard" : 0.9864,
  "fuel_consumption" : 0.9952,
  "overall_dimension_length" : 0.9995,
  "overall_dimension_width" : 0.9996,
  "overall_dimension_height" : 0.9982,
  "container_dimension_length" : 0,
  "container_dimension_width" : 0,
  "container_dimension_height" : 0,
  "spring_quantity" : 0,
  "tire_quantity" : 0.9984,
  "tire_size" : 0.9631,
  "front_wheel_track" : 0.9815,
```


版权所有 © 华为云计算技术有限公司

```
    "rear_wheel_track" : 0.9951,
    "wheelbase" : 0.9762,
    "axle_load" : 0.9978,
    "axle_quantity" : 0.9982,
    "steering_form" : 0.9946,
    "total_weight" : 0.9863,
    "equipment_weight" : 0.9975,
    "maximum_laden_mass" : 0,
    "mass_utilization_coefficient" : 0,
    "traction_weight" : 0,
    "maximum_load_mass" : 0,
    "cab_passenger_capacity" : 0,
    "passenger_capacity" : 0.9992,
    "max_design_speed" : 0.9999,
    "manufacture_date" : 0.9856
   }
  }
}
```

状态码： 400

失败响应示例

```
{
  "error_code" : "AIS.0103",
  "error_msg" : "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入车辆合格证图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeVehicleCertificateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);
```

 版权所有 © 华为云计算技术有限公司

```
        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeVehicleCertificateRequest request = new RecognizeVehicleCertificateRequest();
        VehicleCertificateRequestBody body = new VehicleCertificateRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeVehicleCertificateResponse response = client.recognizeVehicleCertificate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入车辆合格证图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeVehicleCertificateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeVehicleCertificateRequest request = new RecognizeVehicleCertificateRequest();
        VehicleCertificateRequestBody body = new VehicleCertificateRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeVehicleCertificateResponse response = client.recognizeVehicleCertificate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入车辆合格证图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeVehicleCertificateRequest()
        request.body = VehicleCertificateRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_vehicle_certificate(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入车辆合格证图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \
```

 版权所有 © 华为云计算技术有限公司

```
    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeVehicleCertificateRequest()
        request.body = VehicleCertificateRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_vehicle_certificate(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入车辆合格证图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeVehicleCertificateRequest{}
    imageVehicleCertificateRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.VehicleCertificateRequestBody{
        Image: &imageVehicleCertificateRequestBody,
    }
    response, err := client.RecognizeVehicleCertificate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入车辆合格证图片的url进行文字识别

```
package main
```

 版权所有 © 华为云计算技术有限公司

```
import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeVehicleCertificateRequest{}
    urlVehicleCertificateRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.VehicleCertificateRequestBody{
        Url: &urlVehicleCertificateRequestBody,
    }
    response, err := client.RecognizeVehicleCertificate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

 版权所有 © 华为云计算技术有限公司

4.18 道路运输从业资格证识别

功能介绍

识别道路运输从业资格证上的关键文字信息，并返回识别的结构化结果。

图 4-15 道路运输从业资格证示例图



约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中识别区域有效占比超过80%，保证整张表格及其边缘包含在图像内。
- 支持图像任意角度的水平旋转。
- 目前不支持复杂背景（如户外自然场景、防伪水印等）和表格线扭曲图像的文字识别。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

版权所有 © 华为云计算技术有限公司

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/transportation-qualification-certificate

表 4-150 路径参数

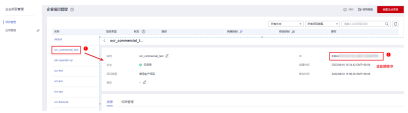
参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-151 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** **该功能仅支持开通在华东-上海一区域的API。** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-152 请求 Body 参数

参数	是否必选	参数类型	描述
image	否	String	与url二选一 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	与image二选一 url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。

响应参数

状态码： 200

版权所有 © 华为云计算技术有限公司

表 4-153 响应 Body 参数

参数	参数类型	描述
result	QualificationCertificateResult object	识别结果。 调用失败时不返回此字段。

表 4-154 QualificationCertificateResult

参数	参数类型	描述
id_number	String	身份证号（非必有，依赖对应从业资格证板式）。
assessment_date	String	考核时间（非必有，依赖对应从业资格证板式）。
certificate_number	String	从业资格证号。
file_number	String	档案号（非必有，依赖对应从业资格证板式）。
union_card_number	String	福路通号（非必有，依赖对应从业资格证板式）。
continuing_education_info	String	继续教育信息（非必有，依赖对应从业资格证板式）。
sex	String	性别（非必有，依赖对应从业资格证板式）。
phone_number	String	联系电话（非必有，依赖对应从业资格证板式）。
registration_date	String	登记时间（非必有，依赖对应从业资格证板式）。
work_unit	String	单位（非必有，依赖对应从业资格证板式）。
integrity_assessment_info	String	诚信考核信息（非必有，依赖对应从业资格证板式）。
nationality	String	国籍（非必有，依赖对应从业资格证板式）。
name	String	姓名。
address	String	住址。
driving_class	String	准驾车型（非必有，依赖对应从业资格证板式）。
issuing_authority	String	发证机关（非必有，依赖对应从业资格证板式）。
birth_date	String	出生日期（非必有，依赖对应从业资格证板式）。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
qualification_category_list	Array of **Qualification Category** objects	从业资格列表。
confidence	**Qualification Confidence** object	各个字段的置信度。

表 4-155 QualificationCategory

参数	参数类型	描述
category	String	从业资格类别（非必有，依赖对应从业资格证板式）。
initial_issue_date	String	初次领证日期（非必有，依赖对应从业资格证板式）。
issue_date	String	有效起始日期（非必有，依赖对应从业资格证板式）。
expiry_date	String	有效截止日期（非必有，依赖对应从业资格证板式）。

表 4-156 QualificationConfidence

参数	参数类型	描述
id_number	Float	身份证号置信度。
assessment_date	Float	考核时间置信度。
certificate_number	Float	从业资格证号置信度。
file_number	Float	档案号置信度。
union_card_number	Float	福路通号置信度。
continuing_education_info	Float	继续教育信息置信度。
sex	Float	性别置信度。
phone_number	Float	联系电话置信度。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
registration_date	Float	登记时间置信度。
work_unit	Float	单位置信度。
integrity_assessment_info	Float	诚信考核信息置信度。
nationality	Float	国籍置信度。
name	Float	姓名置信度。
address	Float	住址置信度。
driving_class	Float	准驾车型置信度。
issuing_authority	Float	发证机关置信度。
birth_date	Float	出生日期置信度。
qualification_category_list	Array of **QualificationCategoryConfidence** objects	从业资格列表置信度。

表 4-157 QualificationCategoryConfidence

参数	参数类型	描述
category	Float	从业资格类别（非必有，依赖对应从业资格证板式）。
initial_issue_date	Float	初次领证日期（非必有，依赖对应从业资格证板式）。
issue_date	Float	有效起始日期（非必有，依赖对应从业资格证板式）。
expiry_date	Float	有效截止日期（非必有，依赖对应从业资格证板式）。

状态码： 400

版权所有 © 华为云计算技术有限公司

表 4-158 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- Endpoint即调用API的请求地址，不同服务不同区域的Endpoint不同，具体请参见**终端节点**。 例如，道路运输从业资格证识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/transportation-qualification-certificate”，“project_id”为项目ID，获取方法请参见**获取项目ID获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入道路运输从业资格证图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/transportation-qualification-certificate
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
  "image" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入道路运输从业资格证图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/transportation-qualification-certificate
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
  "url" : "https://BucketName.obs.myhuaweicloud.com/ObjectName"
}
```

响应示例

状态码： 200

成功响应示例

```
{
  "result" : {
    "id_number" : "140xxxxxxxxxxxxxx",
    "assessment_date" : "11月11日",
    "certificate_number" : "22038119xxxxxxxxxx",
    "file_number" : "3704020030xxxxxxxxxxxxxxxx",
    "union_card_number" : "3105060xxxxxxxxxxxx",
    "continuing_education_info" : "2020年11月-2022年11月",
    "sex" : "男",
    "phone_number" : "139xxxxxxxx",
    "registration_date" : "2016年07月13日",
```


版权所有 © 华为云计算技术有限公司

```
  "work_unit" : "无",
  "integrity_assessment_info" : "AAA 2019年11月-2020年11月",
  "nationality" : "中国",
  "name" : "xxx",
  "address" : "吉林省xxxx",
  "driving_class" : "A2",
  "issuing_authority" : "长春市交通运输局",
  "birth_date" : "1979/12/22",
  "qualification_category_list" : [ {
    "category" : "经营性道路货物运输驾驶员",
    "initial_issue_date" : "2006年04月03日",
    "issue_date" : "2021年06月22日",
    "expiry_date" : "2027年06月22日"
  } ],
  "confidence" : {
    "id_number" : 0.9974,
    "assessment_date" : 0.9917,
    "certificate_number" : 0.9957,
    "file_number" : 0.9873,
    "union_card_number" : 0.9973,
    "continuing_education_info" : 0.9795,
    "sex" : 0.9863,
    "phone_number" : 0.8987,
    "registration_date" : 0.9806,
    "work_unit" : 0.8285,
    "integrity_assessment_info" : 0.9983,
    "nationality" : 0.995,
    "name" : 0.9944,
    "address" : 0.9565,
    "driving_class" : 0.5711,
    "issuing_authority" : 0.9234,
    "birth_date" : 0.9821,
    "qualification_category_list" : [ {
      "category" : 0.9979,
      "initial_issue_date" : 0.9883,
      "issue_date" : 0.9964,
      "expiry_date" : 0.9988
    } ]
  }
 }
}
```


状态码： 400

失败响应示例


```
{
  "error_code" : "AIS.0103",
  "error_msg" : "The image size does not meet the requirements"
}
```


SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入道路运输从业资格证图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
```

```
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeQualificationCertificateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeQualificationCertificateRequest request = new RecognizeQualificationCertificateRequest();
        QualificationCertificateRequestBody body = new QualificationCertificateRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeQualificationCertificateResponse response = client.recognizeQualificationCertificate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入道路运输从业资格证图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeQualificationCertificateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
```

```
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeQualificationCertificateRequest request = new
RecognizeQualificationCertificateRequest();
        QualificationCertificateRequestBody body = new QualificationCertificateRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeQualificationCertificateResponse response =
client.recognizeQualificationCertificate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入道路运输从业资格证图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeQualificationCertificateRequest()
        request.body = QualificationCertificateRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_qualification_certificate(request)
```

 版权所有 © 华为云计算技术有限公司

```
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入道路运输从业资格证图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeQualificationCertificateRequest()
        request.body = QualificationCertificateRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_qualification_certificate(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入道路运输从业资格证图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
```

 版权所有 © 华为云计算技术有限公司

```
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeQualificationCertificateRequest{}
    imageQualificationCertificateRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/
4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.QualificationCertificateRequestBody{
        Image: &imageQualificationCertificateRequestBody,
    }
    response, err := client.RecognizeQualificationCertificate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入道路运输从业资格证图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeQualificationCertificateRequest{}
    urlQualificationCertificateRequestBody:= "https://BucketName.obs.myhuaweicloud.com/
ObjectName"
    request.Body = &model.QualificationCertificateRequestBody{
        Url: &urlQualificationCertificateRequestBody,
    }
    response, err := client.RecognizeQualificationCertificate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.19 车牌识别

功能介绍

识别输入图片中的车牌信息，并以JSON格式返回其坐标和内容。

该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-16 车牌示例图



说明

支持车牌信息、车牌颜色识别，支持双行车牌识别，支持单张图片内多个车牌识别。

目前支持车牌类型含**小型汽车、小型新能源、大型新能源、使馆汽车、领馆汽车、中国-港澳入出境汽车、教练汽车、警用汽车号牌**。

不支持临牌。

约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。

版权所有 © 华为云计算技术有限公司

- 图像中车牌区域有效占比超过10%，保证所有文字及其边缘包含在图像内。
- 推荐图像中车牌清晰无遮挡、正面放置无倾斜。
- 目前支持车牌类型含小型汽车、小型新能源、大型新能源、使馆汽车、领馆汽车、中国-港澳入出境汽车、教练汽车、警用汽车号牌，支持双行号牌。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/license-plate

表 4-159 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-160 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

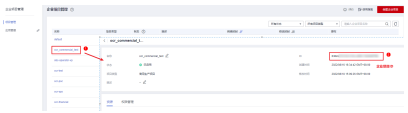
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-161 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（status code）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-162 响应 Body 参数

参数	参数类型	描述
result	Array of **LicensePlateResult** objects	识别结果。 调用失败时不返回此字段。

表 4-163 LicensePlateResult

参数	参数类型	描述
plate_number	String	车牌内容。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
plate_color	String	当前版本支持的车牌底色类型： • blue：蓝色 • green：绿色（小型新能源车牌） • black：黑色 • white：白色 • yellow：黄色 • yellow_green: 黄绿（大型新能源车牌）
plate_location	Array of integers	车牌的区域位置信息，列表形式，包含文字区域四个顶点的二维坐标（x,y）；采用图像坐标系，坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。
confidence	Float	字段的置信度，取值范围0~1。 置信度越大，本次识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于字段的准确率。

状态码： 400

表 4-164 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。
  例如，车牌识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/license-plate”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 传入车牌图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/license-plate
 Request Header:
 Content-Type: application/json
```


版权所有 © 华为云计算技术有限公司

```
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
   "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."}
```

- 传入车牌图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/license-plate
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...

Request Body:
{
    "url":"https://BucketName.obs.xxxx.com/ObjectName"}
```

响应示例

状态码：200

成功响应示例

```
{

  "result": [
     {
        "plate_number": "京XXXXX",
        "plate_color": "blue",
        "confidence": 0.9225,
        "plate_location": [
              [370,881],
              [2591,881],
              [2591,2281],
              [370,2281]
        ]
     }
  ]
}
```

状态码：400

失败响应示例

```
{
   "error_code": "AIS.0103",
   "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入车牌图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
```

 版权所有 © 华为云计算技术有限公司

```
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeLicensePlateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeLicensePlateRequest request = new RecognizeLicensePlateRequest();
        LicensePlateRequestBody body = new LicensePlateRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeLicensePlateResponse response = client.recognizeLicensePlate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入车牌图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeLicensePlateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
```

```
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeLicensePlateRequest request = new RecognizeLicensePlateRequest();
        LicensePlateRequestBody body = new LicensePlateRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeLicensePlateResponse response = client.recognizeLicensePlate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入车牌图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeLicensePlateRequest()
        request.body = LicensePlateRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_license_plate(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

 版权所有 © 华为云计算技术有限公司

```
        print(e.error_code)
        print(e.error_msg)
```

- 传入车牌图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeLicensePlateRequest()
        request.body = LicensePlateRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_license_plate(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入车牌图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()
```

 版权所有 © 华为云计算技术有限公司

```
    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeLicensePlateRequest{}
    imageLicensePlateRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/
4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.LicensePlateRequestBody{
        Image: &imageLicensePlateRequestBody,
    }
    response, err := client.RecognizeLicensePlate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入车牌图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeLicensePlateRequest{}
    urlLicensePlateRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.LicensePlateRequestBody{
        Url: &urlLicensePlateRequestBody,
    }
    response, err := client.RecognizeLicensePlate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.20 名片识别

功能介绍

识别名片图片上的文字信息，并以JSON格式返回识别的结构化结果。支持对多种不同版式名片进行结构化信息提取。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-17 名片示例图



约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中名片的有效占比超过60%，保证整张名片内容包含在图像内。
- 支持图像中名片的任意角度的水平旋转（需开启方向检测）。
- 能处理反光、暗光、防伪标识等干扰的图片但影响识别精度。

调用方法

请参见**如何调用API**。

版权所有 © 华为云计算技术有限公司

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/business-card

表 4-165 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-166 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

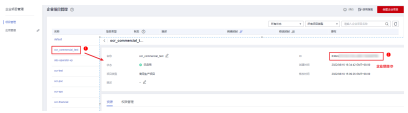
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-167 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
detect_direction	否	Boolean	是否校正图片的倾斜角度，可选值如下。 • true：校正图片的倾斜角度 • false：不校正图片的倾斜角度 支持任意角度的校正，未传入该参数时默认为“false”。 待识别图片如果存在倾斜，建议将此参数设置为“true”。
return_adjusted_image	否	Boolean	返回校正后的名片图像的Base64编码的开关，可选值如下所示。 • true：返回Base64编码 • false：不返回Base64编码 未传入该参数时默认为“false”。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

版权所有 © 华为云计算技术有限公司

表 4-168 响应 Body 参数

参数	参数类型	描述
result	**BusinessCardResult** object	识别结果。 调用失败时不返回此字段。

表 4-169 BusinessCardResult

参数	参数类型	描述
name	Array of strings	姓名列表。
title	Array of strings	职位头衔列表。
company	Array of strings	公司列表。
department	Array of strings	部门列表。
phone	Array of strings	联系方式列表。
address	Array of strings	地址列表。
email	Array of strings	邮箱列表。
fax	Array of strings	传真列表。
postcode	Array of strings	邮编列表。
website	Array of strings	公司网址列表。
extra_info_list	Array of **ExtraInfoList** objects	其余信息列表。
adjusted_image	String	返回矫正后的名片图像的BASE64编码。

表 4-170 ExtraInfoList

参数	参数类型	描述
item	String	表示“key”值，例如：bank等。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
value	String	表示“value”值，例如：bank等。
note	Array of strings	对应“item”关联的额外信息，为“bank”时第一个默认为户名，第二个为开户行。

状态码： 400

表 4-171 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，名片识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/business-card”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 传入名片图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/business-card

Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入名片图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/business-card

Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "url":"https://BucketName.obs.xxxx.com/ObjectName"
}
```


版权所有 © 华为云计算技术有限公司

响应示例

状态码：200

成功响应示例

```
{
   "result": {
      "name": ["XX"],
      "title": ["销售总监"],
      "company": ["XX有限公司"],
      "department": ["XX产品部"],
      "phone": ["+XX XXX XXXX XXXX","XXXX XXXXXXXXX"],
      "address": ["XXXX"],
      "email": ["XX"],
      "fax": ["XXXX XXXXXXXX"],
      "postcode": [],
      "website": ["XX"],
      "extra_info_list": [
         {
            "item": "bank",
            "value": "XXXXXXXXXXXXXXX",
            "note": ["张三","XX"]
         }
      ]
   }
}
```

状态码：400

失败响应示例

```
{
   "error_code": "AIS.0103",
   "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入名片图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeBusinessCardSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
```


版权所有 © 华为云计算技术有限公司

```
        // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeBusinessCardRequest request = new RecognizeBusinessCardRequest();
        BusinessCardRequestBody body = new BusinessCardRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeBusinessCardResponse response = client.recognizeBusinessCard(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入名片图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeBusinessCardSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeBusinessCardRequest request = new RecognizeBusinessCardRequest();
        BusinessCardRequestBody body = new BusinessCardRequestBody();
```

 版权所有 © 华为云计算技术有限公司

```
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeBusinessCardResponse response = client.recognizeBusinessCard(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入名片图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeBusinessCardRequest()
        request.body = BusinessCardRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_business_card(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入名片图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

```
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeBusinessCardRequest()
        request.body = BusinessCardRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_business_card(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入名片图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeBusinessCardRequest{}
    imageBusinessCardRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.BusinessCardRequestBody{
        Image: &imageBusinessCardRequestBody,
    }
    response, err := client.RecognizeBusinessCard(request)
```

 版权所有 © 华为云计算技术有限公司

```
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入名片图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeBusinessCardRequest{}
    urlBusinessCardRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.BusinessCardRequestBody{
        Url: &urlBusinessCardRequestBody,
    }
    response, err := client.RecognizeBusinessCard(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

版权所有 © 华为云计算技术有限公司

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.21 VIN 码识别

功能介绍

识别图片中的车架号信息，并将识别结果以JSON格式返回给用户。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-18 VIN 码示例图



约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 图像各边的像素大小在15px到4096px之间。单个图片对应的Base64编码不超过10MB。
- 能处理反光、暗光、防伪标识等干扰的图片但影响识别精度。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/vin

版权所有 © 华为云计算技术有限公司

表 4-172 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-173 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

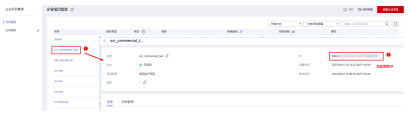
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-174 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最短边不小于15px，最长边不超过4096px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-175 响应 Body 参数

参数	参数类型	描述
result	**VINResult** object	识别结果。 调用失败时不返回此字段。

表 4-176 VINResult

参数	参数类型	描述
vin	String	识别检测到的车架号。

状态码： 400

版权所有 © 华为云计算技术有限公司

表 4-177 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，VIN码识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/vin”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 传入车架号VIN码图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/vin

Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
   "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入车架号VIN码图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/vin

Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
   "url":"https://BucketName.obs.xxxx.com/ObjectName"
}
```

响应示例

状态码：200

成功响应示例

```
{
   "result": {
      "vin": "JN8BYXXXXXXXX7207"
   }
}
```

状态码：400

失败响应示例

版权所有 © 华为云计算技术有限公司

```
{
   "error_code": "AIS.0103",
   "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入车架号VIN码图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeVinSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeVinRequest request = new RecognizeVinRequest();
        VinRequestBody body = new VinRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeVinResponse response = client.recognizeVin(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
```

 版权所有 © 华为云计算技术有限公司

```
    }
}
```

- 传入车架号VIN码图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeVinSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeVinRequest request = new RecognizeVinRequest();
        VinRequestBody body = new VinRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeVinResponse response = client.recognizeVin(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入车架号VIN码图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

 版权所有 © 华为云计算技术有限公司

```
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeVinRequest()
        request.body = VinRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_vin(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入车架号VIN码图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeVinRequest()
        request.body = VinRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_vin(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入车架号VIN码图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeVinRequest{}
    imageVinRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.VinRequestBody{
        Image: &imageVinRequestBody,
    }
    response, err := client.RecognizeVin(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入车架号VIN码图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()
```

 版权所有 © 华为云计算技术有限公司

```
    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeVinRequest{}
    urlVinRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.VinRequestBody{
        Url: &urlVinRequestBody,
    }
    response, err := client.RecognizeVin(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.22 增值税发票识别

功能介绍

识别增值税发票的类别，并以JSON格式返回识别的结构化结果，支持识别图片、PDF、OFD文件。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

发票验真API请参见**发票验真**。

版权所有 © 华为云计算技术有限公司

图 4-19 增值税发票示例图

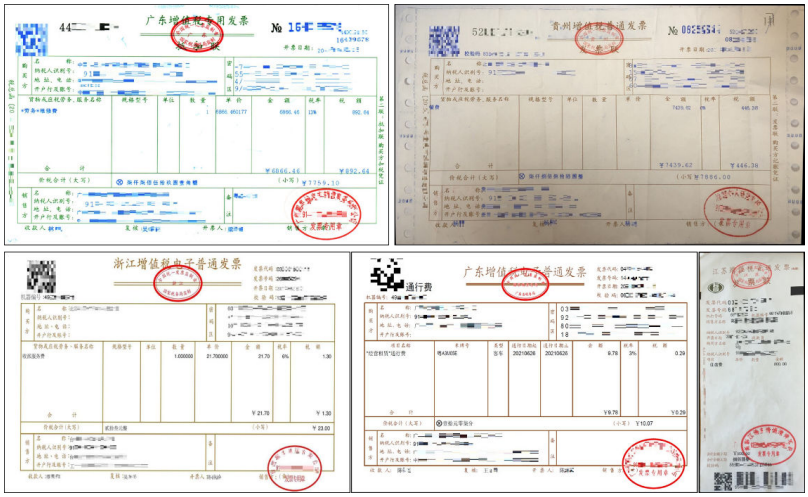


图 4-20 区块链发票示例图

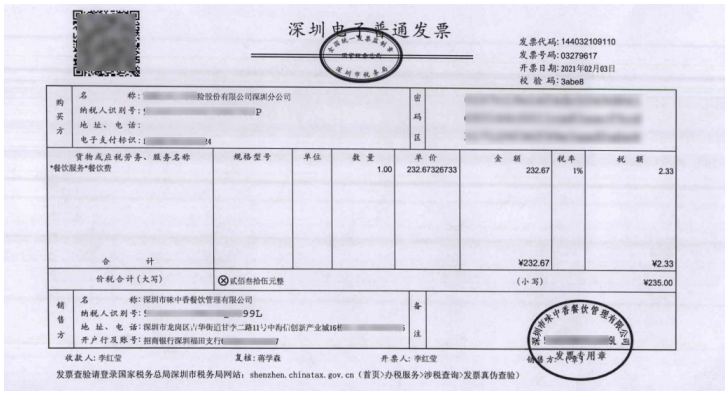


图 4-21 全电发票示例图（专用发票）



版权所有 © 华为云计算技术有限公司

图 4-22 全电发票示例图（普通发票）



图 4-23 全电发票示例图（旅客运输服务电子发票）



对于区块链发票与增值税电子普票的区别参考如下：

- 一般电子普通发票除了地名，均标明“增值税电子普通发票”，区块链发票除了地名外只标明“电子普通发票”。
- 增值税电子普通发票一般有机器编码。
- 校验码数位不一样，增值税电子普通发票校验码是由20位阿拉伯数字组成；区块链发票校验码5位数字或英文字母与阿拉伯数字组合。

说明

- 该增值税发票仅限于中华人民共和国境内使用的增值税发票。
- 如果图片中包含多张卡证票据，请调用**智能分类识别**服务。

约束与限制

- 支持JPEG、JPG、PNG、BMP、TIFF、PDF、OFD格式，多页PDF默认识别第一页，或者您可以指定要识别的页码。
- 图像各边的像素大小在100px到8192px之间。单个图片、PDF文件其对应的Base64编码不超过10MB。

版权所有 © 华为云计算技术有限公司

- 图像中发票区域有效占比超过80%。
- 支持图像中增值税发票任意角度的水平旋转。
- 图片中发票宽高比例与真实发票比例一致。
- 支持的增值税发票包括：增值税专用发票、增值税普通发票、增值税电子专用发票、增值税电子普通发票（含通行费发票、区块链发票）、增值税普通发票（卷票）、全电普通发票、全电专用发票、全电旅客运输服务电子发票。
- 文字识别服务属于公有云服务，线上用户资源共享，如果需要多并发请求，请提前**联系我们**。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/vat-invoice

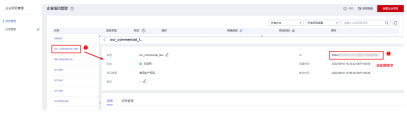
表 4-178 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

版权所有 © 华为云计算技术有限公司

请求参数

表 4-179 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-180 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，要求Base64编码后大小不超过10MB。 单个图片、PDF、OFD文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图片最短边不小于100px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF、PDF、OFD格式，多页OFD仅识别第一页。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	该参数与image二选一。 单个图片、PDF、OFD文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不建议存在中文字符，若存在，中文需要进行utf8编码。
advanced_mode	否	Boolean	默认为false，如果传参为true，则返回更多字段，具体请参见**表4-182**。
return_text_location	否	Boolean	默认为false，如果传参为true，则返回text_location字段表示文本框在原图位置。
page_num	否	Integer	页码，默认为1，返回第一页结果。如果传参不大于PDF页数，则返回对应PDF页的结果。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
ofd_to_image_mode	否	Boolean	OFD转图片的模式开关。针对输入文件为OFD时，有两种处理方案： 方案一：直接解析OFD文件，并提取结构化信息 方案二：转成图片后再进行识别，该方案更鲁棒，但是时延会显著增加 取值范围包括： • false：选择方案一 • true：选择方案二 未传入该参数时默认为false，即选择方案一。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（status code）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-181 响应 Body 参数

参数	参数类型	描述
result	**VatInvoiceResult** object	识别结果。 调用失败时不返回此字段。

表 4-182 VatInvoiceResult

参数	参数类型	描述
title	String	增值税发票标题。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
type	String	增值税发票类型： ● special：增值税专用发票 ● normal：增值税普通发票 ● electronic：增值税电子普通发票（含区块链发票） ● special_electronic：增值税电子专用发票 ● toll：增值税电子普通发票（通行费） ● roll：增值税普通发票（卷票） ● fully_digitalized_special_electronic：全电专用发票 ● fully_digitalized_normal_electronic：全电普通发票
invoice_tag	String	增值税发票左上角标志。当“advanced_mode”设置为“true”时才返回。包含：通行费、代开、成品油、收购、机动车、旅客运输服务。
sum_amount	String	小计金额，当传入为全电发票时返回此字段。
sum_tax	String	小计税额，当传入为全电发票时返回此字段。
serial_number	String	发票联次。 当“advanced_mode”设置为“true”时才返回。
attribution	String	发票归属地。 当“advanced_mode”设置为“true”时才返回。
supervision_seal	Array of strings	发票监制章。 当“advanced_mode”设置为“true”时才返回。
code	String	发票代码。
print_code	String	机打代码。当“advanced_mode”设置为“true”时才返回。
machine_number	String	机器编号。 当“advanced_mode”设置为“true”时才返回。
print_number	String	卷票是机打号码，普通发票和专用发票在右上角，电子发票、通行费发票无此字段。 当“advanced_mode”设置为“true”时才返回。
check_code	String	发票校验码，特定类型增值税发票内不存在该信息时返回空字符串。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
number	String	发票号码。
issue_date	String	开票日期。
encryption_block	String	密码区。
buyer_name	String	购买方名称。
buyer_id	String	购买方纳税人识别号。
buyer_address	String	购买方地址、电话。
buyer_bank	String	购买方开户行及账号。
seller_name	String	销售方名称。
seller_id	String	销售方纳税人识别号。
seller_address	String	销售方地址、电话。
seller_bank	String	销售方开户行及账号。
subtotal_amount	String	合计金额。
subtotal_tax	String	合计税额。
total	String	价税合计。
total_in_words	String	价税合计（大写）。 当“advanced_mode”设置为“true”时才返回。
remarks	String	备注。 当“advanced_mode”设置为“true”时才返回。
receiver	String	收款人。 当“advanced_mode”设置为“true”时才返回。
reviewer	String	复核。 当“advanced_mode”设置为“true”时才返回。
issuer	String	开票人。 当“advanced_mode”设置为“true”时才返回。
seller_seal	Array of strings	销售方发票专用章。 当“advanced_mode”设置为“true”时才返回。
item_list	Array of **ItemList** objects	货物或应税劳务列表。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
province	String	省。
city	String	市。
confidence	Object	各个字段的置信度。 当“advanced_mode”设置为“true”时才返回。
text_location	Object	文本框在原图位置。输出左上、右上、右下、左下四个点坐标。 当“return_text_location”设置为“true”时才返回。 **说明** 输入数据格式是PDF时，返回的字段坐标仅用于参考，表示字段间的相对位置关系。 输入数据格式是OFD时，返回的字段坐标为空列表。
belong_buyer_name	String	销货清单的购买方名称。 当传入图片为发票销货清单时返回此字段。
belong_seller_name	String	销货清单的销售方名称。 当传入图片为发票销货清单时返回此字段。
belong_vat_code	String	所属的增值税发票代码。 当传入图片为发票销货清单时返回此字段。
belong_number	String	销货清单的开票号码。 当传入图片为发票销货清单时返回此字段。
belong_pages	String	销货清单的总页码数。 当传入图片为发票销货清单时返回此字段。
belong_current_page	String	销货清单的当前页码。 当传入图片为发票销货清单时返回此字段。
belong_remarks	String	销货清单的备注。 当传入图片为发票销货清单时返回此字段。
belong_issue_date	String	销货清单的填开日期。 当传入图片为发票销货清单时返回此字段。
sales_mark	Boolean	是否是销货清单： ● true：输入图片是销货清单。 ● false：输入图片不是销货清单。
belong_sum_amount	String	销货清单的小计金额。 当传入图片为发票销货清单时返回此字段。
belong_sum_tax	String	销货清单的小计税额。 当传入图片为发票销货清单时返回此字段。
belong_subtotal_amount	String	销货清单的总计或者合计金额。 当传入图片为发票销货清单时返回此字段。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
belong_subtotal_tax	String	销货清单的总计税额。 当传入图片为发票销货清单时返回此字段。
belong_discount_amount	String	销货清单的折扣金额。 当传入图片为发票销货清单时返回此字段。
belong_discount_tax	String	销货清单的折扣税额。 当传入图片为发票销货清单时返回此字段。
belong_item_list	Array of **BelongItemList** objects	销货清单的货物（劳务）名称列表。 当传入图片为发票销货清单时返回此字段。
passenger_travel_item_list	Array of **PassengerTravelItemList** objects	旅客运输服务的出行信息列表。 当传入图片为旅客运输服务发票时返回此字段。

表 4-183 ItemList

参数	参数类型	描述
name	String	货物或应税劳务、服务名称。
specification	String	规格型号。
unit	String	单位。
quantity	String	数量。
unit_price	String	单价。
license_plate_number	String	车牌号码。 当“type”被识别为“toll”且“advanced_mode”设置为“true”时才返回。
amount	String	金额。
tax_rate	String	税率。
tax	String	税额。
end_date	String	通行日期止。 当“type”被识别为“toll”且“advanced_mode”设置为“true”时才返回。
start_date	String	通行日期起。 当“type”被识别为“toll”且“advanced_mode”设置为“true”时才返回。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
vehicle_type	String	车辆类型。 当“type”被识别为“toll”且“advanced_mode”设置为“true”时才返回。

表 4-184 BelongItemList

参数	参数类型	描述
name	String	货物或应税劳务、服务名称。
item_number	String	序号。
specification	String	规格型号。
unit	String	单位。
quantity	String	数量。
unit_price	String	单价。
amount	String	金额。
tax_rate	String	税率。
tax	String	税额。

表 4-185 PassengerTravelItemList

参数	参数类型	描述
traveler_name	String	出行人。
id_number	String	有效身份证件号。
travel_date	String	出行日期。
departure_location	String	出发地。
arrival_location	String	到达地。
class	String	等级。
transportation_type	String	交通工具类型。

状态码： 400

版权所有 © 华为云计算技术有限公司

表 4-186 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。
  例如，增值税发票识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/vat-invoice”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 传入增值税发票的base64编码进行文字识别，并通过设置“advanced_mode”返回发票类型、联次、金额等信息。

```
POST https://{endpoint}/v2/{project_id}/ocr/vat-invoice
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
    "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAj...",
    "advanced_mode": true
 }
```

- 传入增值税发票的url进行文字识别，并通过设置“advanced_mode”返回发票类型、联次、金额等信息。

```
POST https://{endpoint}/v2/{project_id}/ocr/vat-invoice
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
     "url":"https://BucketName.obs.xxxx.com/ObjectName",
     "advanced_mode": true
 }
```

响应示例

状态码：200

成功响应示例

- 示例 1

```
{
  "result" : {
    "type" : "special",
    "title" : "深圳增值税电子专用发票",
    "serial_number" : "第二联：发票联",
```


版权所有 © 华为云计算技术有限公司

```
  "attribution" : "深圳",
  "supervision_seal" : [ "全国统一发票监制章", "深圳", "国家税务总局监制" ],
  "code" : "310316XXXX",
  "check_code" : "",
  "machine_number" : "310316XXXX",
  "print_number" : "",
  "number" : "60543XXX",
  "issue_date" : "2017年08月30日",
  "encryption_block" : "6/+1+733<672085+063>82>30<1872/1<>*312671<9<1-11208-746599*6/>
+7>2163+141-8737*4932+7970*11892126>0*-+7+78>1",
  "buyer_name" : "XX有限公司",
  "buyer_id" : "917107277650880665",
  "buyer_address" : "XXXX",
  "buyer_bank" : "XX银行20384543252XXXX",
  "seller_name" : "餐饮有限公司",
  "seller_id" : "9351099411892126",
  "seller_address" : "XXXX",
  "seller_bank" : "XX银行991101526087XXXX",
  "subtotal_amount" : "￥8850.00",
  "subtotal_tax" : "￥1504.50",
  "total" : "￥10354.50",
  "total_in_words" : "肆仟肆佰玖拾圆整",
  "remarks" : "校验码11XXXXXXXXXX",
  "receiver" : "XX",
  "reviewer" : "XX",
  "issuer" : "XX",
  "seller_seal" : [ "深圳市XXX有限公司", "91440300736259277E", "发票专用章" ],
  "item_list" : [ {
    "name" : "食品A",
    "specification" : "200g/盒",
    "unit" : "盒",
    "quantity" : "300",
    "unit_price" : "28.00",
    "license_plate_number" : "",
    "vehicle_type" : "",
    "start_date" : "",
    "end_date" : "",
    "amount" : "8400.00",
    "tax_rate" : "17%",
    "tax" : "1428.00"
  } ],
  "confidence" : {
    "type" : 0.996,
    "title" : 0.9962,
    "serial_number" : 0.9652,
    "attribution" : 0.996,
    "supervision_seal" : [ 0.997, 0.9945, 0.996 ],
    "code" : 0.99999,
    "check_code" : 0.843,
    "machine_number" : 0.907,
    "print_number" : 0.0,
    "number" : 1.9856,
    "issue_date" : 0.9848,
    "encryption_block" : 0.9922,
    "buyer_name" : 0.9854,
    "buyer_id" : 0.9869,
    "buyer_address" : 0.0,
    "buyer_bank" : "0. 0000",
    "seller_name" : 0.9883,
    "seller_id" : 0.9914,
    "seller_address" : 0.9952,
    "seller_bank" : 0.9829,
    "subtotal_amount" : 0.9533,
    "subtotal_tax" : 0.9167,
    "total" : 0.9444,
    "total_in_words" : 0.9854,
    "remarks" : 0.8762,
    "receiver" : 0.985,
    "reviewer" : 0.9759,
```


版权所有 © 华为云计算技术有限公司

```
    "issuer" : 0.9872,
    "seller_seal" : [ 0.9883, 0.9914, 0.9999 ],
    "item_list" : [ {
      "name" : 0.9779,
      "specification" : 0.0,
      "unit" : 0.0,
      "quantity" : 0.0,
      "unit_price" : 0.0,
      "license_plate_number" : 0.0,
      "vehicle_type" : 0.0,
      "start_date" : 0.0,
      "end_date" : 0.0,
      "amount" : 0.8227,
      "tax_rate" : 0.5183,
      "tax" : 0.8394
    } ],
    "province" : 0.9936,
    "city" : 0.9936
  },
  "province" : "广东省",
  "city" : "深圳市"
 }
}
```


- 示例 2


```
{
 "result" : {
   "belong_buyer_name" : "XX有限公司合肥分公司",
   "belong_seller_name" : "XX有限公司",
   "belong_vat_code" : "03XXXX900311",
   "belong_number" : "144XXX70",
   "belong_pages" : "2",
   "belong_current_page" : "2",
   "belong_remarks" : "",
   "belong_sum_amount" : "￥27.73",
   "belong_sum_tax" : "￥3.31",
   "belong_subtotal_amount" : "￥133.62",
   "belong_subtotal_tax" : "￥11.86",
   "belong_discount_amount" : "",
   "belong_discount_tax" : "",
   "belong_issue_date" : "",
   "sales_mark" : true,
   "belong_item_list" : [ {
     "name" : "肉及肉制品*帝皇鲜澳洲长期谷饲M3安格斯牛键肉500g/袋",
     "item_number" : "20",
     "specification" : "",
     "unit" : "",
     "quantity" : "",
     "unit_price" : "",
     "amount" : "-1.69",
     "tax_rate" : "免税",
     "tax" : "***"
   } ],
   "confidence" : {
     "belong_buyer_name" : 0.9794,
     "belong_seller_name" : 0.96,
     "belong_vat_code" : 0.9785,
     "belong_number" : 0.9723,
     "belong_pages" : 0.951,
     "belong_current_page" : 0.9184,
     "belong_remarks" : 0,
     "belong_sum_amount" : 0.9697,
     "belong_sum_tax" : "0.9676,",
     "belong_subtotal_amount" : 0.9646,
     "belong_subtotal_tax" : 0.9538,
     "belong_discount_amount" : 0,
     "belong_discount_tax" : 0,
     "belong_issue_date" : 0,
     "sales_mark" : 0.9725,
     "belong_item_list" : [ {
```


 版权所有 © 华为云计算技术有限公司

```
      "name" : "0.9629,",
      "item_number" : 0.9396,
      "specification" : 0,
      "unit" : 0,
      "quantity" : 0,
      "unit_price" : 0,
      "amount" : 0.9693,
      "tax_rate" : 0.9804,
      "tax" : 0.8532
    } ]
   },
   "text_location" : [ ]
  }
}
```

状态码：400

失败响应示例

```
{
   "error_code": "AIS.0103",
   "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入增值税发票的base64编码进行文字识别，并通过设置“advanced_mode”返回发票类型、联次、金额等信息。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeVatInvoiceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
```

 版权所有 © 华为云计算技术有限公司

```
            .build();
        RecognizeVatInvoiceRequest request = new RecognizeVatInvoiceRequest();
        VatInvoiceRequestBody body = new VatInvoiceRequestBody();
        body.withAdvancedMode(true);
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeVatInvoiceResponse response = client.recognizeVatInvoice(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入增值税发票的url进行文字识别，并通过设置“advanced_mode”返回发票类型、联次、金额等信息。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeVatInvoiceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeVatInvoiceRequest request = new RecognizeVatInvoiceRequest();
        VatInvoiceRequestBody body = new VatInvoiceRequestBody();
        body.withAdvancedMode(true);
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeVatInvoiceResponse response = client.recognizeVatInvoice(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
```

```
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入增值税发票的base64编码进行文字识别，并通过设置“advanced_mode”返回发票类型、联次、金额等信息。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeVatInvoiceRequest()
        request.body = VatInvoiceRequestBody(
            advanced_mode=True,
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_vat_invoice(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入增值税发票的url进行文字识别，并通过设置“advanced_mode”返回发票类型、联次、金额等信息。

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
```

 版权所有 © 华为云计算技术有限公司

```
credentials = BasicCredentials(ak, sk) \

client = OcrClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(OcrRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = RecognizeVatInvoiceRequest()
    request.body = VatInvoiceRequestBody(
        advanced_mode=True,
        url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
    )
    response = client.recognize_vat_invoice(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

- 传入增值税发票的base64编码进行文字识别，并通过设置“advanced_mode”返回发票类型、联次、金额等信息。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeVatInvoiceRequest{}
    advancedModeVatInvoiceRequestBody:= true
    imageVatInvoiceRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.VatInvoiceRequestBody{
        AdvancedMode: &advancedModeVatInvoiceRequestBody,
        Image: &imageVatInvoiceRequestBody,
    }
    response, err := client.RecognizeVatInvoice(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
```

 版权所有 © 华为云计算技术有限公司

```
    }
}
```

- 传入增值税发票的url进行文字识别，并通过设置“advanced_mode”返回发票类型、联次、金额等信息。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeVatInvoiceRequest{}
    advancedModeVatInvoiceRequestBody:= true
    urlVatInvoiceRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.VatInvoiceRequestBody{
        AdvancedMode: &advancedModeVatInvoiceRequestBody,
        Url: &urlVatInvoiceRequestBody,
    }
    response, err := client.RecognizeVatInvoice(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

版权所有 © 华为云计算技术有限公司

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.23 发票验真

说明

OCR发票验真识别套餐包计划于2025.07.14正式停售，为避免影响您目前的业务，请您及时进行相关迁移工作，详见**停售公告**。

功能介绍

支持10种增值税发票的信息核验：

增值税专用发票、增值税普通发票、增值税普通发票（卷式）、增值税电子专用发票、增值税电子普通发票、增值税电子普通发票（通行费）、二手车销售统一发票、机动车销售统一发票、区块链电子发票、全电发票，支持返回票面的全部信息。

该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

说明

- 每份发票，每天最多可查验5次。
- 可查验最近5年内（国家税务总局）增值税发票管理系统开具的发票。
- 税务局数据同步有时延，建议当天新开的发票在24小时后查询。

约束与限制

- 支持增值税专用发票、增值税普通发票、增值税普通发票（卷式）、增值税电子专用发票、增值税电子普通发票、增值税电子普通发票（通行费）、二手车销售统一发票、机动车销售统一发票、区块链电子发票和全电发票，共计10种发票信息核验。
- 每份发票每天最多可查验5次。可查验最近5年内（国家税务总局）增值税发票管理系统开具的发票。
- 税务局数据同步有时延，建议当天新开的发票在24小时后查询。
- 文字识别服务属于公有云服务，线上用户资源共享，如果需要多并发请求，请提前**联系我们**。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

版权所有 © 华为云计算技术有限公司

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/invoice-verification

表 4-187 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-188 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

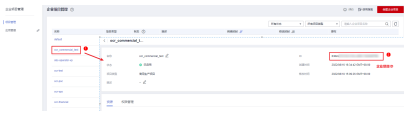
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-189 请求 Body 参数

参数	是否必选	参数类型	说明
code	是	String	发票代码。 发票种类为全电发票时，该参数须为空字符串或者只包含空格的字符串。
number	是	String	发票号码。
issue_date	是	String	发票日期格式YYYY-MM-DD。
check_code	否	String	校验码后六位。 • 以下种类发票，check_code参数为必选。增值税普通发票、增值税电子普通发票、增值税普通发票（卷式）、增值税电子普通发票（通行费）、区块链电子发票。 • 区块链电子发票需要填写5位校验码。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
subtotal_amount	否	String	合计金额。和票据上的金额的有效数字保持一致，例如票据上的金额为88.00，则需要输入字符串为“88.00”，否则可能验真不成功。如果输入“88”或“88.0”可能会产生"result_code": "1010", " Parameter error." 报错。 发票种类为全电发票时，该参数须为价税合计金额，其他票种使用不含税合计金额。 ● 以下种类发票，参数不可为空。 增值税专用发票、增值税电子专用发票、机动车销售统一发票、二手车销售统一发票、区块链电子发票、全电发票。 ● 填写发票合计金额（不含税） 增值税专用发票、增值税电子专用发票、机动车销售统一发票、区块链电子发票。 ● 填写发票价税合计 全电发票（机动车销售统一发票、增值税普通发票、增值税专用发票） ● 填写发票车价合计 二手车销售统一发票、全电发票（二手车销售统一发票） ● 填写发票合计 全电发票（航空运输电子客票行程单）。 ● 填写发票票价或者退票费 全电发票（铁路电子客票）。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（status code）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

版权所有 © 华为云计算技术有限公司

表 4-190 响应 Body 参数

参数	参数类型	描述
result	Object	识别结果。 调用失败时不返回此字段。 依据发票类型不同，返回参数不同。 • **增值税发票** 含增值税专用发票、增值税普通发票、增值税普通发票（卷式）、增值税电子专用发票、增值税电子普通发票、增值税电子普通发票（通行费）、区块链电子发票。 • **机动车销售统一发票** • **二手车销售统一发票**

表 4-191 InvoiceVerificationResult 增值税发票

参数	参数类型	描述
result	Object	调用成功时返回的识别结果。
type	String	发票类型，取值范围： • 增值税专用发票：vat_special • 增值税普通发票：vat_normal • 增值税普通发票（卷式）：vat_normal_roll • 增值税电子专用发票：vat_special_electronic • 增值税电子普通发票：vat_normal_electronic • 增值税电子普通发票（通行费）：vat_normal_electronic_toll • 区块链电子发票：blockchain_electronic • 全电发票（增值税专用发票）：fully_digitalized_special_electronic • 全电发票（增值税普通发票）：fully_digitalized_normal_electronic
code	String	发票代码。
number	String	发票号码。
issue_date	String	开票日期。
machine_number	String	机器编号。
check_code	String	校验码。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
status	String	发票状态，取值范围： ● 正常：valid ● 已作废：invalidated ● 已红冲：reversed
subtotal_amount	String	合计金额(不含税）。
subtotal_tax	String	合计税额。
total	String	价税合计（小写）。
total_in_words	String	价税合计（大写）。
remarks	String	备注。
receiver	String	收款人。
issuer	String	开票人。
reviewer	String	复核人。
buyer_name	String	购买方名称。
buyer_id	String	购买方纳税人识别号。
buyer_address	String	购买方地址、电话。
buyer_bank	String	购买方开户行及账号。
seller_name	String	销售方名称。
seller_id	String	销售方纳税人识别号。
seller_address	String	销售方地址、电话。
seller_bank	String	销售方开户行及账号。
deductible_toll	String	通行费发票返回信息： ● Y：可抵扣通行费 ● N：不可抵扣通行费
result_code	String	取值范围见**表4-197**。
result_message	String	取值范围见**表4-197**。
items	object	发票中包含的消费信息。
name	String	货物或应税劳务、服务名称、项目名称（通行费）。
specification	String	规格型号。
unit	String	单位。
quantity	String	数量。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
unit_price	String	单价。
amount	String	金额。
tax_rate	String	税率。
tax	String	税额。
license_plate_number	String	车牌号。
vehicle_type	String	类型。
start_date	String	通行日期起。
end_date	String	通行日期止。

表 4-192 InvoiceVerificationResult 机动车销售统一发票

参数	参数类型	描述
result	object	调用成功时返回的识别结果。
type	String	发票类型，取值范围： ● 机动车销售统一发票：motor_vehicle_sales ● 全电发票（机动车销售统一发票）：fully_digitalized_motor_vehicle_sales
code	String	发票代码。
number	String	发票号码。
issue_date	String	开票日期。
machine_number	String	机器编号。
status	String	发票状态，取值范围： ● 正常：valid ● 已作废：invalidated ● 已红冲：reversed
subtotal_amount	String	不含税价（小写）。
total	String	价税合计（小写）。
total_in_words	String	价税合计（大写）。
issuer	String	开票人。
buyer_name	String	购方名称。
buyer_organization_number	String	购买方身份证号码/组织机构代码。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
buyer_id	String	购买方纳税人识别号。
seller_name	String	销货单位名称。
seller_id	String	销售方纳税人识别号。
seller_phone	String	销售方电话。
seller_account	String	销售方账号。
seller_address	String	销售方地址。
seller_bank	String	销售方开户银行。
tax_rate	String	税率。
tax	String	税额。
tax_authority	String	主管税务机关名称。
tax_authority_code	String	主管税务机关代码。
tax_payment_receipt	String	完税凭证号码。
tonnage	String	吨位。
seating_capacity	String	限乘人数。
vehicle_type	String	车辆类型。
brand_model	String	厂牌型号。
manufacturing_location	String	产地。
quality_certificate	String	合格证号。
import_certificate	String	进口证明书号。
inspection_number	String	商检单号。
engine_number	String	发动机号码。
vehicle_identification_number	String	车辆识别代号/车架号码。
result_code	String	取值范围见**表4-197**。
result_message	String	取值范围见**表4-197**。

版权所有 © 华为云计算技术有限公司

表 4-193 InvoiceVerificationResult 二手车销售统一发票

参数	参数类型	描述
result	object	调用成功时返回的识别结果。
type	String	发票类型，取值范围： ● 二手车销售统一发票：used_vehicle_sales ● 全电发票（二手车销售统一发票）：fully_digitalized_used_vehicle_sales
code	String	发票代码。
number	String	发票号码。
issue_date	String	开票日期。
machine_number	String	机器编号。
status	String	发票状态，取值范围： ● 正常：valid ● 已作废：invalidated ● 已红冲：reversed
total	String	车价合计（小写）。
total_in_words	String	车价合计（大写）。
remarks	String	备注。
issuer	String	开票人。
buyer_name	String	买方单位/个人。
buyer_organization_number	String	买单位代码/身份证号码。
buyer_address	String	买方单位/个人住址。
buyer_phone	String	买方电话。
seller_name	String	卖方单位/个人。
seller_organization_number	String	卖方单位代码/身份证号码。
seller_address	String	卖方单位 /个人住址。
seller_phone	String	卖方电话。
vehicle_license_plate_number	String	车牌照号。
registration_certificate_number	String	登记证号。
vehicle_type	String	车辆类型。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
vehicle_identification_number	String	车辆识别代号/车架号码。
brand_model	String	厂牌型号。
transfer_motor_vehicle_department_name	String	转入地车辆车管所名称。
auction_unit_name	String	经营，拍卖单位。
auction_unit_address	String	经营，拍卖单位地址。
auction_unit_tax_number	String	经营，拍卖单位纳税人识别号。
auction_unit_bank_and_account	String	开户银行及账号。
auction_unit_phone	String	经营，拍卖单位电话。
used_vehicle_market_name	String	二手车市场。
used_vehicle_market_tax_number	String	二手车市场纳税人识别号。
used_vehicle_market_address	String	二手车市场地址。
used_vehicle_market_bank_and_account	String	二手车市场开户银行及账号。
used_vehicle_market_phone	String	二手车市场电话。
result_code	String	取值范围见**表4-197**。
result_message	String	取值范围见**表4-197**。

表 4-194 InvoiceVerificationResult 全电发票（航空运输电子客票行程单）

参数	参数类型	描述
result	object	调用成功时返回的调用结果。
type	String	发票类型，取值范围： 全电发票（航空运输电子客票行程单）：fully_digitalized_flight_itinerary。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
international_flag	String	国内国际标识。
number	String	发票号码。
status	String	发票状态，取值范围： ● 正常：valid ● 已作废：invalidated ● 已红冲：reversed
passenger_name	String	旅客姓名。
id_number	String	有效身份证件号码。
gp_number	String	gp单号。
tax	String	增值税税额。
total	String	合计。
e_ticket_number	String	电子客票号码。
issue_date	String	填开日期。
buyer_name	String	购买方名称。
buyer_id	String	购买方纳税人识别号。
itinerary_list	Array of **ItineraryList** objects	机票行程列表。

表 4-195 ItineraryList

参数	参数类型	描述
departure_station	String	始发站。
destination_station	String	目的站。
carrier	String	承运人。
flight	String	航班号。
cabin_class	String	座位等级。
date	String	日期。
time	String	时间。
fare_basis	String	客票类别。
flight_segment	String	航段。

版权所有 © 华为云计算技术有限公司

表 4-196 InvoiceVerificationResult 全电发票（铁路电子客票）

参数	参数类型	描述
result	object	调用成功时返回的调用结果。
type	String	发票类型，取值范围： 全电发票（铁路电子客票）：fully_digitalized_train_ticket。
number	String	发票号码。
status	String	发票状态，取值范围： ● 正常：valid ● 已作废：invalidated ● 已红冲：reversed
business_type	String	业务类型，取值范围： ● 售票：ticketing ● 退票：refund
issue_date	String	开票日期。
name	String	车票持有人姓名。
id_number	String	车票持有人的身份证号。
ticket_price	String	票价。
tax_rate	String	税率。
tax	String	税额。
departure_station	String	始发站。
destination_station	String	终点站。
train_number	String	车次。
departure_date	String	乘车日期。
departure_time	String	出发时间。
seat_category	String	席别。
carriage	String	车厢。
seat_number	String	座位号。
e_ticket_number	String	电子客票号码。
air_conditioning	String	空调特征（是否是空调车厢）。
buyer_name	String	购买方名称。
unified_social_credit_code	String	购买方统一社会信用号码。

版权所有 © 华为云计算技术有限公司

表 4-197 业务状态码

业务状态码（result_code）	描述信息（result_message）	处理方法
1000	The invoice verification information is consistent with the tax authority’s information. 发票验证信息与税务机关的信息一致。	无需处理，发票验证信息与税务机关信息一致。
1001	One invoice can be verified for only five times per day. 一张发票每天只能验证5次。	超过当天查验次数，请第二天再查验。
1002	The invoice verification information cannot be found in the tax authority’s information. 在税务机关的信息中找不到发票验证信息。	被查验发票信息有误，请核对发票信息是否填写正确。
1003	The invoice verification information is inconsistent with the tax authority’s information. 发票验证信息与税务机关的信息不一致。	被查验发票信息有误，请核对发票信息是否填写正确。
1004	Only invoices issued within the last five years can be verified. 只能验证最近五年内开具的发票。	超过5年的发票不能查验。
1005	Unsupported invoice type. 不支持的发票类型。	被查验发票信息有误，请核对发票信息是否填写正确。
1006	The invoice verification information is abnormal in the tax authority’s information. The tax authority's interface is malfunctioning. Retry in 2 hours. 发票认证信息在税务机关信息中异常。	税局机关接口异常，请等候税局机关恢复后再重试，建议2小时后重试。

版权所有 © 华为云计算技术有限公司

业务状态码（result_code）	描述信息（result_message）	处理方法
1007	This batch of invoices is being verified. 该批发票正在校验中。	该批次发票正在核验中，请第二天再查验。
1008	The request to the invoice verification service provided by the State Taxation Administration of China timed out. The request to verify the invoice with the tax agency timed out. Retry in 2 hours. 向税务机关提供的发票验证服务的请求超时。	（税局机关）发票查询服务请求超时，请等候税局机关恢复后再重试，建议2小时后重试。
1009	Verification request rejected. Online invoice verification is currently not supported in this region. 验证请求被拒绝，此区域暂未开通线上查验通道。	（税局机关）查验请求被拒绝，此区域暂未开通线上查验通道。
1010	Parameter error. 参数错误。	参数错误，请检查请求参数是否填写正确。
0118	Service request error or the tax authority’s interface was malfunctioning. Service request error as the tax agency's interface response timed out. 服务请求错误或税务机关的接口出现故障。	请等待税局机关接口恢复，建议2小时后重试。

状态码： 400

表 4-198 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

版权所有 © 华为云计算技术有限公司

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，发票验真服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/invoice-verification”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 传入发票的代码、号码、发票日期、校验码后六位进行发票真伪核验

```
POST https://{endpoint}/v2/{project_id}/ocr/invoice-verification
 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
    "code": "031001900000",
    "number": "12528000",
    "issue_date": "2020-08-09",
    "check_code": "000000"
 }
```

响应示例

状态码：200

成功响应示例（增值税普通发票）

```
{
   "result": {
      "type": "vat_normal",
      "code": "03xxxxx00204",
      "number": "12528000",
      "issue_date": "20200809",
      "machine_number": "661xxxxx1000",
      "check_code": "79xxxxx8040xxxx00000",
      "status": "valid",
      "subtotal_amount": "1400.00",
      "subtotal_tax": "0.00",
      "total": "1400.00",
      "total_in_words": "壹仟肆佰元整",
      "remarks": "3091",
      "receiver": "",
      "issuer": "",
      "reviewer": "",
      "buyer_name": "xxx公司",
      "buyer_id": "91xxxxx0192xxxx00",
      "buyer_address": "",
      "buyer_bank": "",
      "seller_name": "xxx投资发展有限公司",
      "seller_id": "913101177858977000",
      "seller_address": "xx市xx区xx镇xx路xx号 021-xxxx1000",
      "seller_bank": "xx银行xx支行310xxxxxxxxx00000000",
      "deductible_toll": "N",
      "items": [
         {
            "name": "",
            "specification": "",
            "unit": "1",
            "quantity": "1",
            "unit_price": "1400.00",
            "amount": "1400.00",
```


版权所有 © 华为云计算技术有限公司

```
            "tax_rate": "免税",
            "tax": "***",
            "license_plate_number": "",
            "vehicle_type": "",
            "start_date": "",
            "end_date": ""
         }
      ],
      "result_code": "1000",
      "result_message": "The invoice verification information is consistent with the tax authority’s information."
   }
}
```


成功响应示例（机动车销售统一发票）


```
{
   "result": {
      "type": "motor_vehicle_sales",
      "code": "14xxxxx20000",
      "number": "00xxxx94",
      "issue_date": "20171128",
      "machine_number": "49xxxxx10000",
      "status": "valid",
      "subtotal_amount": "68376.07",
      "total": "80000.00",
      "total_in_words": "捌万元整",
      "issuer": "",
      "buyer_name": "xx",
      "buyer_organization_number": "",
      "buyer_id": "",
      "seller_name": "xx汽车销售服务有限公司",
      "seller_id": "91xxxxx059xxxx0000",
      "seller_phone": "0898-xxxxxx00",
      "seller_account": "265xxxxxx000",
      "seller_address": "xx省xx市xx大道176号",
      "seller_bank": "中国银行xx支行",
      "tax_rate": "0.17",
      "tax": "11623.93",
      "tax_authority": "xx区局二分局户籍二组",
      "tax_authority_code": "146xxxxxx00",
      "tax_payment_receipt": "",
      "tonnage": "-",
      "seating_capacity": "5",
      "vehicle_type": "轿车",
      "brand_model": "北京牌BJxxxxxx000",
      "manufacturing_location": "北京",
      "quality_certificate": "YJxxxxxx270000",
      "import_certificate": "-",
      "inspection_number": "-",
      "engine_number": "BAxxxxx000",
      "vehicle_identification_number": "LNBSCxxxxx0000",
      "result_code": "1000",
      "result_message": "The invoice verification information is consistent with the tax authority’s information."
   }
}
```


成功响应示例（二手车销售统一发票）


```
{
   "result": {
      "type": "used_vehicle_sales",
      "code": "03xxxxxx0000",
      "number": "00xxx000",
      "issue_date": "20191202",
      "machine_number": "661xxxxxx000",
      "status": "valid",
      "total": "20000.00",
      "total_in_words": "贰万元整",
```


 版权所有 © 华为云计算技术有限公司

```
        "remarks": "卖方纳税人识别号:33xxxxx900xxxxx00 买方纳税人识别号:37xxxxxx561xxxxx00 初次登记:
2014-07-24 免征税款",
        "issuer": "",
        "buyer_name": "xxx",
        "buyer_organization_number": "37xxxxxx561xxxxx00",
        "buyer_address": "xx省xx市xx路街道办事处xx庄居委会xx号",
        "buyer_phone": ".",
        "seller_name": "xxx",
        "seller_organization_number": "339xxxxxx009xxxx00",
        "seller_address": "xx省xx市xx市xx路xx号",
        "seller_phone": ".",
        "vehicle_license_plate_number": "苏Kxxxx0",
        "registration_certificate_number": "32xxxxxx0000",
        "vehicle_type": "小型轿车",
        "vehicle_identification_number": "LBxxxxxxZ3xxxxxx000",
        "brand_model": "xx牌HQxxxx000",
        "transfer_motor_vehicle_department_name": "xx车辆管理所",
        "auction_unit_name": "",
        "auction_unit_address": "",
        "auction_unit_tax_number": "",
        "auction_unit_bank_and_account": "",
        "auction_unit_phone": "",
        "used_vehicle_market_name": "xx旧机动车交易市场",
        "used_vehicle_market_tax_number": "91xxxxx2MAxxxxx000",
        "used_vehicle_market_address": "xx市xx区xx路xxx号",
        "used_vehicle_market_bank_and_account": "xxxx银行股份有限公司xxxx物流园分理处10xxxxx040xxxx00",
        "used_vehicle_market_phone": "13xxxxx0000",
        "result_code": "1000",
        "result_message": "The invoice verification information is consistent with the tax authority’s
information."
    }
}
```


状态码：400

失败响应示例


```
{
    "error_code": "AIS.0118",
    "error_msg": "Service request error."
}
```


SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

传入发票的代码、号码、发票日期、校验码后六位进行发票真伪核验

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeInvoiceVerificationSolution {
```

 版权所有 © 华为云计算技术有限公司

```
    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeInvoiceVerificationRequest request = new RecognizeInvoiceVerificationRequest();
        InvoiceVerificationRequestBody body = new InvoiceVerificationRequestBody();
        body.withCheckCode("000000");
        body.withIssueDate("2020-08-09");
        body.withNumber("12528000");
        body.withCode("031001900000");
        request.withBody(body);
        try {
            RecognizeInvoiceVerificationResponse response = client.recognizeInvoiceVerification(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

传入发票的代码、号码、发票日期、校验码后六位进行发票真伪核验

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()
```

 版权所有 © 华为云计算技术有限公司

```
    try:
        request = RecognizeInvoiceVerificationRequest()
        request.body = InvoiceVerificationRequestBody(
            check_code="000000",
            issue_date="2020-08-09",
            number="12528000",
            code="031001900000"
        )
        response = client.recognize_invoice_verification(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

传入发票的代码、号码、发票日期、校验码后六位进行发票真伪核验

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeInvoiceVerificationRequest{}
    checkCodeInvoiceVerificationRequestBody:= "000000"
    request.Body = &model.InvoiceVerificationRequestBody{
        CheckCode: &checkCodeInvoiceVerificationRequestBody,
        IssueDate: "2020-08-09",
        Number: "12528000",
        Code: "031001900000",
    }
    response, err := client.RecognizeInvoiceVerification(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例（增值税普通发票）
400	失败响应示例

状态码请参见**状态码**。

错误码

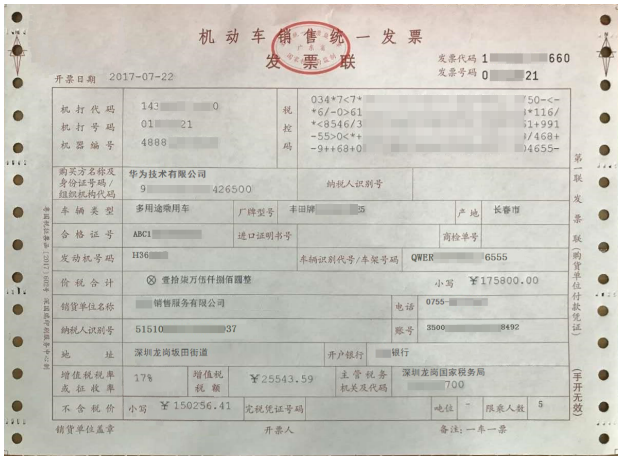
错误码请参见**错误码**。

4.24 机动车销售发票识别

功能介绍

识别机动车销售发票、二手车销售发票图片中的文字内容，服务能自动分辨两种类型，并将识别的结果以JSON格式返回给用户。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-24 机动车销售发票示例图



版权所有 © 华为云计算技术有限公司

图 4-25 机动车销售发票示例图

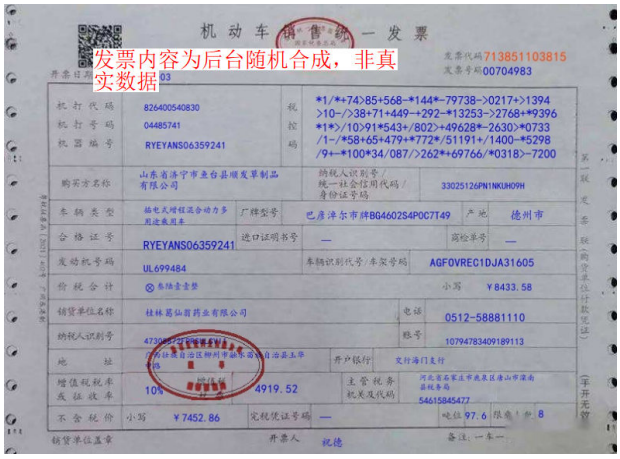
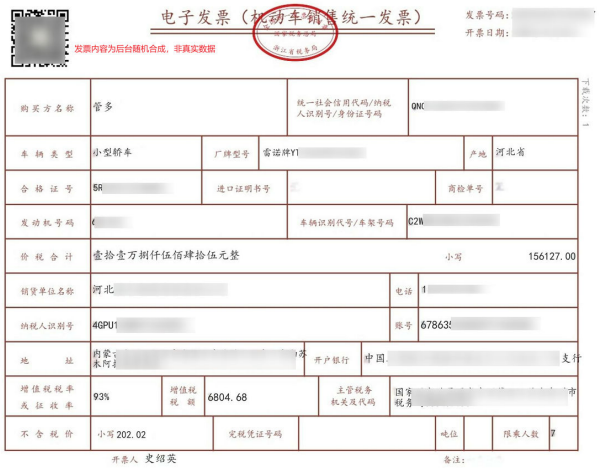


图 4-26 二手车销售发票示例图

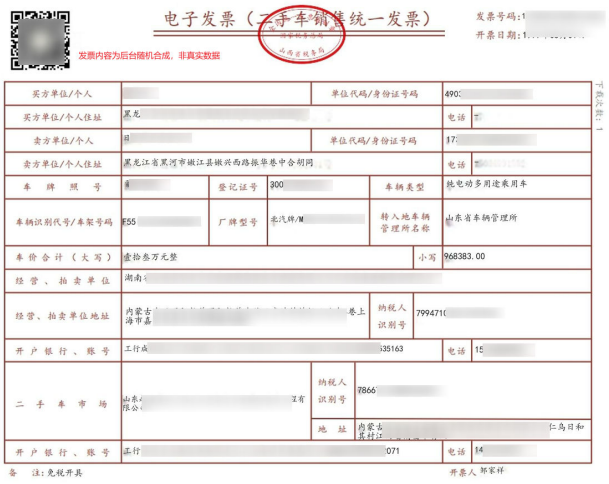


图 4-27 电子发票（机动车销售统一发票）示例图



版权所有 © 华为云计算技术有限公司

图 4-28 电子发票（二手车销售统一发票）示例图



说明

如果图片中包含多张卡证票据，请调用**智能分类识别**服务。

约束与限制

- 支持识别PNG、JPG、JPEG、BMP、TIFF、PDF格式。
- 图像各边的像素大小在100px到8000px之间。单个图片、PDF文件其对应的Base64编码不超过10MB。
- 图像中识别区域有效占比超过80%，保证整张发票及其边缘包含在图像内。
- 支持图像中发票任意角度的水平旋转。
- 支持少量扭曲，扭曲后图像中的发票长宽比与实际发票相差不超过10%。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/mvs-invoice

版权所有 © 华为云计算技术有限公司

表 4-199 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-200 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

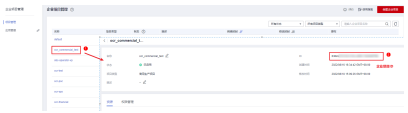
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-201 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，要求 单个图片、PDF文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图片最短边不小于100px，最长边不超过8192px。支持JPEG、JPG、PNG、BMP、TIFF、PDF格式。多页PDF仅识别第一页。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
url	否	String	该参数与image二选一。 单个图片、PDF文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
return_text_location	否	Boolean	如果为True，返回体中会包含text_location对象，内容是各字段的检测框四点坐标。 如果为False或者没有这个key，返回体中不包含text_location对象。
return_confidence	否	Boolean	如果为True，返回体中包含confidence对象，内容是各字段的置信度。 如果为False或者没有这个key，返回体中不包含confidence对象。
type	否	String	如果没有type字段则默认返回原机动车销售发票出参。 如果存在type字段但是不属于 auto、new或者used三个枚举值，API返回AIS.0101入参错误。 如果type为auto，API自动判断发票类型，并在返回参数中添加type出参以指明发票类型。 如果type为new，API在检测出的类型为机动车发票时返回原版机动车发票出参并添加type出参（机动车销售统一发票），不一致时报错AIS.0104图像质量差。 如果type为used，API在检测出的类型为二手车时返回二手车发票出参，并添加type出参（“二手车销售统一发票”或“电子发票（二手车销售统一发票）”），不一致时报错AIS.0104图像质量差。

版权所有 © 华为云计算技术有限公司

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-202 响应 Body 参数

参数	参数类型	描述
result	MvsInvoiceResult object	识别结果。 调用失败时不返回此字段。

表 4-203 MvsInvoiceResult

参数	参数类型	描述
code	String	发票代码。
number	String	发票号码。
machine_printed_code	String	机打代码。
machine_printed_number	String	机打号码。
issue_date	String	开票日期。
machine_number	String	机器编号。
buyer_name	String	购买方的名称、二手车买方单位/个人。
buyer_organization_number	String	购买方身份证号码/组织机构代码。
buyer_id	String	购买方纳税人识别号、二手车买方单位代码/身份证号。
buyer_address	String	二手车买方单位/个人住址，当请求参数"type"设置为"auto"或"used"时才返回。
buyer_phone	String	二手车买方单位/个人电话，当请求参数"type"设置为"auto"或"used"时才返回。
seller_name	String	销货单位名称、二手车卖方单位/个人。
seller_phone	String	销售方电话、二手车卖方电话。
seller_id	String	销售方纳税人识别号、二手车卖方单位代码/身份证号。
seller_account	String	销售方账号。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
seller_address	String	销售方地址、二手车卖方单位/个人地址。
licence_plate_number	String	二手车车牌照号，当请求参数"type"设置为"auto"或"used"时才返回。
registration_number	String	二手车登记证号，当请求参数"type"设置为"auto"或"used"时才返回。
dept_motor_vehicles	String	二手车转入地车管所名称，当请求参数"type"设置为"auto"或"used"时才返回。
seller_bank	String	销售方开户行。
vehicle_type	String	车辆类型。
brand_model	String	厂牌型号。
manufacturing_location	String	产地。
quality_certificate	String	合格证号。
import_certificate	String	进口证明书号。
inspection_number	String	商检单号。
engine_number	String	发动机号码。
vehicle_identification_number	String	车辆识别代号/车架号码。
tonnage	String	吨位。
seating_capacity	String	限乘人数。
tax_authority	String	主管税务机关。
tax_authority_code	String	主管税务机关代码。
tax_payment_receipt	String	完税凭证号码。
tax_rate	String	增值税税率或征收率。
tax	String	增值税税额。
tax_exclusive_price	String	不含税价。
total	String	价税合计、二手车车价合计（小写）。
total_chinese	String	价税合计大写、二手车车价合计（大写）。
fiscal_code	String	税控码。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
auction_org_name	String	二手车经营拍卖单位名称，当请求参数"type"设置为"auto"或"used"时才返回。
auction_org_address	String	二手车经营拍卖单位地址，当请求参数"type"设置为"auto"或"used"时才返回。
auction_org_id	String	二手车经营拍卖单位纳税人识别号，当请求参数"type"设置为"auto"或"used"时才返回。
auction_org_bank_account	String	二手车经营拍卖单位银行和账号，当请求参数"type"设置为"auto"或"used"时才返回。
auction_org_phone	String	二手车经营拍卖单位电话，当请求参数"type"设置为"auto"或"used"时才返回。
used_vehicle_market_name	String	二手车市场名称，当请求参数"type"设置为"auto"或"used"时才返回。
used_vehicle_market_id	String	二手车市场纳税人识别号，当请求参数"type"设置为"auto"或"used"时才返回。
used_vehicle_market_address	String	二手车市场地址，当请求参数"type"设置为"auto"或"used"时才返回。
used_vehicle_market_bank_account	String	二手车市场银行和账号，当请求参数"type"设置为"auto"或"used"时才返回。
used_vehicle_market_phone	String	二手车市场电话，当请求参数"type"设置为"auto"或"used"时才返回。
reverse_issue	Boolean	二手车反向开具标识，True表示是反向开具发票，False表示不是反向开具发票。 当请求参数"type"设置为"auto"或"used"时才返回。
remark	String	备注。
drawer_name	String	开票人。
type	String	枚举值，当入参中包含type时返回。取值范围包括： ● 机动车销售统一发票 ● 电子发票（机动车销售统一发票） ● 二手车销售统一发票 ● 电子发票（二手车销售统一发票）
text_location	Object	检测框对象，内部为字段-框坐标对。如 code: [[x0, y0],[x1,y1],[x2,y2],[x3,y3]], 点的顺序是左上角、右上角、右下角、左下角。如果原图找不到字段，返回空列表。 **说明** 输入数据格式是PDF时，返回的字段坐标仅用于参考，表示字段间的相对位置关系。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
confidence	Object	字段文字内容置信度，内容为字段-字符置信度对,如code:0.9999。这个数值为字段中每个字符置信度，格式为fp32，保留四位。若字段不存在则返回0.0。

状态码： 400

表 4-204 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。
  例如，机动车销售发票识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/mvs-invoice”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 传入机动车销售发票图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/mvs-invoice

 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:



{
  "image" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...",
  "type" : "auto"
}
```

- 传入机动车销售发票图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/mvs-invoice

 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
 Request Body:
 {
```


版权所有 © 华为云计算技术有限公司

```
    "url":"https://BucketName.obs.xxxx.com/ObjectName"
}
```

响应示例

状态码：200

成功响应示例1

```
{
  "result" : {
    "code" : "144031624150",
    "number" : "00446874",
    "machine_printed_code" : "144031624150",
    "machine_printed_number" : "00446874",
    "issue_date" : "2017-07-19",
    "machine_number" : "49991450XXXX",
    "buyer_name" : "深圳市宝安区XXX",
    "buyer_organization_number" : "XXX",
    "buyer_id" : "XXX",
    "seller_name" : "深圳市XXX服务有限公司",
    "seller_phone" : "0755-XXX",
    "seller_id" : "XXX",
    "seller_account" : "XXX",
    "seller_address" : "深圳市宝安区松岗街道办XXX",
    "seller_bank" : "建设银行深圳XX支行",
    "vehicle_type" : "多用途乘用车",
    "brand_model" : "东风日产牌/XXX",
    "manufacturing_location" : "大连市",
    "quality_certificate" : "XXX",
    "import_certificate" : "",
    "inspection_number" : "",
    "engine_number" : "XXX",
    "vehicle_identification_number" : "XXX",
    "tonnage" : "",
    "seating_capacity" : "5",
    "tax_authority" : "深圳市宝安区国家税务局XXX税务分局",
    "tax_authority_code" : "144030614",
    "tax_payment_receipt" : "",
    "tax_rate" : "17%",
    "tax" : "￥21475.21",
    "tax_exclusive_price" : "￥126324.79",
    "total" : "￥147800.00",
    "total_chinese" : "壹拾肆万柒仟捌佰圆整",
    "fiscal_code" : ""
  }
}
```

成功响应示例2

```
{
  "result" : {
    "used_vehicle_market_name" : "动车交易有限公司",
    "auction_org_bank_account" : "10468888888830",
    "seller_phone" : "13988888888",
    "buyer_id" : "AB37836SDWGDUOHDOEHD3",
    "machine_printed_number" : "00888858",
    "auction_org_id" : "AB37836SDWGDUOHDOEHD3",
    "auction_org_name" : "某拍卖行",
    "buyer_name" : "汽车贸易有限公司",
    "used_vehicle_market_id" : "91DEDD3782733DJ6R",
    "used_vehicle_market_phone" : "07588888888",
    "code" : "044008888817",
    "machine_number" : "4888888182",
    "used_vehicle_market_address" : "佛山市",
    "seller_address" : "广东省佛山市",
    "auction_org_address" : "广东省佛山市",
    "brand_model" : "吉利莫车型",
    "licence_plate_number" : "粤A88888",
```


版权所有 © 华为云计算技术有限公司

```
    "seller_id" : "AB37836SDWGDUOHDOEHD3",
    "machine_printed_code" : "10468888888830",
    "registration_number" : "1234EHWEGG8888830",
    "buyer_phone" : "13988888888",
    "vehicle_type" : "小型轿车",
    "drawer_name" : "张三",
    "total" : "￥20000.00",
    "total_chinese" : "贰万圆整",
    "issue_date" : "1987-12-27",
    "seller_name" : "李四",
    "vehicle_identification_number" : "L6T7804826379E3GG875",
    "auction_org_phone" : "13988888888",
    "used_vehicle_market_bank_account" : "中国农业银行",
    "buyer_address" : "佛山市",
    "dept_motor_vehicles" : "广东省佛山市车管所",
    "remark" : "免税开具",
    "reverse_issue" : false,
    "number" : "003753534",
    "fiscal_code" : "017807816386738245",
    "type" : "二手车销售统一发票"
  }
}
```

状态码：400

失败响应示例

```
{
    "error_code": "AIS.0103",
    "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入机动车销售发票图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeMvsInvoiceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
```

 版权所有 © 华为云计算技术有限公司

```
        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeMvsInvoiceRequest request = new RecognizeMvsInvoiceRequest();
        MvsInvoiceRequestBody body = new MvsInvoiceRequestBody();
        body.withType("auto");
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeMvsInvoiceResponse response = client.recognizeMvsInvoice(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入机动车销售发票图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeMvsInvoiceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeMvsInvoiceRequest request = new RecognizeMvsInvoiceRequest();
        MvsInvoiceRequestBody body = new MvsInvoiceRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeMvsInvoiceResponse response = client.recognizeMvsInvoice(request);
            System.out.println(response.toString());
```

 版权所有 © 华为云计算技术有限公司

```
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入机动车销售发票图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeMvsInvoiceRequest()
        request.body = MvsInvoiceRequestBody(
            type="auto",
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_mvs_invoice(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入机动车销售发票图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
```

 版权所有 © 华为云计算技术有限公司

```
environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeMvsInvoiceRequest()
        request.body = MvsInvoiceRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_mvs_invoice(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入机动车销售发票图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeMvsInvoiceRequest{}
    typeMvsInvoiceRequestBody:= "auto"
    imageMvsInvoiceRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.MvsInvoiceRequestBody{
        Type: &typeMvsInvoiceRequestBody,
        Image: &imageMvsInvoiceRequestBody,
    }
    response, err := client.RecognizeMvsInvoice(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
```

 版权所有 © 华为云计算技术有限公司

```
        fmt.Println(err)
    }
}
```

- 传入机动车销售发票图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeMvsInvoiceRequest{}
    urlMvsInvoiceRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.MvsInvoiceRequestBody{
        Url: &urlMvsInvoiceRequestBody,
    }
    response, err := client.RecognizeMvsInvoice(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

版权所有 © 华为云计算技术有限公司

错误码

错误码请参见**错误码**。

4.25 出租车发票识别

功能介绍

识别出租车发票中的文字信息，并以JSON格式返回识别的结构化结果。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-29 出租车发票示例图



说明

如果图片中包含多张卡证票据，请调用**智能分类识别**服务。

约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式的图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中发票区域有效占比超过25%，保证整张发票内容及其边缘包含在图像内。
- 支持少量扭曲，扭曲后图像中的发票长宽比与实际发票相差不超过10%。
- 支持图像中出租车发票任意角度的水平旋转。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

版权所有 © 华为云计算技术有限公司

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/taxi-invoice

表 4-205 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-206 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

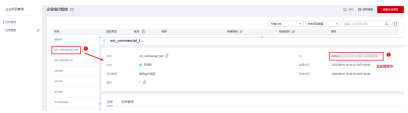
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见产品介绍。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-207 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片最短边不小于15px，最长边不超过4096px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
return_text_location	否	Boolean	识别到的文字块的区域位置信息。可选值包括： true：返回各个文字块区域 false：不返回各个文字块区域 如果无该参数，系统默认不返回文字块区域。如果输入参数不是Boolean类型，则会报非法参数错误。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-208 响应 Body 参数

参数	参数类型	描述
result	**TaxiInvoiceResult** object	识别结果。 调用失败时不返回此字段。

版权所有 © 华为云计算技术有限公司

表 4-209 TaxiInvoiceResult

参数	参数类型	描述
location	String	归属地区。
code	String	发票代码。
number	String	发票号码。
phone_number	String	电话（包括电话、监督电话）。
company	String	单位。
taxi_number	String	车号。
certificate_number	String	证号。
identification_number	String	识别编号。
date	String	开票日期。
boarding_time	String	上车时间。
alighting_time	String	下车时间。
time	String	时间(起止时间、上下车时间)。
unit_price	String	单价。
distance	String	总里程。
waiting_time	String	等候时间。
fare	String	金额。
fuel_oil_surcharge	String	燃油附加费。
call_service_surcharge	String	电调费（预约费）。
total	String	实收金额。
confidence	Object	字段的置信度，取值范围0~1。 置信度越大，本次识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于字段的准确率。
text_location	Object	对应所有在原图上识别到的字段位置信息，包含所有文字区域四个顶点的二维坐标（x,y）。采用图像坐标系，坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。

状态码： 400

版权所有 © 华为云计算技术有限公司

表 4-210 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，出租车发票识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/taxi-invoice”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 传入出租车发票图片的base64编码进行文字识别

```
POST  https://{endpoint}/v2/{project_id}/ocr/taxi-invoice
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
   "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入出租车发票图片的url进行文字识别

```
POST  https://{endpoint}/v2/{project_id}/ocr/taxi-invoice
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
   "url":"https://BucketName.obs.xxxx.com/ObjectName"
}
```

响应示例

状态码：200

成功响应示例

```
{
   "result": {
      "location": "上海市",
      "code": "130XXXXXXXX88",
      "number": "00XXXX08",
      "phone_number": "62XXX22",
      "company": "",
      "taxi_number": "GXXXXXX",
      "certificate_number": "3XXXX9",
      "identification_number": "",
      "date": "2019-03-15",
```


版权所有 © 华为云计算技术有限公司

```
        "boarding_time": "12:03",
        "alighting_time": "12:15",
        "time": "12:03-12:15",
        "unit_price": "￥2.50",
        "distance": "7.8",
        "waiting_time": "00:02:09",
        "fare": "￥29.00",
        "fuel_oil_surcharge": "￥0.00",
        "call_service_surcharge": "￥0.00",
        "total": "￥29.00",
        "confidence": {
            "location": 0.9977,
            "code": 0.9998,
            "number": 0.9976,
            "phone_number": 0.859,
            "company": 0,
            "taxi_number": 0.9607,
            "certificate_number": 0.9411,
            "identification_number": 0,
            "date": 0.9665,
            "boarding_time": 0.9847,
            "alighting_time": 0.9798,
            "time": 0.9798,
            "unit_price": 0.9174,
            "distance": 0.9731,
            "waiting_time": 0.9936,
            "fare": 0.9896,
            "fuel_oil_surcharge": 0.8433,
            "call_service_surcharge": 0.7843,
            "total": 0.9896
        }
    }
}
```


状态码：400

失败响应示例


```
{
    "error_code": "AIS.0103",
    "error_msg": "The image size does not meet the requirements."
}
```


SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入出租车发票图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeTaxiInvoiceSolution {
```

```
    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeTaxiInvoiceRequest request = new RecognizeTaxiInvoiceRequest();
        TaxiInvoiceRequestBody body = new TaxiInvoiceRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeTaxiInvoiceResponse response = client.recognizeTaxiInvoice(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入出租车发票图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeTaxiInvoiceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
```

```
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeTaxiInvoiceRequest request = new RecognizeTaxiInvoiceRequest();
        TaxiInvoiceRequestBody body = new TaxiInvoiceRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeTaxiInvoiceResponse response = client.recognizeTaxiInvoice(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入出租车发票图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeTaxiInvoiceRequest()
        request.body = TaxiInvoiceRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_taxi_invoice(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入出租车发票图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
```

 版权所有 © 华为云计算技术有限公司

```
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeTaxiInvoiceRequest()
        request.body = TaxiInvoiceRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_taxi_invoice(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入出租车发票图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeTaxiInvoiceRequest{}
    imageTaxiInvoiceRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
```

 版权所有 © 华为云计算技术有限公司

```
    request.Body = &model.TaxiInvoiceRequestBody{
        Image: &imageTaxiInvoiceRequestBody,
    }
    response, err := client.RecognizeTaxiInvoice(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入出租车发票图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeTaxiInvoiceRequest{}
    urlTaxiInvoiceRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.TaxiInvoiceRequestBody{
        Url: &urlTaxiInvoiceRequestBody,
    }
    response, err := client.RecognizeTaxiInvoice(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例

 版权所有 © 华为云计算技术有限公司

状态码	描述
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.26 火车票识别

功能介绍

识别火车票中的文字信息，并以JSON格式返回识别的结构化结果。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-30 火车票示例图

图 1-1 纸质火车票示例图



图 1-2 铁路电子客票示例图



图 1-3 退票费报销凭证示例图



图 1-4 纸质火车票退票凭证版示例图



说明

如果图片中包含多张卡证票据，请调用**智能分类识别**服务。

约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。

版权所有 © 华为云计算技术有限公司

- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中火车票区域有效占比超过25%。
- 支持图像中火车票任意角度的水平旋转。
- 支持少量扭曲，扭曲后图像中的火车票长宽比与实际火车票相差不超过10%。
- 支持识别铁路电子客票（含OFD格式）。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/train-ticket

表 4-211 路径参数

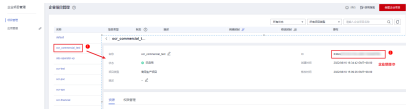
参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-212 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-213 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，要求 单个图片、OFD文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	该参数与image二选一。 单个图片、OFD文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
return_text_location	否	Boolean	识别到的文字块的区域位置信息。可选值包括： true：返回各个文字块区域 false：不返回各个文字块区域 如果无该参数，系统默认不返回文字块区域。如果输入参数不是Boolean类型，则会报非法参数错误。

版权所有 © 华为云计算技术有限公司

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-214 响应 Body 参数

参数	参数类型	描述
result	TrainTicketResult object	识别结果。 调用失败时不返回此字段。

表 4-215 TrainTicketResult

参数	参数类型	描述	归属
ticket_id	String	火车票左上角的车票ID。	纸质火车票/纸质火车票退票凭证
check_port	String	检票口信息。	纸质火车票/纸质火车票退票凭证
sale_method	String	售票方式。	纸质火车票/纸质火车票退票凭证
log_id	String	车票最下方的售票码。	纸质火车票/纸质火车票退票凭证
sale_location	String	车票售票地点。	纸质火车票/纸质火车票退票凭证
train_number	String	车次。	纸质火车票/纸质火车票退票凭证/铁路电子客票
departure_station	String	始发站。	纸质火车票/纸质火车票退票凭证/铁路电子客票

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述	归属
destination_station	String	终点站。	纸质火车票/纸质火车票退票凭证/铁路电子客票
departure_station_en	String	始发站拼音。	纸质火车票/纸质火车票退票凭证/铁路电子客票
destination_station_en	String	终点站拼音。	纸质火车票/纸质火车票退票凭证/铁路电子客票
departure_time	String	开车时间。	纸质火车票/纸质火车票退票凭证/铁路电子客票
seat_number	String	座位号。	纸质火车票/纸质火车票退票凭证/铁路电子客票
ticket_price	String	票价。	纸质火车票/纸质火车票退票凭证/铁路电子客票
seat_category	String	座位类别。	纸质火车票/纸质火车票退票凭证/铁路电子客票
ticket_changing	String	改签信息。	纸质火车票/纸质火车票退票凭证/铁路电子客票
id_number	String	车票持有人的身份证号。	纸质火车票/纸质火车票退票凭证/铁路电子客票
name	String	车票持有人姓名。	纸质火车票/纸质火车票退票凭证/铁路电子客票

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述	归属
air_conditioning	String	是否是空调车厢。	纸质火车票/纸质火车票退票凭证/铁路电子客票
invoice_style	String	类型。包含以下几种类型： • paper：纸质火车票 • electronic：铁路电子客票 • refund_old：退票费报销凭证 • refund_new：纸质火车票退票凭证 • refund_electronic：铁路电子客票退票凭证	共享字段
issue_date	String	开票时间。	铁路电子客票
discount_mark	String	打折标识。	铁路电子客票
serial_number	String	电子客票号。	铁路电子客票
tax_amount	String	税金价格。	铁路电子客票
tax_rate	String	税率。	铁路电子客票
original_invoice_number	String	原发票号码。	铁路电子客票
unified_social_credit_code	String	统一社会信用号码。	铁路电子客票
buyer_name	String	购买方名称。	铁路电子客票
total_amount_excluding_tax	String	不含税价格	铁路电子客票
invoice_number	String	发票号码。	铁路电子客票
seal_mark	Boolean	是否有印章，True表示有印章，False表示不含印章，字段默认为False。	退票费报销凭
title	String	标题。	退票费报销凭
area	String	地区。	退票费报销凭
receipt_number	String	收据编码。	退票费报销凭
amount_in_figures	String	小写票据金额。	退票费报销凭
amount_in_words	String	大写票据金额。	退票费报销凭

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述	归属
confidence	Object	字段的置信度，取值范围0~1。 置信度越大，本次识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于字段的准确率。	共享字段
text_location	Object	对应所有在原图上识别到的字段位置信息，包含所有文字区域四个顶点的二维坐标（x,y）。采用图像坐标系，坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。 **说明** 输入数据格式是OFD时，返回的字段坐标为空列表。	共享字段

状态码： 400

表 4-216 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，火车票识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/train-ticket”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 传入火车票图片的base64编码进行文字识别

```
POST  https://{endpoint}/v2/{project_id}/ocr/train-ticket
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
```


版权所有 © 华为云计算技术有限公司

```
    "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入火车票图片的url进行文字识别

```
POST  https://{endpoint}/v2/{project_id}/ocr/train-ticket
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "url":"https://BucketName.obs.xxxx.com/ObjectName"
}
```

响应示例

状态码：200

成功响应示例

```
{
  "result" : {
    "ticket_id" : "K090841",
    "check_port" : "B6",
    "train_number" : "G1138",
    "departure_station" : "长沙南站",
    "destination_station" : "武汉站",
    "departure_station_en" : "Changshanan",
    "destination_station_en" : "Wuhan",
    "departure_time" : "2017年12月12日20:39",
    "seat_number" : "03车03C号",
    "ticket_price" : "164.5元",
    "sale_method" : "网",
    "seat_category" : "二等座",
    "ticket_changing" : "No",
    "id_number" : "1402271998****0012",
    "name" : "xxx",
    "log_id" : "65818301991213K090841",
    "sale_location" : "长沙南售",
    "invoice_style" : "paper",
    "issue_date" : "",
    "discount_mark" : "",
    "serial_number" : "",
    "tax_amount" : "",
    "tax_rate" : "",
    "air_conditioning" : "",
    "original_invoice_number" : "",
    "unified_social_credit_code" : "",
    "buyer_name" : "",
    "total_amount_excluding_tax" : "",
    "invoice_number" : "",
    "seal_mark" : false,
    "title" : "",
    "area" : "",
    "receipt_number" : "",
    "amount_in_figures" : "",
    "amount_in_words" : "",
    "confidence" : {
      "ticket_id" : 0.9992,
      "check_port" : 0.9165,
      "train_number" : 0.9968,
      "starting_station" : 0.9982,
      "destination_station" : 0.9998,
      "starting_station_en" : 0.9254,
      "destination_station_en" : 0.9848,
      "starting_time" : 0.5141,
      "seat_number" : 0.9985,
      "ticket_rates" : 0.9977,
      "sale_method" : 0.9836,
```

 版权所有 © 华为云计算技术有限公司

```
    "seat_category" : 0.9958,
    "ticket_changing" : 0.9,
    "id_number" : 0.9986,
    "name" : 0.996,
    "log_id" : 0.9968,
    "ticket_office" : 0.9984,
    "invoice_style" : 0.0,
    "issue_date" : 0.0,
    "discount_mark" : 0.0,
    "serial_number" : 0.0,
    "tax_amount" : 0.0,
    "tax_rate" : 0.0,
    "air_conditioning" : 0.0,
    "original_invoice_number" : 0.0,
    "unified_social_credit_code" : 0.0,
    "buyer_name" : 0.0,
    "total_amount_excluding_tax" : 0.0,
    "invoice_number" : 0.0,
    "seal_mark" : 0.0,
    "title" : 0.0,
    "area" : 0.0,
    "receipt_number" : 0.0,
    "amount_in_figures" : 0.0,
    "amount_in_words" : 0.0
  }
 }
}
```


状态码：400

失败响应示例


```
{
   "error_code": "AIS.0103",
   "error_msg": "The image size does not meet the requirements."
}
```


SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入火车票图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeTrainTicketSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
```

```
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeTrainTicketRequest request = new RecognizeTrainTicketRequest();
        TrainTicketRequestBody body = new TrainTicketRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeTrainTicketResponse response = client.recognizeTrainTicket(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入火车票图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeTrainTicketSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeTrainTicketRequest request = new RecognizeTrainTicketRequest();
        TrainTicketRequestBody body = new TrainTicketRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
```

 版权所有 © 华为云计算技术有限公司

```
        request.withBody(body);
        try {
            RecognizeTrainTicketResponse response = client.recognizeTrainTicket(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入火车票图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeTrainTicketRequest()
        request.body = TrainTicketRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_train_ticket(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入火车票图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
```

 版权所有 © 华为云计算技术有限公司

```
environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeTrainTicketRequest()
        request.body = TrainTicketRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_train_ticket(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入火车票图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeTrainTicketRequest{}
    imageTrainTicketRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.TrainTicketRequestBody{
        Image: &imageTrainTicketRequestBody,
    }
    response, err := client.RecognizeTrainTicket(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
```

 版权所有 © 华为云计算技术有限公司

```
    } else {
        fmt.Println(err)
    }
}
```

- 传入火车票图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeTrainTicketRequest{}
    urlTrainTicketRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.TrainTicketRequestBody{
        Url: &urlTrainTicketRequestBody,
    }
    response, err := client.RecognizeTrainTicket(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

 版权所有 © 华为云计算技术有限公司

错误码

错误码请参见**错误码**。

4.27 定额发票识别

功能介绍

识别定额发票中的文字信息，并以JSON格式返回识别的结构化结果。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-31 定额发票示例图



说明

如果图片中包含多张卡证票据，请调用**智能分类识别**服务。

约束与限制

- 支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中发票区域有效占比超过25%。
- 支持图像中发票任意角度的水平旋转。
- 支持少量扭曲，扭曲后图像中的发票长宽比与实际发票相差不超过10%。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

版权所有 © 华为云计算技术有限公司

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/quota-invoice

表 4-217 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-218 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

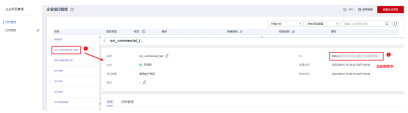
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-219 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url例如https://support.huaweicloud.com/api-ocr/zh-cn_image_0000001528088841.png • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（status code）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-220 响应 Body 参数

参数	参数类型	描述
result	**QuotaInvoiceResult** object	识别结果。 调用失败时不返回此字段。

表 4-221 QuotaInvoiceResult

参数	参数类型	描述
number	String	发票号码。
code	String	发票代码。
location	String	地址。
amount	String	发票金额。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
confidence	Object	字段的置信度，取值范围0~1。 置信度越大，本次识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于字段的准确率。

状态码： 400

表 4-222 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，定额发票识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/quota-invoice”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 传入定额发票图片的base64编码进行文字识别

```
POST  https://{endpoint}/v2/{project_id}/ocr/quota-invoice
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "image":"/9j/4AAQSkZJRgABAQAAAQABAAD/2wBDAAIBAQEBAQIBAQECAgICAgQDAgICAg..."
}
```

- 传入定额发票图片的url进行文字识别

```
POST  https://{endpoint}/v2/{project_id}/ocr/quota-invoice
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "url":"https://BucketName.obs.xxxx.com/ObjectName"
}
```


版权所有 © 华为云计算技术有限公司

响应示例

状态码：200

成功响应示例

```
{
   "result": {
      "amount": "￥50.00",
      "code": "244031XXXX11",
      "location": "广东省深圳市",
      "number": "91XXXX61",
      "confidence": {
         "amount": 0.9971,
         "code": 0.9908,
         "location": 0.9706,
         "number": 0.9943
      }
   }
}
```

状态码：400

失败响应示例

```
{
   "error_code": "AIS.0103",
   "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入定额发票图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeQuotaInvoiceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
```


版权所有 © 华为云计算技术有限公司

```
            .withAk(ak)
            .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeQuotaInvoiceRequest request = new RecognizeQuotaInvoiceRequest();
        QuotaInvoiceRequestBody body = new QuotaInvoiceRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeQuotaInvoiceResponse response = client.recognizeQuotaInvoice(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入定额发票图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeQuotaInvoiceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeQuotaInvoiceRequest request = new RecognizeQuotaInvoiceRequest();
        QuotaInvoiceRequestBody body = new QuotaInvoiceRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeQuotaInvoiceResponse response = client.recognizeQuotaInvoice(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
```

 版权所有 © 华为云计算技术有限公司

```
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入定额发票图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeQuotaInvoiceRequest()
        request.body = QuotaInvoiceRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_quota_invoice(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入定额发票图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
```

 版权所有 © 华为云计算技术有限公司

```
    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeQuotaInvoiceRequest()
        request.body = QuotaInvoiceRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_quota_invoice(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入定额发票图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeQuotaInvoiceRequest{}
    imageQuotaInvoiceRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.QuotaInvoiceRequestBody{
        Image: &imageQuotaInvoiceRequestBody,
    }
    response, err := client.RecognizeQuotaInvoice(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

- 传入定额发票图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeQuotaInvoiceRequest{}
    urlQuotaInvoiceRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.QuotaInvoiceRequestBody{
        Url: &urlQuotaInvoiceRequestBody,
    }
    response, err := client.RecognizeQuotaInvoice(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

版权所有 © 华为云计算技术有限公司

错误码

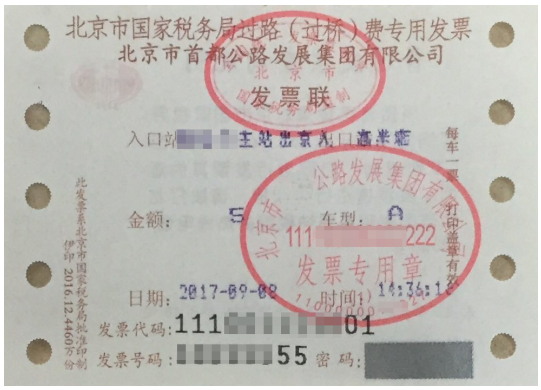
错误码请参见**错误码**。

4.28 车辆通行费发票识别

功能介绍

识别车辆通行费发票中的关键文字信息，并以JSON格式返回识别的结构化结果。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

图 4-32 车辆通行费发票示例图



说明

如果图片中包含多张卡证票据，请调用**智能分类识别**服务。

约束与限制

- 支持识别PNG、JPG、JPEG、BMP、TIFF格式的图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中发票区域有效占比超过25%。
- 支持图像中车辆通行费发票任意角度的水平旋转。
- 目前只支持中华人民共和国境内的车辆通行费发票识别。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

版权所有 © 华为云计算技术有限公司

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/toll-invoice

表 4-223 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-224 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

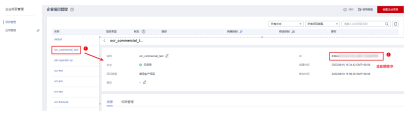
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-225 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
return_text_location	否	Boolean	识别到的文字块的区域位置信息。可选值包括： • true：返回各个文字块区域 • false：不返回各个文字块区域 如果无该参数,系统默认不返回文字块区域。如果输入参数不是Boolean类型,则会报非法参数错误。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-226 响应 Body 参数

参数	参数类型	描述
result	**TollInvoiceResult** object	识别结果。 调用失败时不返回此字段。

版权所有 © 华为云计算技术有限公司

表 4-227 TollInvoiceResult

参数	参数类型	描述
code	String	发票代码。
number	String	发票号码。
entry	String	入口。
exit	String	出口。
amount	String	收费金额。
cashier	String	收费员。
vehicle_type	String	车辆类型。
date	String	日期。
time	String	时间。
confidence	Object	字段的置信度，取值范围0~1。 置信度越大，本次识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于字段的准确率。
text_location	Object	对应所有在原图上识别到的字段位置信息，包含所有文字区域四个顶点的二维坐标（x,y）。采用图像坐标系，坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。

状态码： 400

表 4-228 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

版权所有 © 华为云计算技术有限公司

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，车辆通行费发票识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/toll-invoice”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 传入车辆通行费发票图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/toll-invoice
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
   "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAgAB..."
}
```

- 传入车辆通行费发票图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/toll-invoice
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
   "url":"https://BucketName.obs.xxxx.com/ObjectName"
}
```

响应示例

状态码：200

成功响应示例

```
{
   "result": {
      "code": "11XXXXXXXX88",
      "number": "56XXXXX08",
      "entry": "吴各庄南线出京",
      "exit": "顺义城站",
      "cashier": "213",
      "vehicle_type": "A",
      "date": "2017-12-04",
      "time": "18:45:10",
      "amount": "15",
      "confidence": {
         "code": 0.9957,
         "number": 0.9497,
         "exit": 0.9662,
         "cashier": 0.987,
         "vehicle_type": 0.9639,
         "entry": 0.9985,
         "date": 0.9974,
         "time": 0.9984,
         "amount": 0.9943
      }
   }
}
```

状态码：400

版权所有 © 华为云计算技术有限公司

失败响应示例

```
{
   "error_code": "AIS.0103",
   "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入定额发票图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeTollInvoiceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeTollInvoiceRequest request = new RecognizeTollInvoiceRequest();
        TollInvoiceRequestBody body = new TollInvoiceRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeTollInvoiceResponse response = client.recognizeTollInvoice(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
```


版权所有 © 华为云计算技术有限公司

```
        }
    }
}
```

- 传入定额发票图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeTollInvoiceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeTollInvoiceRequest request = new RecognizeTollInvoiceRequest();
        TollInvoiceRequestBody body = new TollInvoiceRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeTollInvoiceResponse response = client.recognizeTollInvoice(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入定额发票图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
```

 版权所有 © 华为云计算技术有限公司

```
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeTollInvoiceRequest()
        request.body = TollInvoiceRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_toll_invoice(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入定额发票图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeTollInvoiceRequest()
        request.body = TollInvoiceRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_toll_invoice(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

 版权所有 © 华为云计算技术有限公司

Go

- 传入定额发票图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeTollInvoiceRequest{}
    imageTollInvoiceRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.TollInvoiceRequestBody{
        Image: &imageTollInvoiceRequestBody,
    }
    response, err := client.RecognizeTollInvoice(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入定额发票图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
```

 版权所有 © 华为云计算技术有限公司

```
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeTollInvoiceRequest{}
    urlTollInvoiceRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.TollInvoiceRequestBody{
        Url: &urlTollInvoiceRequestBody,
    }
    response, err := client.RecognizeTollInvoice(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.29 飞机行程单识别

功能介绍

识别飞机行程单中的文字信息，并以JSON格式返回识别的结构化结果。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

版权所有 © 华为云计算技术有限公司

图 4-33 飞机行程单示例图



说明

如果图片中包含多张卡证票据，请调用**智能分类识别**服务。

约束与限制

- 支持识别PNG、JPG、JPEG、BMP、TIFF格式的图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 支持图像中行程单任意角度的水平旋转。
- 能处理反光、暗光等干扰的图片但影响识别精度。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/flight-itinerary

版权所有 © 华为云计算技术有限公司

表 4-229 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-230 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

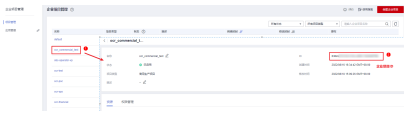
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-231 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	该参数与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
url	否	String	该参数与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（ status code ）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-232 响应 Body 参数

参数	参数类型	描述
result	**FlightItineraryResult** object	识别结果。 调用失败时不返回此字段。

表 4-233 FlightItineraryResult

参数	参数类型	描述
serial_number	String	印刷序号。
passenger_name	String	旅客姓名。
id_number	String	有效身份证件号码。
endorsements_restrictions	String	备注。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
order_number	String	订单号。
fare	String	票价。
caac_development_fund	String	民航（CAAC)发展基金。
fuel_surcharge	String	燃油附加费。
other_taxes	String	其他税费。
total	String	合计。
e_ticket_number	String	电子客票号码。
check_code	String	验证码。
reference_information	String	提示信息。
insurance	String	保险费。
agent_code	String	销售单位代号。
issue_organization	String	填开单位。
issue_date	String	填开日期。
tax	String	增值税税额。
tax_rate	String	增值税税率。
buyer_name	String	购买方名称。
buyer_id	String	购买方纳税人识别号。
number	String	发票号码。
international_flag	String	国内国际标签。
issue_status	String	开具状态。
gp_number	String	gp单号。
itinerary_list	Array of **ItineraryList** objects	机票行程列表。
confidence	Object	字段的置信度，取值范围0~1。 置信度越大，本次识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于字段的准确率。

版权所有 © 华为云计算技术有限公司

表 4-234 ItineraryList

参数	参数类型	描述
departure_station	String	始发站。
destination_station	String	目的站。
carrier	String	承运人。
flight	String	航班号。
cabin_class	String	座位等级。
date	String	日期。
time	String	时间。
fare_basis	String	客票类别。
effective_date	String	客票生效日期。
expiry_date	String	有效截止日期。
baggage_allowance	String	免费行李。

状态码： 400

表 4-235 响应 Body 参数

参数	参数类型	说明
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，飞机行程单识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/flight-itinerary”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

版权所有 © 华为云计算技术有限公司

- 传入飞机行程单图片的base64编码进行文字识别

```
POST  https://{endpoint}/v2/{project_id}/ocr/flight-itinerary
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
   "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入飞机行程单图片的url进行文字识别

```
POST  https://{endpoint}/v2/{project_id}/ocr/flight-itinerary
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
   "url":"https://BucketName.obs.xxxx.com/ObjectName"
}
```

响应示例

状态码：200

成功响应示例

```
{
  "result": {
      "serial_number": "12XXXXXXX22",
      "passenger_name": "WANG/SAN",
      "id_number": "GXXXXXX08",
      "endorsements_restrictions": "Q/不得签转/退改收费",
      "fare": "CNY 2340.00",
      "caac_development_fund": "CN 100.00",
      "fuel_surcharge": "XXXX",
      "other_taxes": "",
      "total": "CNY 2440.00",
      "e_ticket_number": "73XXXXXXXX008",
      "check_code": "3XXX",
      "reference_information": "白云机场起飞前45分停办值机",
      "insurance": "XXX",
      "agent_code": "SXXXXX 00XXXX66",
      "issue_organization": "XX航空股份有限公司",
      "issue_date": "2017-12-27",
      "tax" : "",
      "tax_rate" : "",
      "buyer_name" : "",
      "buyer_id" : "",
      "number" : "",
      "international_flag" : "",
      "issue_status" : "",
      "gp_number" : "",
      "itinerary_list": [
         {
            "departure_station": "T2上海虹桥",
            "destination_station": "广州",
            "carrier": "X航",
            "flight": "XXXXXX",
            "cabin_class": "B",
            "date": "2017-12-27",
            "time": "13:00",
            "fare_basis": "B",
            "effective_date": "",
            "expiry_date": "",
            "baggage_allowance": "20K"
         },
```

 版权所有 © 华为云计算技术有限公司

```
        {
            "departure_station": "广州",
            "destination_station": "上海虹桥",
            "carrier": "XXXX",
            "flight": "XXXXXX",
            "cabin_class": "B",
            "date": "2017-12-29",
            "time": "16:30",
            "fare_basis": "B",
            "effective_date": "",
            "expiry_date": "",
            "baggage_allowance": "20K"
        }
    ],
    "confidence":{
            "serial_number": 0.9993,
            "passenger_name": 0.9851,
            "id_number": 0.9836,
            "endorsements_restrictions": 0.998,
            "fare": 0.9882,
            "caac_development_fund": 0.993,
            "fuel_surcharge": 0.9968,
            "other_taxes": 0,
            "total": 0.9999,
            "e_ticket_number": 0.9631,
            "check_code": 0.9985,
            "reference_information": 0.9982,
            "insurance": 0.7749,
            "agent_code": 0.9913,
            "issue_organization": 0.7142,
            "issue_date": 0.9936,
            "tax" : "0,",
            "tax_rate" : "0,",
            "buyer_name" : "0,",
            "buyer_id" : "0,",
            "number" : "0,",
            "international_flag" : "0,",
            "issue_status" : "0,",
            "itinerary_list": [
                {
                    "departure_station": 0.9987,
                    "destination_station": 0.9577,
                    "carrier": 0.9677,
                    "flight": 0.9957,
                    "cabin_class": 0.8744,
                    "date": 0.9993,
                    "time": 0.9978,
                    "fare_basis": 0.4006,
                    "effective_date": 0,
                    "expiry_date": 0,
                    "baggage_allowance": 0.9897
                },
                {
                    "departure_station": 0.9995,
                    "destination_station": 0.9844,
                    "carrier": 0.9763,
                    "flight": 0.7453,
                    "cabin_class": 0.4239,
                    "date": 0.76,
                    "time": 0.7311,
                    "fare_basis": 0.4353,
                    "effective_date": 0,
                    "expiry_date": 0,
                    "baggage_allowance": 0.5265
                }
            ]
    }
  }
}
```


 版权所有 © 华为云计算技术有限公司

状态码：400

失败响应示例

```
{
    "error_code": "AIS.0103",
    "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入飞机行程单图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeFlightItinerarySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeFlightItineraryRequest request = new RecognizeFlightItineraryRequest();
        FlightItineraryRequestBody body = new FlightItineraryRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeFlightItineraryResponse response = client.recognizeFlightItinerary(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
```


版权所有 © 华为云计算技术有限公司

```
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入飞机行程单图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeFlightItinerarySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeFlightItineraryRequest request = new RecognizeFlightItineraryRequest();
        FlightItineraryRequestBody body = new FlightItineraryRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeFlightItineraryResponse response = client.recognizeFlightItinerary(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入飞机行程单图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *
```

 版权所有 © 华为云计算技术有限公司

```
if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeFlightItineraryRequest()
        request.body = FlightItineraryRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_flight_itinerary(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入飞机行程单图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeFlightItineraryRequest()
        request.body = FlightItineraryRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_flight_itinerary(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

 版权所有 © 华为云计算技术有限公司

Go

- 传入飞机行程单图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeFlightItineraryRequest{}
    imageFlightItineraryRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.FlightItineraryRequestBody{
        Image: &imageFlightItineraryRequestBody,
    }
    response, err := client.RecognizeFlightItinerary(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入飞机行程单图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
```


版权所有 © 华为云计算技术有限公司

```
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeFlightItineraryRequest{}
    urlFlightItineraryRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.FlightItineraryRequestBody{
        Url: &urlFlightItineraryRequestBody,
    }
    response, err := client.RecognizeFlightItinerary(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.30 电子面单识别

说明

OCR电子面单识别已于2025.09.04正式下架，为避免影响您目前的业务，请您及时进行相关迁移工作，详见**下架公告**。

功能介绍

识别用户上传的电子面单图片中的文字内容，并将识别的结果以JSON格式返回给用户。

版权所有 © 华为云计算技术有限公司

图 4-34 电子面单示例图



约束与限制

- 电子面单支持的版式以样例为准。
- 支持识别JPG、PNG、BMP、TIFF格式图片。单个图片对应的Base64编码不超过10MB。
- 图像各边的像素大小在15px到8192px之间。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**开通服务。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/waybill-electronic

版权所有 © 华为云计算技术有限公司

表 4-236 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-237 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

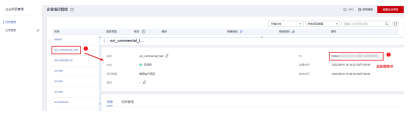
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-238 请求 Body 参数

参数	是否必选	参数类型	描述
image	否	String	与url二选一 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图像各边的像素大小在15到8192px之间。支持JPG/PNG/BMP/TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
url	否	String	与image二选一，url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的URL路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。

响应参数

状态码： 200

表 4-239 响应 Body 参数

参数	参数类型	描述
result	**WaybillElectronicResult** object	识别结果。 调用失败时无此字段。

表 4-240 WaybillElectronicResult

参数	参数类型	描述
code	String	三段码。
receiver_name	String	收件人姓名。
receiver_phone	String	收件人电话。
receiver_address	String	收件人地址。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
sender_name	String	寄件人姓名。
sender_phone	String	寄件人电话。
sender_address	String	寄件人地址。
waybill_number	String	条形码运单号。
confidence	Object	相关字段的置信度信息，取值范围0~1。 置信度越大，表示本次识别的对应字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于对应字段的准确率。

状态码： 400

表 4-241 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

请求示例

- 传入快递单图片的base64编码进行文字识别

```
POST https://ocr.cn-north-4.xxxx.com/v1.0/ocr/waybill-electronic

 Request Header:
 Content-Type: application/json
 X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...

 Request Body:
 { "image":"/9j/4AAQSkZJRgABAgEASABIAAD/
4RFZRXhpZgAATU0AKgAAAAgABwESAAMAAAABAAEAAAEaAAUAAAABAAAAYgEbAAUAAAABAAAAag
EoAAMAAAABAAIAAAExAAIAAAAcAAAAcgEyAAIAAAAUAAAAjodpAAQAAAABAAAApAAAANAACvyAAA
AnEAAK/
IAAACcQQWRvYmUgUGhvdG9zaG9wIENTMyBXaW5kb3dzADIwMTc6MTA6MjAgMTA6NDU6MzYAAAAA
AA6ABAAMAAAAB//
8AAKACAAQAAAABAAALIKADAAQAAAABAAAGQAAAAAAAAAAGAQMAAwAAAAEABgAAARoABQAAA
AEAAAEeARsABQAAAAEAAAEmASgAAwAAAAEAAgAAAgEABAAAAAEAAAEuAgIABAAAAAEAABAj..."
 }
```

- 传入快递单图片的url进行文字识别

```
POST https://ocr.cn-north-4.myhuaweicloud.com/v1.0/ocr/waybill-electronic

 Request Header:
 Content-Type: application/json
```


版权所有 © 华为云计算技术有限公司

```
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...

Request Body:
{
    "url":"https://BucketName.obs.myhuaweicloud.com/ObjectName"
}
```

响应示例

状态码： 200

成功响应示例

```
{
  "result" : {
    "code" : "12345678901",
    "receiver_name" : "张三",
    "receiver_phone" : "12345678901",
    "receiver_address" : "深圳市龙岗区坂田街道华为基地",
    "sender_name" : "生活用品用品专营店",
    "sender_phone" : "1234567890",
    "sender_address" : "httpe-//",
    "waybill_number" : "",
    "confidence" : {
      "code" : 0.9899,
      "receiver_name" : 0.9898,
      "receiver_phone" : 0.9899,
      "receiver_address" : 0.9897,
      "sender_name" : 0.9896,
      "sender_phone" : 0.9899,
      "sender_address" : 0.8855,
      "waybill_number" : 0.0
    }
  }
}
```

状态码： 400

失败响应示例

```
{
  "error_code" : "AIS.0103",
  "error_msg" : "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入快递单图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
```

 版权所有 © 华为云计算技术有限公司

```
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeWaybillElectronicSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeWaybillElectronicRequest request = new RecognizeWaybillElectronicRequest();
        WaybillElectronicRequestBody body = new WaybillElectronicRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeWaybillElectronicResponse response = client.recognizeWaybillElectronic(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入快递单图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeWaybillElectronicSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
```

```
            .withAk(ak)
            .withSk(sk);

    OcrClient client = OcrClient.newBuilder()
            .withCredential(auth)
            .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
            .build();
    RecognizeWaybillElectronicRequest request = new RecognizeWaybillElectronicRequest();
    WaybillElectronicRequestBody body = new WaybillElectronicRequestBody();
    body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
    request.withBody(body);
    try {
        RecognizeWaybillElectronicResponse response = client.recognizeWaybillElectronic(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
  }
}
```

Python

- 传入快递单图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeWaybillElectronicRequest()
        request.body = WaybillElectronicRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_waybill_electronic(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入快递单图片的url进行文字识别

 版权所有 © 华为云计算技术有限公司

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeWaybillElectronicRequest()
        request.body = WaybillElectronicRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_waybill_electronic(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入快递单图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
```

 版权所有 © 华为云计算技术有限公司

```
        Build())

    request := &model.RecognizeWaybillElectronicRequest{}
    imageWaybillElectronicRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/
4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.WaybillElectronicRequestBody{
        Image: &imageWaybillElectronicRequestBody,
    }
    response, err := client.RecognizeWaybillElectronic(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入快递单图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeWaybillElectronicRequest{}
    urlWaybillElectronicRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.WaybillElectronicRequestBody{
        Url: &urlWaybillElectronicRequestBody,
    }
    response, err := client.RecognizeWaybillElectronic(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.31 保险单识别

功能介绍

识别保险单图片上的文字信息，并将识别的结构化结果返回给用户。支持对多板式保险单的扫描图片及手机照片进行结构化信息提取。

约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 图像各边的像素在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中保险单区域有效占比超过70%，保证整张保险单及其边缘包含在图像内。
- 支持图像中保险单旋转、支持少量扭曲。
- 能处理暗光等干扰的图片但影响识别精度。
- 覆盖常见保险公司的常见保单版式，由于即使是同一家保险公司，保险种类也繁多而且都在动态变化，实际支持情况请以实际测试效果为准。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/insurance-policy

版权所有 © 华为云计算技术有限公司

表 4-242 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-243 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

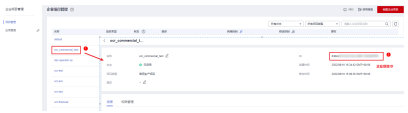
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-244 请求 Body 参数

参数	是否必选	参数类型	描述
image	否	String	与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
url	否	String	与image二选一 url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的URL路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
detect_direction	否	Boolean	是否校正图片的倾斜角度，可选值如下。 • true：校正图片的倾斜角度 • false：不校正图片的倾斜角度 支持任意角度的校正，未传入该参数时默认为“false”。 待识别图片如果存在倾斜，建议将此参数设置为“true”。

响应参数

状态码： 200

表 4-245 响应 Body 参数

参数	参数类型	描述
result	**InsurancePolicyResult** object	识别结果。 调用失败时不返回此字段。

版权所有 © 华为云计算技术有限公司

表 4-246 InsurancePolicyResult

参数	参数类型	描述
bank_name	String	发卡行。
bill_number	**InsurancePolicyDetail** object	保险单号（没有单号时，返回为空）。
company	**InsurancePolicyDetail** object	保险公司。
effective_date	**InsurancePolicyDetail** object	保险单生效日期。
applicant_name	**InsurancePolicyDetail** object	投保人姓名。
applicant_sex	**InsurancePolicyDetail** object	投保人性别。
applicant_birthday	**InsurancePolicyDetail** object	投保人出生日期。
applicant_id_type	**InsurancePolicyDetail** object	投保人证件类型。
applicant_id_number	**InsurancePolicyDetail** object	投保人证件号。
insurant_list	Array of **InsurantItem** objects	被保人列表（第一个默认为主被保人）。
beneficiary_list	Array of **BeneficiaryItem** objects	受益人列表。
insurance_list	Array of **InsuranceItem** objects	保险项目信息列表。

表 4-247 InsurancePolicyDetail

参数	参数类型	描述
words	String	对应识别出的文本内容。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
location	Array<Array<Integer>>	对应识别出的四个顶点坐标。

表 4-248 InsurantItem

参数	参数类型	描述
insurant_name	**InsurancePolicyDetail** object	被保人姓名。
insurant_sex	**InsurancePolicyDetail** object	被保人性别。
insurant_birthday	**InsurancePolicyDetail** object	被保人出生日期。
insurant_id_type	**InsurancePolicyDetail** object	被保人证件类型。
insurant_id_number	**InsurancePolicyDetail** object	被保人证件号码。

表 4-249 BeneficiaryItem

参数	参数类型	描述
beneficiary_name	**InsurancePolicyDetail** object	受益人姓名。
beneficiary_type	**InsurancePolicyDetail** object	受益人类型。
beneficiary_order	**InsurancePolicyDetail** object	受益顺序。
beneficiary_share	**InsurancePolicyDetail** object	受益比例/受益顺序。

版权所有 © 华为云计算技术有限公司

表 4-250 InsuranceItem

参数	参数类型	描述
insurance_name	**InsurancePolicyDetail** object	产品名称。
insurance_period	**InsurancePolicyDetail** object	保险期限。
insurance_amount	**InsurancePolicyDetail** object	保险金额。
payment_frequency	**InsurancePolicyDetail** object	交费频率。
payment_period	**InsurancePolicyDetail** object	交费期间。
payment_amount	**InsurancePolicyDetail** object	每期交费金额。

状态码： 400

表 4-251 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- Endpoint即调用API的请求地址，不同服务不同区域的Endpoint不同，具体请参见**终端节点**。 例如，保险单识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/insurance-policy”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入保险单图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/insurance-policy
Request Header:
```


版权所有 © 华为云计算技术有限公司

```
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
  "image" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入保险单图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/insurance-policy
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
  "url" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

响应示例

状态码： 200

成功响应示例

```
{
  "result" : {
    "bill_number" : {
      "words" : "192365331963636",
      "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
    },
    "company" : {
      "words" : "华夏人寿保险股份有限公司",
      "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
    },
    "effective_date" : {
      "words" : "见特别约定",
      "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
    },
    "applicant_name" : {
      "words" : "杨建国",
      "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
    },
    "applicant_id_type" : {
      "words" : "身份证",
      "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
    },
    "applicant_id_number" : {
      "words" : "214631270006529342",
      "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
    },
    "insurant_list" : [ {
      "insurant_name" : {
        "words" : "惠秀梅",
        "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
      },
      "insurant_sex" : {
        "words" : "女",
        "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
      },
      "insurant_id_type" : {
        "words" : "身份证",
        "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
      },
      "insurant_id_number" : {
        "words" : "XXXXXXXXXXXXXXXX",
        "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
      }
    } ],
```

 版权所有 © 华为云计算技术有限公司

```
  "beneficiary_list" : [ {
    "beneficiary_name" : {
      "words" : "XXX",
      "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
    },
    "beneficiary_type" : {
      "words" : "生存保险金",
      "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
    },
    "beneficiary_order" : {
      "words" : "第一顺序",
      "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
    },
    "beneficiary_share" : null
  } ],
  "insurance_list" : [ {
    "insurance_name" : {
      "words" : "华夏黄金甲两全保险（分红型）",
      "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
    },
    "insurance_period" : {
      "words" : "2025年7月17日零时",
      "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
    },
    "insurance_amount" : {
      "words" : "RMB50000.00",
      "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
    },
    "payment_frequency" : {
      "words" : "月交",
      "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
    },
    "payment_period" : {
      "words" : "2017年7月17日零时",
      "location" : [ [ 10, 10 ], [ 20, 25 ], [ 20, 35 ], [ 10, 35 ] ]
    }
  } ]
 }
}
```


状态码： 400

失败响应示例


```
{
  "error_code" : "AIS.0103",
  "error_msg" : "The image size does not meet the requirements"
}
```


SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入保险单图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
```

```
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeInsurancePolicySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeInsurancePolicyRequest request = new RecognizeInsurancePolicyRequest();
        InsurancePolicyRequestBody body = new InsurancePolicyRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeInsurancePolicyResponse response = client.recognizeInsurancePolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入保险单图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeInsurancePolicySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
```

```
        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeInsurancePolicyRequest request = new RecognizeInsurancePolicyRequest();
        InsurancePolicyRequestBody body = new InsurancePolicyRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeInsurancePolicyResponse response = client.recognizeInsurancePolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入保险单图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeInsurancePolicyRequest()
        request.body = InsurancePolicyRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_insurance_policy(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

 版权所有 © 华为云计算技术有限公司

- 传入保险单图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeInsurancePolicyRequest()
        request.body = InsurancePolicyRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_insurance_policy(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入保险单图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
```

 版权所有 © 华为云计算技术有限公司

```
            WithCredential(auth).
            Build())

    request := &model.RecognizeInsurancePolicyRequest{}
    imageInsurancePolicyRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/
4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.InsurancePolicyRequestBody{
        Image: &imageInsurancePolicyRequestBody,
    }
    response, err := client.RecognizeInsurancePolicy(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入保险单图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeInsurancePolicyRequest{}
    urlInsurancePolicyRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.InsurancePolicyRequestBody{
        Url: &urlInsurancePolicyRequestBody,
    }
    response, err := client.RecognizeInsurancePolicy(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

 版权所有 © 华为云计算技术有限公司

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.32 财务报表识别

功能介绍

识别用户上传的表格图片中的文字内容，并将识别的结果返回给用户。

约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中识别区域有效占比超过80%，保证整张表格及其边缘包含在图像内。
- 支持图像任意角度的水平旋转。
- 目前不支持复杂背景（如户外自然场景、防伪水印等）和表格线扭曲图像的文字识别。

调用方法

请参见**如何调用API**。

前提条件

在使用财务报表识别之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/financial-statement

版权所有 © 华为云计算技术有限公司

表 4-252 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-253 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

表 4-254 请求 Body 参数

参数	是否必选	参数类型	描述
image	否	String	与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
url	否	String	与image二选一。 url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的URL路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
return_text_location	否	Boolean	返回文本块坐标及单元格坐标信息，可选值包括： • true：返回文本块和单元格坐标; • false：不返回。 未传入该参数时默认为false，即不返回。
return_confidence	否	Boolean	返回字段识别置信度，小数点后四位。可选值包括： • true：返回字段置信度; • false：不返回。 未传入该参数时默认为false，即不返回字段置信度。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
return_excel	否	Boolean	是否返回表格转换Microsoft Excel的Base64编码字段。可选值包括： ● true：返回’excel’字段，表示xlsx格式的表格识别结果的Base64编码; ● false：不返回。 对返回的Excel编码，可用Python函数 base64.b64decode解码后保存为xlsx文件。
return_table_location	否	Boolean	返回表格坐标，可选值包括： ● true：返回表格坐标; ● false：不返回。 未传入该参数时默认为false，即不返回。
return_image_size	否	Boolean	返回矫正后的图像大小，可选值包括： ● true：返回矫正图像大小; ● false：不返回。 未传入该参数时默认为false，即不返回。
return_rectification_matrix	否	Boolean	透视变换矩阵，可选值包括： ● true：返回透视变换矩阵; ● false：不返回。 **说明** 未传入该参数时默认为false，即不返回透视变换矩阵。

响应参数

状态码： 200

表 4-255 响应 Body 参数

参数	参数类型	描述
result	**FinancialStatementResult** object	识别结果。 调用失败时不返回此字段。

版权所有 © 华为云计算技术有限公司

表 4-256 FinancialStatementResult

参数	参数类型	描述
words_region_count	Integer	识别出来的表格、文本区域个数。
words_region_list	Array of **FinancialStatementWordsRegionList** objects	返回的表格、文本区域列表。输出顺序从左到右，从上到下。
excel	String	表格图像转换为excel的Base64编码，图像中的文字和表格按位置写入excel，可编辑。对返回的excel编码，可用base64.b64decode解码并保存为xlsx文件。
image_size	**image_size** object	矫正后图像的高宽信息。
rectification_matrix	Array<Array<Number>>	返回透视变换矩阵。

表 4-257 FinancialStatementWordsRegionList

参数	参数类型	描述
type	String	区域属性：文本或表格。
words_block_count	Float	检测到的文字块数目。对文本区，文字块以文本字段为单位；对表格区，文字块以单元格内所有字段为单位。
table_location	Array<Array<Integer>>	表格位置信息，列表形式，分别表示表格4个顶点的x, y坐标;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。
words_block_list	Array of **FinancialStatementWordsBlockList** objects	区域内文字块列表，输出顺序从左到右，从上到下。

表 4-258 FinancialStatementWordsBlockList

参数	参数类型	描述
words	String	文字块内容。当入参"return_text_location"为false时，每个单元格返回一个文本值，不同行文本由换行符 "\n" 拼接。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
location	Array<Array<Integer>>	文字块位置信息，列表形式，分别表示文字块4个顶点的x, y坐标;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。
confidence	Float	文字块识别结果置信度信息，置信度越大，表示本次识别的对应字段的可靠性越大，在统计意义上，置信度越大正确率越高。注：置信度由算法给出，其不直接等价于对应字段的精度。
rows	Array of integers	单元格行信息，列表形式。多个连续值表示单元格垮多行。
columns	Array of integers	单元格列信息，列表形式。多个连续值表示单元格垮多列。
cell_location	Array<Array<Integer>>	单元格位置信息，列表形式，分别表示单元格4个顶点的x, y坐标;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。

表 4-259 image_size

参数	参数类型	描述
height	Integer	矫正后图像的高。
width	Integer	矫正后图像的宽。

状态码： 400

表 4-260 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

版权所有 © 华为云计算技术有限公司

请求示例

说明

- Endpoint即调用API的请求地址，不同服务不同区域的Endpoint不同，具体请参见**终端节点**。 例如，财务报表识别服务部署在“华东-上海一”区域的“endpoint”为“ocr.cn-east-3.myhuaweicloud.com”或“ocr.cn-east-3.myhuaweicloud.cn”，请求URL为“https://ocr.cn-east-3.myhuaweicloud.com/v2/{project_id}/ocr/financial-statement”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入财务报表图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/financial-statement
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
  "image" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入财务报表图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/financial-statement
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
  "url" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

响应示例

状态码： 200

成功响应示例

```
{
  "result" : {
    "words_region_count" : 2,
    "words_region_list" : [ {
      "type" : "text",
      "words_block_count" : 1,
      "words_block_list" : [ {
        "words" : "文字区域识别文字块1",
        "confidence" : 0.999
      } ]
    }, {
      "type" : "table",
      "table_location" : [ [ 120, 106 ], [ 200, 106 ], [ 200, 351 ], [ 120, 351 ] ],
      "words_block_count" : 2,
      "words_block_list" : [ {
        "words" : "负债和所有者权益（或股东权益）",
        "confidence" : 0.9963,
        "rows" : [ 0 ],
        "columns" : [ 0 ]
      }, {
        "words" : "行次",
        "confidence" : 0.9999,
        "rows" : [ 0 ],
        "columns" : [ 1 ]
      } ]
    } ],
    "excel" : "AQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKg…",
    "image_size" : {
```


版权所有 © 华为云计算技术有限公司

```
    "height" : 2501,
    "width" : 1701
  }
 }
}
```

状态码： 400

失败响应示例

```
{
 "error_code" : "AIS.0103",
 "error_msg" : "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入财务报表图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeFinancialStatementSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeFinancialStatementRequest request = new RecognizeFinancialStatementRequest();
        FinancialStatementRequestBody body = new FinancialStatementRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeFinancialStatementResponse response = client.recognizeFinancialStatement(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入财务报表图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeFinancialStatementSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeFinancialStatementRequest request = new RecognizeFinancialStatementRequest();
        FinancialStatementRequestBody body = new FinancialStatementRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeFinancialStatementResponse response = client.recognizeFinancialStatement(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

 版权所有 © 华为云计算技术有限公司

Python

- 传入财务报表图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeFinancialStatementRequest()
        request.body = FinancialStatementRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_financial_statement(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入财务报表图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeFinancialStatementRequest()
        request.body = FinancialStatementRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
```


版权所有 © 华为云计算技术有限公司

```
        response = client.recognize_financial_statement(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入财务报表图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeFinancialStatementRequest{}
    imageFinancialStatementRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.FinancialStatementRequestBody{
        Image: &imageFinancialStatementRequestBody,
    }
    response, err := client.RecognizeFinancialStatement(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入财务报表图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

 版权所有 © 华为云计算技术有限公司

```
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeFinancialStatementRequest{}
    urlFinancialStatementRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.FinancialStatementRequestBody{
        Url: &urlFinancialStatementRequestBody,
    }
    response, err := client.RecognizeFinancialStatement(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.33 承兑汇票识别

功能介绍

识别承兑汇票识别中的关键字段，并以json格式返回结构化结果。

 版权所有 © 华为云计算技术有限公司

约束与限制

- 只支持中英文。
- 只支持识别PNG、JPG、JPEG、BMP、GIF、TIFF、WEBP、PCX、ICO、PSD格式的图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中识别区域有效占比超过80%，保证所有文字及其边缘包含在图像内。
- 不支持角度旋转的图片识别。
- 支持电子承兑汇票，对于拍照场景、复杂背景支持欠佳。
- 文字识别服务属于公有云服务，线上用户资源共享，如果需要多并发请求，请提前**联系我们**。

调用方法

请参见**如何调用API**。

前提条件

在使用承兑汇票识别之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/acceptance-bill

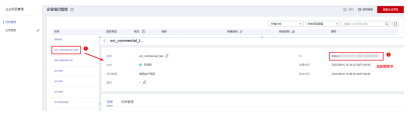
表 4-261 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

版权所有 © 华为云计算技术有限公司

请求参数

表 4-262 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-263 请求 Body 参数

参数	是否必选	参数类型	描述
image	否	String	与url二选一。 图像数据，base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最小边不小于15像素，最长边不超过8192像素。支持JPG/PNG/BMP/TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	与image二选一。 url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的URL路径目前仅支持华为云上OBS提供的匿名公开授权访问的URL以及公网URL。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。

响应参数

状态码： 200

表 4-264 响应 Body 参数

参数	参数类型	描述
result	**AcceptanceBillResult** object	调用成功时表示调用结果。 调用失败时无此字段。

版权所有 © 华为云计算技术有限公司

表 4-265 AcceptanceBillResult

参数	参数类型	描述
issue_date	String	出票日期。
due_date	String	汇票到期日。
bill_status	String	票据状态。
bill_number	String	票据号码。
issuer_full_name	String	出票人全称。
issuer_account	String	出票人账号。
issuer_bank_name	String	出票人开户银行。
issuer_bank_number	String	出票人开户行号。
payee_full_name	String	收款人全称。
payee_account	String	收款人账号。
payee_bank_name	String	收款人开户银行。
payee_bank_number	String	收款人开户行号。
issuance_guarantor_name	String	出票保证人名称。
issuance_guarantor_address	String	出票保证人地址。
issuance_guarantor_account	String	出票保证人账号。
issuance_guarantee_date	String	出票保证日期。
issuance_guarantor_bank_number	String	出票保证人开户行行号。
issuance_guarantor_bank_name	String	出票保证人开户行名称。
amount_in_words	String	大写票据金额。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
amount_in_figures	String	小写票据金额。
acceptor_full_name	String	承兑人全称。
acceptor_account	String	承兑人账号。
acceptor_bank_number	String	承兑人开户行行号。
acceptor_bank_name	String	承兑人开户行名称。
contract_number	String	交易合同号。
assignability	String	能否转让。
issuer_commitment	String	出票人承诺。
acceptor_commitment	String	承兑人承诺。
acceptance_date	String	承兑日期。
acceptance_guarantor_name	String	承兑保证人名称。
acceptance_guarantor_address	String	承兑保证人地址。
acceptance_guarantor_account	String	承兑保证人账号。
acceptance_guarantee_date	String	承兑保证日期。
acceptance_guarantor_bank_number	String	承兑保证人开户行行号。
acceptance_guarantor_bank_name	String	承兑保证人开户行名称。
issuer_rating_entity	String	出票人评级主体。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
issuer_credit_rating	String	出票人信用等级。
issuer_rating_due_date	String	出票人评级到期日。
acceptor_rating_entity	String	承兑人评级主体。
acceptor_credit_rating	String	承兑人信用等级。
acceptor_rating_due_date	String	承兑人评级到期日。
bill_package_number	String	票据包号。
remarks	String	备注。
confidence	Object	各个字段的置信度。

状态码： 400

表 4-266 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

请求示例

说明

- Endpoint即调用API的请求地址，不同服务不同区域的Endpoint不同，具体请参见**终端节点**。 例如，承兑汇票识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/acceptance-bill”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入承兑汇票图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/acceptance-bill
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
```


版权所有 © 华为云计算技术有限公司

```
{
  "image" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入承兑汇票图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/acceptance-bill
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
  "url" : "https://BucketName.obs.myhuaweicloud.com/ObjectName"
}
```

响应示例

状态码： 200

成功响应示例

```
{
  "result" : {
    "issue_date" : "20xx-07-10",
    "bill_status" : "背书待签收",
    "due_date" : "20xx-01-10",
    "bill_number" : "130212XXX",
    "issuer_account" : "811180XXX",
    "payee_account" : "793201XXX",
    "issuer_full_name" : "XX集团有限公司",
    "issuer_bank_name" : "XX银行股份有限公司XX支行",
    "issuer_bank_number" : "302121XXX",
    "payee_full_name" : "XX有限公司",
    "payee_bank_name" : "XX银行股份有限公司XX分行",
    "payee_bank_number" : "303124XXX",
    "amount_in_figures" : "￥XXX",
    "amount_in_words" : "XXX元整",
    "acceptor_bank_name" : "XX银行XX分行",
    "acceptor_full_name" : "XX银行XX分行",
    "acceptor_bank_number" : "302121XXX",
    "issuer_commitment" : "本汇票请予以承兑，到期无条件付款",
    "acceptor_commitment" : "本汇票已经承兑，到期无条件付款",
    "assignability" : "可再转让",
    "acceptance_date" : "20xx-07-10",
    "issuance_guarantor_name" : "xx",
    "issuance_guarantor_address" : "xx",
    "issuance_guarantor_account" : "xx",
    "issuance_guarantee_date" : "xx",
    "issuance_guarantor_bank_number" : "xx",
    "issuance_guarantor_bank_name" : "xx",
    "acceptor_account" : "xx",
    "contract_number" : "xx",
    "acceptance_guarantor_name" : "xx",
    "acceptance_guarantor_address" : "xx",
    "acceptance_guarantor_account" : "xx",
    "acceptance_guarantee_date" : "xx",
    "acceptance_guarantor_bank_number" : "xx",
    "acceptance_guarantor_bank_name" : "xx",
    "issuer_rating_entity" : "xx",
    "issuer_credit_rating" : "xx",
    "issuer_rating_due_date" : "xx",
    "acceptor_rating_entity" : "xx",
    "acceptor_credit_rating" : "xx",
    "acceptor_rating_due_date" : "xx",
    "remarks" : "xxx",
    "bill_package_number" : "xx",
    "confidence" : {
      "issue_date" : 0.9998,
      "bill_status" : 0.9997,
```


版权所有 © 华为云计算技术有限公司

```
    "due_date" : 0.9999,
    "bill_number" : 1.0,
    "issuer_account" : 1.0,
    "payee_account" : 1.0,
    "issuer_full_name" : 1.0,
    "issuer_bank_name" : 0.9999,
    "issuer_bank_number" : 0.999,
    "payee_full_name" : 1.0,
    "payee_bank_name" : 0.9999,
    "payee_bank_number" : 0.9999,
    "amount_in_figures" : 0.9999,
    "amount_in_words" : 0.9998,
    "acceptor_account" : 0.0,
    "acceptor_bank_name" : 0.9997,
    "acceptor_full_name" : 0.9999,
    "acceptor_bank_number" : 0.9998,
    "issuer_commitment" : 0.9998,
    "acceptor_commitment" : 1.0,
    "assignability" : 1.0,
    "acceptance_date" : 1.0,
    "issuance_guarantor_name" : 0.0,
    "issuance_guarantor_address" : 0.0,
    "issuance_guarantor_account" : 0.0,
    "issuance_guarantee_date" : 0.0,
    "issuance_guarantor_bank_number" : 0.0,
    "issuance_guarantor_bank_name" : 0.0,
    "contract_number" : 0.0,
    "acceptance_guarantor_name" : 0.0,
    "acceptance_guarantor_address" : 0.0,
    "acceptance_guarantor_account" : 0.0,
    "acceptance_guarantee_date" : 0.0,
    "acceptance_guarantor_bank_number" : 0.0,
    "acceptance_guarantor_bank_name" : 0.0,
    "issuer_rating_entity" : 0.0,
    "issuer_credit_rating" : 0.0,
    "issuer_rating_due_date" : 0.0,
    "acceptor_rating_entity" : 0.0,
    "acceptor_credit_rating" : 0.0,
    "acceptor_rating_due_date" : 0.0,
    "remarks" : 0.0,
    "bill_package_number" : 0.0
   }
  }
}
```


状态码： 400

失败响应示例


```
{
  "error_code" : "AIS.0103",
  "error_msg" : "The image size does not meet the requirements."
}
```


SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入承兑汇票图片的base64编码进行文字识别


```
package com.huaweicloud.sdk.test;
```


版权所有 © 华为云计算技术有限公司

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeAcceptanceBillSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeAcceptanceBillRequest request = new RecognizeAcceptanceBillRequest();
        AcceptanceBillRequestBody body = new AcceptanceBillRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeAcceptanceBillResponse response = client.recognizeAcceptanceBill(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入承兑汇票图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeAcceptanceBillSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
```

```
        // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeAcceptanceBillRequest request = new RecognizeAcceptanceBillRequest();
        AcceptanceBillRequestBody body = new AcceptanceBillRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeAcceptanceBillResponse response = client.recognizeAcceptanceBill(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入承兑汇票图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeAcceptanceBillRequest()
        request.body = AcceptanceBillRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_acceptance_bill(request)
        print(response)
```

 版权所有 © 华为云计算技术有限公司

```
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入承兑汇票图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeAcceptanceBillRequest()
        request.body = AcceptanceBillRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_acceptance_bill(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入承兑汇票图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
```

 版权所有 © 华为云计算技术有限公司

```
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeAcceptanceBillRequest{}
    imageAcceptanceBillRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/
4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.AcceptanceBillRequestBody{
        Image: &imageAcceptanceBillRequestBody,
    }
    response, err := client.RecognizeAcceptanceBill(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入承兑汇票图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeAcceptanceBillRequest{}
    urlAcceptanceBillRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.AcceptanceBillRequestBody{
        Url: &urlAcceptanceBillRequestBody,
    }
    response, err := client.RecognizeAcceptanceBill(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.34 银行回单识别

功能介绍

支持对银行回单版式进行文字识别及键值对提取，实现高效的自动化结构化返回。

该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

约束与限制

- 支持JPEG、JPG、PNG、BMP、TIFF、PDF格式，多页PDF默认识别第一页。
- 图像各边的像素大小在15px到8192px之间。单个图片、PDF文件其对应的Base64编码不超过10MB。
- 支持同时返回单张图像中存在的多张回单识别结果。
- 图像中key值对应的value值为空时，不会返回对应的键值对。
- 目前不支持复杂背景（如户外自然场景、防伪水印等）和文字扭曲图像的文字识别。
- 文字识别服务属于公有云服务，线上用户资源共享，如果需要多并发请求，请提前**联系我们**。

调用方法

请参见**如何调用API**。

前提条件

在使用财务报表识别之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

版权所有 © 华为云计算技术有限公司

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/bank-receipt

表 4-267 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-268 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。

版权所有 © 华为云计算技术有限公司

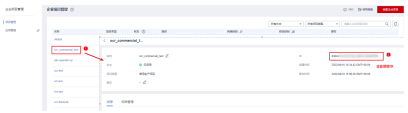
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-269 请求 Body 参数

参数	是否必选	参数类型	描述
data	否	String	该参数与url二选一。 base64编码，要求单个图片、PDF文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。 图像尺寸不小于15×15像素，最长边不超过8192像素，支持JPG/PNG/BMP/TIFF格式。 PDF以200dpi的分辨率转为图像进行识别，需符合上述图像尺寸规定。若PDF有多页，当前仅对第1页进行识别。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
url	否	String	与data二选一 单个图片、PDF文件其对应的Base64编码不超过10MB。文件在Base64编码后会大于文件原本大小，请注意做好边界判断，建议文件大小不超过7MB。图片的URL路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。
page_num	否	Integer	指定PDF页码识别。传入该参数时，则识别指定页码的内容。如果不传该参数，则默认识别第1页，该参数仅在文件为PDF格式时有效。
single_orientation_mode	否	Boolean	单朝向模式开关。可选值包括： • true：打开单朝向模式。 • false：关闭单朝向模式。 图片文字方向一致时，打开该开关可提升识别精度；图片文字方向不一致时，关闭该开关可支持多朝向文字识别。未传入该参数时默认为"true"，既默认图片中的文字方向为单朝向。
erase_seal	否	Boolean	是否打开印章擦除功能。可选值包括： • true：打开印章擦除功能。 • false：关闭印章擦除功能。 开启后，可提升印章遮挡区域的文字识别精度。

版权所有 © 华为云计算技术有限公司

响应参数

说明

根据识别的结果，可能有不同的HTTP响应状态码（status code）。例如，200表示API调用成功，400表示调用失败，详细的状态码和响应参数说明如下。

状态码： 200

表 4-270 响应 Body 参数

参数	参数类型	描述
result	**BankReceiptResult** object	调用成功时表示调用结果。 调用失败时无此字段。

表 4-271 BankReceiptResult

参数	参数类型	描述
bank_receipt_count	Integer	银行回单数量
bank_receipt_list	Array of **BankReceiptDict** objects	银行回单键值对提取结果。

表 4-272 BankReceiptDict

参数	参数类型	描述
kv_pair_count	Integer	键值对数量
bank_receipt_location	Array<Array<Integer>>	银行回单的区域位置信息，列表形式，包含文字区域四个顶点的二维坐标（x,y）;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。
kv_pair_list	Array of **BankReceiptKvPair** objects	键值对识别结果列表。

表 4-273 BankReceiptKvPair

参数	参数类型	描述
key	String	key-value对（键值对）中的key，例如"币别：人民币"中的"币别"。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
value	String	key-value对（键值对）中的value，例如"币别：人民币"中的“人民币”

状态码： 400

表 4-274 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见错误码。 调用成功时无此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。

  例如，银行回单识别服务部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/bank-receipt”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**认证鉴权**。

- 传入银行回单图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/bank-receipt

{
  "data" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入银行回单图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/bank-receipt

{
  "url" : "https://BucketName.obs.myhuaweicloud.com/ObjectName"
}
```

响应示例

状态码： 200

成功响应示例

```
{
  "result" : {
    "bank_receipt_count" : 2,
    "bank_receipt_list" : [ {
      "kv_pair_count" : 2,
      "bank_receipt_location" : [ [ 44, 34 ], [ 1063, 34 ], [ 1063, 393 ], [ 44, 393 ] ],
      "kv_pair_list" : [ {
        "key" : "币别",
        "value" : "人民币元"
```


版权所有 © 华为云计算技术有限公司

```
    }, {
      "key" : "日期",
      "value" : 20230511
    } ]
   }, {
    "kv_pair_count" : 5,
    "bank_receipt_location" : [ [ 40, 456 ], [ 1063, 456 ], [ 1062, 821 ], [ 40, 821 ] ],
    "kv_pair_list" : [ {
      "key" : "币别",
      "value" : "人民币元"
    }, {
      "key" : "日期",
      "value" : 20230512
    }, {
      "key" : "凭证号",
      "value" : 604914889863
    }, {
      "key" : "用途",
      "value" : "货款"
    }, {
      "key" : "钞汇标志",
      "value" : "钞"
    } ]
   } ]
  }
}
```

状态码： 400

失败响应示例

```
{
  "result" : {
    "error_code" : "AIS.0103",
    "error_msg" : "The image size does not meet the requirements."
  }
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入银行回单图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeBankReceiptSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
```


版权所有 © 华为云计算技术有限公司

```
        // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeBankReceiptRequest request = new RecognizeBankReceiptRequest();
        BankReceiptRequestBody body = new BankReceiptRequestBody();
        body.withData("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeBankReceiptResponse response = client.recognizeBankReceipt(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入银行回单图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeBankReceiptSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeBankReceiptRequest request = new RecognizeBankReceiptRequest();
        BankReceiptRequestBody body = new BankReceiptRequestBody();
```

```
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeBankReceiptResponse response = client.recognizeBankReceipt(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入银行回单图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeBankReceiptRequest()
        request.body = BankReceiptRequestBody(
            data="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_bank_receipt(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入银行回单图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

 版权所有 © 华为云计算技术有限公司

```
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeBankReceiptRequest()
        request.body = BankReceiptRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_bank_receipt(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入银行回单图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeBankReceiptRequest{}
    dataBankReceiptRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.BankReceiptRequestBody{
        Data: &dataBankReceiptRequestBody,
    }
    response, err := client.RecognizeBankReceipt(request)
    if err == nil {
```

 版权所有 © 华为云计算技术有限公司

```
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入银行回单图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeBankReceiptRequest{}
    urlBankReceiptRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.BankReceiptRequestBody{
        Url: &urlBankReceiptRequestBody,
    }
    response, err := client.RecognizeBankReceipt(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

版权所有 © 华为云计算技术有限公司

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.35 泰文身份证识别

功能介绍

识别泰国身份证中的文字信息，并以JSON格式返回识别的结构化结果。该接口的使用限制请参见**约束与限制**。

图 4-35 泰文身份证示例图



说明

使用中国站账号开通泰文身份证识别时，需要在**OCR控制台**的左上角，将区域切换至“亚太-曼谷”。

图 4-36 切换区域至亚太曼谷



版权所有 © 华为云计算技术有限公司

约束与限制

- 只支持泰国身份证的识别。
- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 泰国身份证识别图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中身份证区域有效占比超过25%，保证整张身份证内容及其边缘包含在图像内。
- 支持图像中身份证任意角度的水平旋转。
- 支持少量扭曲，扭曲后图像中的身份证长宽比与实际身份证相差不超过10%。
- 能处理反光、暗光等干扰的图片但影响识别精度。
- 目前只支持识别单张身份证的正面或者反面。

调用方法

请参见**如何调用API**。

前提条件

在使用泰文身份证识别之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/thailand-id-card

表 4-275 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

版权所有 © 华为云计算技术有限公司

请求参数

表 4-276 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-277 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。 图片最短边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	与image二选一。 url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。
side	否	String	标示身份证正面或背面。可选值如下： • front：身份证正面 • back：身份证背面 如果参数值为空或无该参数，系统自动识别，建议填写，准确率更高。
return_portrait_image	否	Boolean	是否返回身份证头像照片的 Base64 编码，可选值如下： • true：返回身份证头像照片的 Base64 编码 • false：不返回身份证头像照片的 Base64 编码 未传入该参数时默认为false，即不返回身份证头像照片的 Base64 编码。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	说明
return_portrait_location	否	Boolean	是否返回头像位置坐标，可选值如下： • true ： 返回身份证头像的位置坐标 • false：不返回身份证头像的位置坐标
return_idcard_type	否	Boolean	是否返回身份证类型，可选值如下： • true: 返回身份证类型，类型包括身份证原件和身份证复印件 • false：不返回身份证类型
return_text_location	否	Boolean	识别到的文字块的区域位置信息。可选值包括： true：返回各个文字块区域 false：不返回各个文字块区域 如果无该参数，系统默认不返回文字块区域。如果输入参数不是Boolean类型，则会报非法参数错误。

响应参数

状态码： 200

表 4-278 响应 Body 参数

参数	参数类型	描述
result	**ThailandIdcardResult** object	识别结果。 调用失败时不返回此字段。

表 4-279 ThailandIdcardResult

参数	参数类型	描述
type	String	返回证件类型，"normal"表示普通泰文证件，"pink"表示外国人身份证件
name_en	String	英文名。
ref_number	String	参考编码。
side	String	标示正面还是反面，取值为front或back。
id_number	String	身份证号。
name_th	String	泰文名字。

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
first_name_en	String	英文名字。
last_name_en	String	英文姓氏。
date_of_birth_th	String	泰文出生日期。
date_of_birth_en	String	英文出生日期。
religion_th	String	宗教。
address_th	String	地址。
date_of_issue_th	String	泰文签发日期。
date_of_issue_en	String	英文签发日期。
date_of_expiry_th	String	泰文有效期。
date_of_expiry_en	String	英文有效期。
serial_number	String	序列号。
card_number	String	身份证反面卡号。
laser_number	String	激光码。
confidence	ThailandIdcardConfidence object	字段的置信度，取值范围0~1。 置信度越大，本次识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。 置信度由算法给出，不直接等价于字段的准确率。
portrait_image	String	头像的Base64编码。 当输入参数“return_portrait_image”为“true”时，才返回该参数。
portrait_location	Array<Array<Integer>>	头像在原图上的位置。 当输入参数“return_portrait_location”为“true”时，才返回该参数。以列表形式显示，包含头像区域四个顶点的二维坐标（x,y），坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向
idcard_type	String	身份证的类型。当输入参数“return_idcard_type”为“true”时，才返回该参数。取值如下所示： • normal：身份证原件 • copy：复印的身份证

版权所有 © 华为云计算技术有限公司

参数	参数类型	描述
text_location	Object	对应所有在原图上识别到的字段位置信息，包含所有文字区域四个顶点的二维坐标（x,y）。采用图像坐标系，坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。

表 4-280 ThailandIdcardConfidence

参数	参数类型	描述
id_number	Float	身份证号置信度。
name_th	Float	泰文名字置信度。
name_en	Float	英文名置信度。
ref_number	Float	参考编码置信度。
first_name_en	Float	英文名字置信度。
last_name_en	Float	英文姓氏置信度。
date_of_birth_th	Float	泰文出生日期置信度。返回值例如，5 ก.พ. 2493。
date_of_birth_en	Float	英文出生日期置信度。返回值例如，5 Feb. 1950。
religion_th	Float	宗教置信度。
address_th	Float	地址置信度。
date_of_issue_th	Float	泰文签发日期置信度。
date_of_issue_en	Float	英文签发日期置信度。
date_of_expiry_th	Float	泰文有效期置信度。
date_of_expiry_en	Float	英文有效期置信度。
serial_number	Float	序列号置信度。
card_number	Float	身份证反面卡号置信度。
laser_number	Float	激光码置信度。

状态码： 200

版权所有 © 华为云计算技术有限公司

表 4-281 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。
  例如，泰文身份证识别服务部署在“亚太-曼谷”区域的“endpoint”为“ocr.ap-southeast-2.myhuaweicloud.com”或“ocr.ap-southeast-2.myhuaweicloud.cn”，请求URL为“https://ocr.ap-southeast-2.myhuaweicloud.com/v2/{project_id}/thailand-id-card”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**构造请求**。

- 请求示例（方式一：使用图片的Base64编码）

```
POST  https://{endpoint}/v2/{project_id}/ocr/thailand-id-card

Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:

{
  "image" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...",
  "side" : "front",
  "return_portrait_image" : true,
  "return_portrait_location" : true,
  "return_idcard_type" : true
}
```

- 请求示例（方式二：使用图片URL）

```
POST  https://{endpoint}/v2/{project_id}/ocr/thailand-id-card

Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "url":"https://BucketName.obs.xxxx.com/ObjectName"
}
```

响应示例

状态码：200

成功响应示例（正面）

```
{
  "result" : {
    "side" : "front",
    "id_number" : "X XXXX XXXXX XX X",
```


版权所有 © 华为云计算技术有限公司

```
    "name_th" : "XXX",
    "first_name_en" : "XX",
    "last_name_en" : "XX",
    "date_of_birth_th" : "5 ก.พ. 2493",
    "date_of_birth_en" : "5 Feb. 1950",
    "religion_th" : "XX",
    "address_th" : "XXXXX",
    "date_of_issue_th" : "XX",
    "date_of_issue_en" : "4 Mar. 2011",
    "date_of_expiry_th" : "22 ก.พ. 2561",
    "date_of_expiry_en" : "22 Feb. 2018",
    "serial_number" : "XXXX-XX-XXXXX",
    "confidence" : {
      "id_number" : 0.9999,
      "name_th" : 0.9994,
      "first_name_en" : 0.998,
      "last_name_en" : 0.9997,
      "date_of_birth_th" : 0.9996,
      "date_of_birth_en" : 0.9997,
      "religion_th" : 0.686,
      "address_th" : 0.624,
      "date_of_issue_th" : 0.9999,
      "date_of_issue_en" : 0.9999,
      "date_of_expiry_th" : 0.9969,
      "date_of_expiry_en" : 0.61,
      "serial_number" : 0.9887
    },
    "portrait_image" : "/9j/4AA... ",
    "portrait_location" : [ [ 576, 237 ], [ 741, 237 ], [ 739, 430 ], [ 574, 431 ] ],
    "idcard_type" : "normal"
  }
}
```


成功响应示例（背面）


```
{
    "result": {
        "side": "back",
        "card_number": "XXXX-XXX-XX",
        "laser_number": "XXXX-XXXXXXX-XX",
        "confidence": {
            "id_number": 0.9999,
            "laser_number": 0.9994
        }
    }
}
```


状态码：400

失败响应示例


```
{
    "error_code": "AIS.0103",
    "error_msg": "The image size does not meet the requirements."
}
```


SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

版权所有 © 华为云计算技术有限公司

Java

- 传入泰文身份证图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeThailandIdcardSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeThailandIdcardRequest request = new RecognizeThailandIdcardRequest();
        ThailandIdcardRequestBody body = new ThailandIdcardRequestBody();
        body.withReturnIdcardType(true);
        body.withReturnPortraitLocation(true);
        body.withReturnPortraitImage(true);
        body.withSide("front");
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeThailandIdcardResponse response = client.recognizeThailandIdcard(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入泰文身份证图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
```

 版权所有 © 华为云计算技术有限公司

```
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeThailandIdcardSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeThailandIdcardRequest request = new RecognizeThailandIdcardRequest();
        ThailandIdcardRequestBody body = new ThailandIdcardRequestBody();
        body.withReturnIdcardType(true);
        body.withReturnPortraitLocation(true);
        body.withReturnPortraitImage(true);
        body.withSide("front");
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeThailandIdcardResponse response = client.recognizeThailandIdcard(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入泰文身份证图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \
```

 版权所有 © 华为云计算技术有限公司

```
client = OcrClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(OcrRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = RecognizeThailandIdcardRequest()
    request.body = ThailandIdcardRequestBody(
        return_idcard_type=True,
        return_portrait_location=True,
        return_portrait_image=True,
        side="front",
        image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    )
    response = client.recognize_thailand_idcard(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- 传入泰文身份证图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeThailandIdcardRequest()
        request.body = ThailandIdcardRequestBody(
            return_idcard_type=True,
            return_portrait_location=True,
            return_portrait_image=True,
            side="front",
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_thailand_idcard(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入泰文身份证图片的base64编码进行文字识别

 版权所有 © 华为云计算技术有限公司

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeThailandIdcardRequest{}
    returnIdcardTypeThailandIdcardRequestBody:= true
    returnPortraitLocationThailandIdcardRequestBody:= true
    returnPortraitImageThailandIdcardRequestBody:= true
    sideThailandIdcardRequestBody:= "front"
    imageThailandIdcardRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.ThailandIdcardRequestBody{
        ReturnIdcardType: &returnIdcardTypeThailandIdcardRequestBody,
        ReturnPortraitLocation: &returnPortraitLocationThailandIdcardRequestBody,
        ReturnPortraitImage: &returnPortraitImageThailandIdcardRequestBody,
        Side: &sideThailandIdcardRequestBody,
        Image: &imageThailandIdcardRequestBody,
    }
    response, err := client.RecognizeThailandIdcard(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入泰文身份证图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
```

 版权所有 © 华为云计算技术有限公司

```
environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeThailandIdcardRequest{}
    returnIdcardTypeThailandIdcardRequestBody:= true
    returnPortraitLocationThailandIdcardRequestBody:= true
    returnPortraitImageThailandIdcardRequestBody:= true
    sideThailandIdcardRequestBody:= "front"
    urlThailandIdcardRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.ThailandIdcardRequestBody{
        ReturnIdcardType: &returnIdcardTypeThailandIdcardRequestBody,
        ReturnPortraitLocation: &returnPortraitLocationThailandIdcardRequestBody,
        ReturnPortraitImage: &returnPortraitImageThailandIdcardRequestBody,
        Side: &sideThailandIdcardRequestBody,
        Url: &urlThailandIdcardRequestBody,
    }
    response, err := client.RecognizeThailandIdcard(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

版权所有 © 华为云计算技术有限公司

4.36 哥伦比亚身份证识别

功能介绍

识别哥伦比亚身份证中的文字信息，并将识别的结构化结果返回给用户。

约束与限制

- 支持哥伦比亚身份证的识别。
- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式图片。
- 图像各边的像素大小在15px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中身份证区域有效占比超过80%，保证整张身份证内容及其边缘包含在图像内。
- 支持图像中身份证任意角度的水平旋转。
- 支持少量扭曲，扭曲后图像中的身份证长宽比与实际身份证相差不超过10%。
- 能处理反光、暗光等干扰的图片但影响识别精度。
- 目前只支持识别单张身份证的正面或者反面。

调用方法

请参见**如何调用API**。

前提条件

在使用之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/colombia-id-card

表 4-282 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。

版权所有 © 华为云计算技术有限公司

参数	是否必选	说明
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-283 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。 企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-284 请求 Body 参数

参数	是否必选	参数类型	描述
image	否	String	该参数与url二选一。 图像数据，base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片最短边不小于15像素，最长边不超过8192像素，支持JPG/PNG/BMP/TIFF格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	与image二选一。url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片的url路径目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**>。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。 • url中不能存在中文字符，若存在，中文需要进行utf8编码。

响应参数

状态码： 200

版权所有 © 华为云计算技术有限公司

表 4-285 响应 Body 参数

参数	参数类型	描述
result	表4-286 object	调用成功时表示调用结果。调用失败时无此字段。

表 4-286 ColombiaIdCardResult

参数	参数类型	描述
side	String	证件图片正反面信息。可选值包括： • front: 证件图片正面 • back: 证件图片反面
number	String	卡证编号。当响应字段"side"为front时，返回此字段。
name	String	名。当响应字段"side"为front时，返回此字段。
last_name	String	姓。当响应字段"side"为front时，返回此字段。
birth_date	String	出生日期。
birth_place	String	出生地。
gender	String	性别。
blood_type	String	血型。
issue_date	String	签发日期。
issue_authority	String	签发地区。
height	String	身高。
citizen_code1	String	公民编码一。
citizen_code2	String	公民编码二。
citizen_code3	String	公民编码三。
confidence	Map<String,Number>	相关字段的置信度信息，置信度越大，表示本次识别的对应字段的可靠性越高，在统计意义上，置信度越大，准确率越高。注：置信度由算法给出，不直接等价于对应字段的准确率。

状态码： 400

版权所有 © 华为云计算技术有限公司

表 4-287 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

请求示例

说明

- Endpoint即调用API的请求地址，不同服务不同区域的Endpoint不同，具体请参见**终端节点**。
  例如，哥伦比亚身份证识别服务部署在“拉美-墨西哥城二”区域的“endpoint”为“ocr.la-north-2.myhuaweicloud.com”，请求URL为“https://ocr.la-north-2.myhuaweicloud.com/v2/{project_id}/ocr/colombia-id-card”，“project_id”为项目ID，获取方法请参见**获取项目ID**。
- 如何获取Token请参见**构造请求**。

- 传入哥伦比亚身份证图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/colombia-id-card
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
  "image" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 传入哥伦比亚身份证图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/colombia-id-card
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
  "url" : "https://BucketName.obs.myhuaweicloud.com/ObjectName"
}
```

响应示例

状态码： 200

成功响应示例

```
{
  "result" : {
    "side" : "front",
    "number" : "1.039xxxx",
    "name" : "VALENxxx",
    "last_name" : "SANCHxxxx",
    "confidence" : {
      "number" : 0.9888,
      "name" : 0.9969,
      "last_name" : 0.9959
    }
```


版权所有 © 华为云计算技术有限公司

```
  }
}
```

状态码： 400

失败响应示例

```
{
  "error_code" : "AIS.0103",
  "error_msg" : "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入哥伦比亚身份证图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeColombiaIdCardSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeColombiaIdCardRequest request = new RecognizeColombiaIdCardRequest();
        ColombiaIdCardRequestBody body = new ColombiaIdCardRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeColombiaIdCardResponse response = client.recognizeColombiaIdCard(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入哥伦比亚身份证图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeColombiaIdCardSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeColombiaIdCardRequest request = new RecognizeColombiaIdCardRequest();
        ColombiaIdCardRequestBody body = new ColombiaIdCardRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeColombiaIdCardResponse response = client.recognizeColombiaIdCard(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入哥伦比亚身份证图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
```

```
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeColombiaIdCardRequest()
        request.body = ColombiaIdCardRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_colombia_id_card(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入哥伦比亚身份证图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeColombiaIdCardRequest()
        request.body = ColombiaIdCardRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_colombia_id_card(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

 版权所有 © 华为云计算技术有限公司

```
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入哥伦比亚身份证图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeColombiaIdCardRequest{}
    imageColombiaIdCardRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.ColombiaIdCardRequestBody{
        Image: &imageColombiaIdCardRequestBody,
    }
    response, err := client.RecognizeColombiaIdCard(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入哥伦比亚身份证图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```

 版权所有 © 华为云计算技术有限公司

```
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeColombiaIdCardRequest{}
    urlColombiaIdCardRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.ColombiaIdCardRequestBody{
        Url: &urlColombiaIdCardRequestBody,
    }
    response, err := client.RecognizeColombiaIdCard(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

请参见**错误码**。

4.37 泰国车牌识别

功能介绍

识别泰国车牌图片中的车牌信息，返回识别结果和车牌的区域位置信息。该接口的使用限制请参见**约束与限制**，详细使用指导请参见**OCR服务使用简介**章节。

版权所有 © 华为云计算技术有限公司

图 4-37 泰国车牌示例图



约束与限制

- 只支持识别PNG、JPG、JPEG、BMP、TIFF格式的图片。
- 图像各边的像素大小在15到8192px之间。单个图片对应的Base64编码不超过10MB。
- 支持图像中车牌任意角度的水平旋转。
- 能处理反光、暗光等干扰的图片但影响识别精度。

调用方法

请参见**如何调用API**。

前提条件

在使用泰国车牌识别之前，需要您完成服务申请和认证鉴权，具体操作流程请参见**开通服务**和**认证鉴权**章节。

说明

用户首次使用需要先**申请开通**。服务只需要开通一次即可，后面使用时无需再次申请。如未开通服务，调用服务时会提示ModelArts.4204报错，请在调用服务前先进入控制台开通服务，并注意开通服务区域与调用服务的区域保持一致。

URI

POST /v2/{project_id}/ocr/thailand-license-plate

表 4-288 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

版权所有 © 华为云计算技术有限公司

请求参数

表 4-289 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“**企业项目管理**”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

版权所有 © 华为云计算技术有限公司

表 4-290 请求 Body 参数

参数	是否必选	参数类型	说明
image	否	String	与url二选一。 图片的Base64编码，单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。图片尺寸不小于15×15像素，最长边不超过8192像素，支持JPG/PNG/BMP/TIFF/JPEG格式。 图片Base64编码示例如/9j/4AAQSkZJRgABAg...，带有多余前缀会产生The image format is not supported报错。
url	否	String	与image二选一。 图片的url路径，url中单个图片其对应的Base64编码不超过10MB。图片在Base64编码后会大于图片原本大小，请注意做好边界判断，建议图片大小不超过7MB。目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。

响应参数

状态码： 200

表 4-291 响应 Body 参数

参数	参数类型	描述
result	Array of **ThailandLicensePlateItem** objects	调用结果。 调用失败时不返回此字段。

版权所有 © 华为云计算技术有限公司

表 4-292 ThailandLicensePlateItem

参数	参数类型	描述
plate_number	String	车牌内容。
plate_location	Array<Array<Integer>>	车牌的区域位置信息，列表形式，包含文字区域四个顶点的二维坐标（x,y）;坐标原点为图片左上角，x轴沿水平方向，y轴沿竖直方向。
confidence	Float	字段的置信度，取值范围0~1。 置信度越大，本次识别的字段的可靠性越高，在统计意义上，置信度越大，准确率越高。注：置信度由算法给出，不直接等价于字段的准确率。
province	String	车牌所属府。

状态码： 400

表 4-293 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时不返回此字段。
error_msg	String	调用失败时返回的错误信息。 调用成功时不返回此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。
  例如，泰国车牌识别服务部署在“亚太-曼谷”区域的“endpoint”为“ocr.ap-southeast-2.myhuaweicloud.com”或“ocr.ap-southeast-2.myhuaweicloud.cn”，请求URL为“https://ocr.ap-southeast-2.myhuaweicloud.com/v2/{project_id}/ocr/thailand-license-plate”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**构造请求**。

- 请求样例（方式一：使用图片的Base64编码）

```
POST https://{endpoint}/v2/{project_id}/ocr/thailand-license-plate

Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
   "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
}
```

- 请求样例（方式二：使用图片URL）

```
POST https://{endpoint}/v2/{project_id}/ocr/thailand-license-plate
```


版权所有 © 华为云计算技术有限公司

```
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
   "url":"https://BucketName.obs.xxxx.com/ObjectName"
}
```

- Python3语言请求代码示例（其他语言参照下列示例编写或使用OCR SDK）

```
# encoding:utf-8

import requests
import base64

url = "https://{endpoint}/v2/{project_id}/ocr/thailand-license-plate"
token = "用户获取得到的实际token值"
headers = {'Content-Type': 'application/json', 'X-Auth-Token': token}

imagepath = r'./data/thailand-license-plate-demo.png' # 读取本地图片
with open(imagepath, "rb") as bin_data:
    image_data = bin_data.read()
image_base64 = base64.b64encode(image_data).decode("utf-8")  # 使用图片的Base64编码
payload = {"image": image_base64}
response = requests.post(url, headers=headers, json=payload)
print(response.text)
```

响应示例

状态码：200

成功响应示例

```
{
  "result" : [ {
    "plate_number" : "กง XXX4",
    "province" : "มหาสารคาม",
    "confidence" : 0.9225,
    "plate_location" : [ [ 370, 881 ], [ 2591, 881 ], [ 2591, 2281 ], [ 370, 2281 ] ]
  } ]
}
```

状态码：400

失败响应示例

```
{
   "error_code": "AIS.0103",
   "error_msg": "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入泰国车牌图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
```

 版权所有 © 华为云计算技术有限公司

```
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeThailandLicensePlateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeThailandLicensePlateRequest request = new RecognizeThailandLicensePlateRequest();
        ThailandLicensePlateRequestBody body = new ThailandLicensePlateRequestBody();
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeThailandLicensePlateResponse response = client.recognizeThailandLicensePlate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入泰国车牌图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeThailandLicensePlateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
```

```
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeThailandLicensePlateRequest request = new RecognizeThailandLicensePlateRequest();
        ThailandLicensePlateRequestBody body = new ThailandLicensePlateRequestBody();
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeThailandLicensePlateResponse response =
client.recognizeThailandLicensePlate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- 传入泰国车牌图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeThailandLicensePlateRequest()
        request.body = ThailandLicensePlateRequestBody(
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_thailand_license_plate(request)
        print(response)
```

 版权所有 © 华为云计算技术有限公司

```
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入泰国车牌图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeThailandLicensePlateRequest()
        request.body = ThailandLicensePlateRequestBody(
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_thailand_license_plate(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入泰国车牌图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
```

 版权所有 © 华为云计算技术有限公司

```
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeThailandLicensePlateRequest{}
    imageThailandLicensePlateRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/
4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.ThailandLicensePlateRequestBody{
        Image: &imageThailandLicensePlateRequestBody,
    }
    response, err := client.RecognizeThailandLicensePlate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入泰国车牌图片的url进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeThailandLicensePlateRequest{}
    urlThailandLicensePlateRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.ThailandLicensePlateRequestBody{
        Url: &urlThailandLicensePlateRequestBody,
    }
    response, err := client.RecognizeThailandLicensePlate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

 版权所有 © 华为云计算技术有限公司

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

错误码

错误码请参见**错误码**。

4.38 自定义模板 OCR

说明

OCR自定义模板将于2025.12.31正式下线，为避免影响您目前的业务，请您及时进行相关迁移工作，迁移指导详见**OCR自定义模板下线公告**。

功能介绍

自定义模板OCR，支持用户自定义模板，对于版式固定的各种票据和卡证，通过可视化界面操作，指定需要识别的关键字段，实现用户特定格式图片的自动识别和结构化提取。

约束与限制

- 只支持识别PNG、JPG、PNG、BMP、TIFF、GIF、WEBP格式图片。
- 图像各边的像素大小在100px到8192px之间。单个图片对应的Base64编码不超过10MB。
- 图像中识别区域有效占比超过80%，保证整张表格及其边缘包含在图像内。
- 支持图像任意角度的水平旋转。
- 目前不支持复杂背景（如户外自然场景、防伪水印等）和表格线扭曲图像的文字识别。
- 支持中英文以及部分繁体字。
- API支持识别的图片版式依赖于创建自定义OCR工作流时的类型。

调用方法

请参见**如何调用API**。

版权所有 © 华为云计算技术有限公司

前提条件

在使用之前，你需要通过OCR**控制台**，完成自定义OCR工作流的开发。开发、部署完成后，将在工作流的部署页面生成API的URI。

自定义OCR工作流开发过程，可参考**自定义OCR介绍**。

URI

POST /v2/{project_id}/ocr/custom-template

表 4-294 路径参数

参数	是否必选	说明
endpoint	是	终端节点，即调用API的请求地址。 不同服务不同区域的endpoint不同，您可以从**终端节点**中获取。 例如，OCR服务在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”。
project_id	是	项目ID，您可以从**获取项目ID**中获取。

请求参数

表 4-295 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 用于获取操作API的权限。**获取Token接口**响应消息头中X-Subject-Token的值即为Token。
Content-Type	是	String	发送的实体的MIME类型，参数值为“application/json”。

版权所有 © 华为云计算技术有限公司

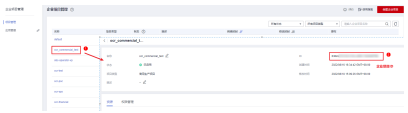
参数	是否必选	参数类型	描述
Enterprise-Project-Id	否	String	企业项目ID。OCR支持通过企业项目管理（EPS）对不同用户组和用户的资源使用，进行分账。 获取方法：进入“企业项目管理”页面，单击企业项目名称，在企业项目详情页获取Enterprise-Project-Id（企业项目ID）。  企业项目创建步骤请参见用户指南。 **说明** 创建企业项目后，在传参时，有以下三类场景。 • 携带正确的ID，正常使用OCR服务，账单的企业项目会被分类到企业ID对应的企业项目中。 • 携带格式正确但不存在的ID，正常使用OCR服务，账单的企业项目会显示对应不存在的企业项目ID。 • 不携带ID或格式错误ID（包含特殊字符等），正常使用OCR服务，账单的企业项目会被分类到"default"中。

表 4-296 请求 Body 参数

参数	是否必选	参数类型	描述
image	否	String	与url二选一 图像数据，base64编码，要求base64编码后大小不超过10MB。图片最小边不小于15px，最长边不超过8192px，支持JPEG、JPG、PNG、BMP、TIFF、GIF、WEBP格式。 图片文件Base64编码字符串，单击这里查看详细获取方式。

版权所有 © 华为云计算技术有限公司

参数	是否必选	参数类型	描述
url	否	String	与image二选一 图片的URL路径，目前支持： • 公网http/https url • OBS提供的url，使用OBS数据需要进行授权。包括对服务授权、临时授权、匿名公开授权，详情参见**配置OBS访问权限**。 **说明** • 接口响应时间依赖于图片的下载时间，如果图片下载时间过长，会返回接口调用失败。 • 请保证被检测图片所在的存储服务稳定可靠，推荐使用OBS服务存储图片数据。
template_id	否	String	该参数与classifier_id二选一。模型id，如果传入，启用单模板识别模式。
classifier_id	否	String	该参数与template_id二选一。分类器id，如果传入，启用多模板识别模式。
classifier_mode	否	Boolean	该参数与classifier_id参数配合使用，可选值如下所示： • true：仅返回模板分类结果 • false：正常返回多模板识别结果 **说明** 如果未传入该参数时默认为false，即正常返回多模板识别结果。

响应参数

状态码： 200

表 4-297 响应 Body 参数

参数	参数类型	描述
result	Object	调用成功时表示调用结果。 调用失败时无此字段。
template_id	String	调用成功时返回调用模板id。 调用失败时无此字段。

版权所有 © 华为云计算技术有限公司

状态码： 400

表 4-298 响应 Body 参数

参数	参数类型	描述
error_code	String	调用失败时的错误码，具体请参见**错误码**。 调用成功时无此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

请求示例

说明

- “endpoint”即调用API的请求地址，不同服务不同区域的“endpoint”不同，具体请参见**终端节点**。
  例如，自定义模板OCR部署在“华北-北京四”区域的“endpoint”为“ocr.cn-north-4.myhuaweicloud.com”或“ocr.cn-north-4.myhuaweicloud.cn”，请求URL为“https://ocr.cn-north-4.myhuaweicloud.com/v2/{project_id}/ocr/custom-template”，“project_id”为项目ID，获取方法请参见**获取项目ID**
- 如何获取Token请参见**认证鉴权**。

- 传入自定义模板OCR的图片的base64编码进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/custom-template
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
  "image" : "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...",
  "classifier_id" : "3bed12e0-7c1e-11ed-9bdb-0255ac10009d",
  "classifier_mode" : true
}
```

- 传入自定义模板OCR的图片的url进行文字识别

```
POST https://{endpoint}/v2/{project_id}/ocr/custom-template
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
  "url" : "https://BucketName.obs.myhuaweicloud.com/ObjectName",
  "classifier_id" : "3bed12e0-7c1e-11ed-9bdb-0255ac10009d",
  "classifier_mode" : true
}
```

响应示例

状态码： 200

成功响应示例

```
{
  "result" : {
    "template_id" : "43580526-658f-11ed-863a-0242ac11000b",
    "template_name" : "3",
```


版权所有 © 华为云计算技术有限公司

```
    "confidence" : 0.4057
  },
  "template_id" : "43580526-658f-11ed-863a-0242ac11000b"
}
```

状态码： 400

失败响应示例

```
{
  "error_code" : "AIS.0103",
  "error_msg" : "The image size does not meet the requirements."
}
```

SDK 代码示例

SDK代码示例如下。

说明

使用SDK前建议将SDK更新至最新版，防止本地旧版SDK无法使用最新的OCR功能。

Java

- 传入自定义模板OCR的图片的base64编码进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeCustomTemplateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeCustomTemplateRequest request = new RecognizeCustomTemplateRequest();
        CustomTemplateRequestBody body = new CustomTemplateRequestBody();
        body.withClassifierMode(true);
        body.withClassifierId("3bed12e0-7c1e-11ed-9bdb-0255ac10009d");
        body.withImage("/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...");
        request.withBody(body);
        try {
            RecognizeCustomTemplateResponse response = client.recognizeCustomTemplate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

 版权所有 © 华为云计算技术有限公司

```
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- 传入自定义模板OCR的图片的url进行文字识别

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;


public class RecognizeCustomTemplateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        OcrClient client = OcrClient.newBuilder()
                .withCredential(auth)
                .withRegion(OcrRegion.valueOf("<YOUR REGION>"))
                .build();
        RecognizeCustomTemplateRequest request = new RecognizeCustomTemplateRequest();
        CustomTemplateRequestBody body = new CustomTemplateRequestBody();
        body.withClassifierMode(true);
        body.withClassifierId("3bed12e0-7c1e-11ed-9bdb-0255ac10009d");
        body.withUrl("https://BucketName.obs.myhuaweicloud.com/ObjectName");
        request.withBody(body);
        try {
            RecognizeCustomTemplateResponse response = client.recognizeCustomTemplate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

 版权所有 © 华为云计算技术有限公司

Python

- 传入自定义模板OCR的图片的base64编码进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeCustomTemplateRequest()
        request.body = CustomTemplateRequestBody(
            classifier_mode=True,
            classifier_id="3bed12e0-7c1e-11ed-9bdb-0255ac10009d",
            image="/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
        )
        response = client.recognize_custom_template(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 传入自定义模板OCR的图片的url进行文字识别

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkocr.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = OcrClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(OcrRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RecognizeCustomTemplateRequest()
        request.body = CustomTemplateRequestBody(
```


版权所有 © 华为云计算技术有限公司

```
            classifier_mode=True,
            classifier_id="3bed12e0-7c1e-11ed-9bdb-0255ac10009d",
            url="https://BucketName.obs.myhuaweicloud.com/ObjectName"
        )
        response = client.recognize_custom_template(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 传入自定义模板OCR的图片的base64编码进行文字识别

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeCustomTemplateRequest{}
    classifierModeCustomTemplateRequestBody:= true
    classifierIdCustomTemplateRequestBody:= "3bed12e0-7c1e-11ed-9bdb-0255ac10009d"
    imageCustomTemplateRequestBody:= "/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA..."
    request.Body = &model.CustomTemplateRequestBody{
        ClassifierMode: &classifierModeCustomTemplateRequestBody,
        ClassifierId: &classifierIdCustomTemplateRequestBody,
        Image: &imageCustomTemplateRequestBody,
    }
    response, err := client.RecognizeCustomTemplate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 传入自定义模板OCR的图片的url进行文字识别

```
package main

import (
    "fmt"
```

 版权所有 © 华为云计算技术有限公司

```
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := ocr.NewOcrClient(
        ocr.OcrClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RecognizeCustomTemplateRequest{}
    classifierModeCustomTemplateRequestBody:= true
    classifierIdCustomTemplateRequestBody:= "3bed12e0-7c1e-11ed-9bdb-0255ac10009d"
    urlCustomTemplateRequestBody:= "https://BucketName.obs.myhuaweicloud.com/ObjectName"
    request.Body = &model.CustomTemplateRequestBody{
        ClassifierMode: &classifierModeCustomTemplateRequestBody,
        ClassifierId: &classifierIdCustomTemplateRequestBody,
        Url: &urlCustomTemplateRequestBody,
    }
    response, err := client.RecognizeCustomTemplate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见**API Explorer**的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	成功响应示例
400	失败响应示例

状态码请参见**状态码**。

版权所有 © 华为云计算技术有限公司

错误码

错误码请参见**错误码**。

版权所有 © 华为云计算技术有限公司

5 状态码

HTTP 状态码为三位数，分成五个类别：1xx：相关信息；2xx：操作成功；3xx：重定向；4xx：客户端错误；5xx：服务器错误。

状态码如下所示。

状态码	编码	状态说明
100	Continue	继续请求。 这个临时响应用来通知客户端，它的部分请求已经被服务器接收，且仍未被拒绝。
101	Switching Protocols	切换协议。只能切换到更高级的协议。 例如，切换到HTTPS的新版本协议。
200	OK	服务器已成功处理了请求。
201	Created	创建类的请求完全成功。
202	Accepted	已经接受请求，但未处理完成。
203	Non-Authoritative Information	非授权信息，请求成功。
204	No Content	请求完全成功，同时HTTP响应不包含响应体。 在响应OPTIONS方法的HTTP请求时返回此状态码。
205	Reset Content	重置内容，服务器处理成功。
206	Partial Content	服务器成功处理了部分GET请求。
300	Multiple Choices	多种选择。请求的资源可包括多个位置，相应可返回一个资源特征与地址的列表用于用户终端（例如：浏览器）选择。
301	Moved Permanently	永久移动，请求的资源已被永久的移动到新的URI，返回信息会包括新的URI。
302	Found	资源被临时移动。

版权所有 © 华为云计算技术有限公司

状态码	编码	状态说明
303	See Other	查看其它地址，使用GET和POST请求查看。
304	Not Modified	所请求的资源未修改，服务器返回此状态码时，不会返回任何资源。
305	Use Proxy	所请求的资源必须通过代理访问。
306	Unused	已经被废弃的HTTP状态码。
400	Bad Request	非法请求。 建议直接修改该请求，不要重试该请求。
401	Unauthorized	在客户端提供认证信息后，返回该状态码，表明服务端指出客户端所提供的认证信息不正确或非法。
402	Payment Required	保留请求。
403	Forbidden	请求被拒绝访问。 返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
404	Not Found	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。
405	Method Not Allowed	请求中带有该资源不支持的方法。 建议直接修改该请求，不要重试该请求。
406	Not Acceptable	服务器无法根据客户端请求的内容特性完成请求。
407	Proxy Authentication Required	请求要求代理的身份认证，与401类似，但请求者应当使用代理进行授权。
408	Request Timeout	服务器等候请求时发生超时。 客户端可以随时再次提交该请求而无需进行任何更改。
409	Conflict	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
410	Gone	客户端请求的资源已经不存在。 返回该状态码，表明请求的资源已被永久删除。
411	Length Required	服务器无法处理客户端发送的不带Content-Length的请求信息。

版权所有 © 华为云计算技术有限公司

状态码	编码	状态说明
412	Precondition Failed	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。
413	Request Entity Too Large	由于请求的实体过大，服务器无法处理，因此拒绝请求。为防止客户端的连续请求，服务器可能会关闭连接。如果只是服务器暂时无法处理，则会包含一个Retry-After的响应信息。
414	Request URI Too Long	请求的URI过长（URI通常为网址），服务器无法处理。
415	Unsupported Media Type	服务器无法处理请求附带的媒体格式。
416	Requested Range Not Satisfiable	客户端请求的范围无效。
417	Expectation Failed	服务器无法满足Expect的请求头信息。
422	Unprocessable Entity	请求格式正确，但是由于含有语义错误，无法响应。
429	Too Many Requests	表明请求超出了客户端访问频率的限制或者服务端接收到多于它能处理的请求。建议客户端读取相应的Retry-After首部，然后等待该首部指出的时间后再重试。
500	Internal Server Error	表明服务端能被请求访问到，但是不能理解用户的请求。
501	Not Implemented	服务器不支持请求的功能，无法完成请求。
502	Bad Gateway	充当网关或代理的服务器，从远端服务器接收到了一个无效的请求。
503	Service Unavailable	被请求的服务无效。 建议直接修改该请求，不要重试该请求。
504	Gateway Timeout	请求在给定的时间内无法完成。客户端仅在为请求指定超时（Timeout）参数时会得到该响应。
505	HTTP Version Not Supported	服务器不支持请求的HTTPS协议的版本，无法完成处理。

版权所有 © 华为云计算技术有限公司

6 错误码

调用接口出错后，将不会返回结果数据。调用方可根据每个接口对应的错误码来定位错误原因。当调用出错时，HTTPS请求返回一个 4xx 或 5xx 的 HTTPS状态码。返回的消息体中是具体的错误代码及错误信息。在调用方找不到错误原因时，可以联系华为云客服技术支持，并提供错误码，以便尽快帮您解决问题。

错误响应 Body 体格式说明

当接口调用出错时，会返回错误码及错误信息说明，错误响应的Body体格式如下所示。

```
{
    "error_msg": "The input parameter is invalid.",
    "error_code": "AIS.0101"
}
```

其中，error_code表示错误码，error_msg表示错误描述信息。

错误码说明

由于OCR服务部署在ModelArts平台，使用API网关（API Gateway）系统，因此OCR服务错误码会出现“ModelArts”、“APIGW”、“APIG”开头的错误码。

当您调用API时，如果遇到“APIGW”开头的错误码，请参见**API网关错误码**进行处理。遇到“APIG”开头的错误码，请参考本文档进行处理。

版权所有 © 华为云计算技术有限公司

模块	错误码	错误信息	说明	建议解决方法
文字识别	AIS.0101	The input parameter is invalid.	输入参数不符合规范。	请检查输入参数是否正确。出现此报错可能原因如下： • image、url参数为二选一，请勿同时填写。填写时注意格式规范。 • 功能开关true、false拼写错误。 • 传参时中英文字符错误。 具体参数格式参见各API的请求参数说明。
	AIS.0102	The image format is not supported.	图片格式不支持，或image参数值不是base64编码字符串。	请检查图片格式，各服务支持图片格式请参考**使用限制**。
	AIS.0103	The image size does not meet the requirements.	图片尺寸不满足要求。	请检查图片尺寸，各服务支持图片尺寸请参考**使用限制**。
	AIS.0104	The image is not supported or the image quality is poor.	非支持的图片类型或图片质量差。	请检查图片类型与图片质量。 请检查调用的接口是否正确。
	AIS.0105	Recognition failed.	算法计算失败。	请**联系华为云支持人员**。
	AIS.0117	Input ID does not exist.	输入ID不存在。	请检查入参中的classifier_id或template_id是否存在。
	AIS.0118	Service request error or the tax authority’s interface was malfunctioning.	税局机关接口超时未响应，导致服务请求错误。	请等待税局机关接口恢复(建议2小时后重试)。
	AIS.0119	Template match failed, template_id={xxx}.	输入图片与模板匹配失败。	请检查模板中的参照字段是否与输入图片匹配。
	AIS.0120	Classification failed.	输入图片分类失败。	请检查分类器中模板是否与输入图片匹配。

版权所有 © 华为云计算技术有限公司

模块	错误码	错误信息	说明	建议解决方法
	AIS.0121	The recognition area of template is not configured, template_id={xxx}	模板未配置识别区。	请检查模板是否配置了识别区。
	AIS.0122	The image contains two or more ID cards with the same side.	存在两张及以上同面身份证。	请检查输入图片是否存在两张及以上同面身份证。身份证服务参数side为double_side时，图片中不能存在两张及以上同面身份证。
平台	ModelArts.0203	Invalid token.	非法Token。	请检查Token是否正确。
	ModelArts.4101	Token header cannot be empty.	Token为空。	发送的HTTP请求header中没有带x-auth-token的token请求认证信息，请检查发送的请求。
	ModelArts.4102	Parse Token error.	解析Token失败。	发送的HTTP请求header中x-auth-token的token请求认证信息错误，请检查发送的请求和token。
	ModelArts.4103	Invalid Token header.	Token不合法。	发送的HTTP请求header中x-auth-token的token请求认证信息错误，请检查发送的请求和token。
	ModelArts.4104	Invalid Request Content Length.	请求body体长度不合法。	请检查请求body体长度。要求Base64编码后大小不超过10MB。
	ModelArts.4105	The JSON format of the input data is incorrect.	请求body体JSON格式不对。	请检查请求body体是否符合JSON格式。
		Failed to obtain the temporary AK/SK	获取用户临时AK，SK失败	请联系技术支持人员（如果有使用OBS，请先检查授权）。
	ModelArts.4106	Invalid authorization request.	用户账号受限。	请检查用户资源状态。账号受限原因请参考**帐号中心相关描述**。

版权所有 © 华为云计算技术有限公司

模块	错误码	错误信息	说明	建议解决方法
	ModelArts.4107	Get user temp ak sk error.	获取用户临时AK，SK异常。	请**联系华为云支持人员**。
	ModelArts.4201	Request url need service id.	请求url中需要包含服务ID。	请检查请求url中的服务ID。
	ModelArts.4202	Request url format invalid.	请求url格式不合法。	请检查请求url格式。
	ModelArts.4203	Access denied! You do not have permission.	没有权限访问。	请检查访问权限。
	ModelArts.4204	Request api error! The API XXX is not subscribed.	请求服务失败，服务未开通。	• 请前往**文字识别**控制台开通需要调用的API，注意API的开通区域。 • 如已开通服务，请检查开通服务的区域（或账号）与调用服务的区域（或账号）是否一致；若一致请检查API的URL是否拼写正确。
	ModelArts.4301	Have error when get ai-service admin token.	获取服务admin Token失败。	请联系华为云支持人员。
	ModelArts.4302	Gateway forwarding error.	访问服务失败。	请联系华为云支持人员。
	ModelArts.4401	Max concurrency error.	最大并发错误，请求并发过大，超过限额。	请**联系华为云支持人员**。
	ModelArts.4402	Backend service timeout error.	服务处理超时。	请**联系华为云支持人员**。
	ModelArts.4403	Backend service unavailable error.	服务无法正常响应，或者失败率过高，不可用。	请**联系华为云支持人员**。
	ModelArts.4502	Backend service response error.	服务响应失败。	请**联系华为云支持人员**。
	ModelArts.4503	Backend service found error.	服务不存在。	请**联系华为云支持人员**。
	ModelArts.4504	Backend service api not found.	访问的API不存在。	请**联系华为云支持人员**。

版权所有 © 华为云计算技术有限公司

模块	错误码	错误信息	说明	建议解决方法
	ModelArts.4505	Backend service internal error.	服务内部错误。	请**联系华为云支持人员**。
	ModelArts.4506	SubService service found error.	该API对应的子服务不存在。	请**联系华为云支持人员**。
	ModelArts.4508	Service not start.	服务不存在或者服务停止。	请**联系华为云支持人员**。
	ModelArts.4601	Failed to download the file because the file path is not valid, please check it format.	外网下载地址不合法。	请检查输入的url下载地址参数格式。
	ModelArts.4603	Obtaining the file from the URL failed.	下载外网URL文件失败。	请检查网络与URL。
	ModelArts.4702	Query Obs agency failed.	查询OBS委托失败。	请检查是否已开通服务的OBS委托。
	ModelArts.4703	The Obs URL is invalid.	OBS地址不合法。	请检查OBS地址。
	ModelArts.4704	Obtaining the file from the OBS failed.	获取OBS文件失败。	请检查OBS文件。
	ModelArts.4705	The file stored on the OBS is oversized.	OBS文件大小超限。	请检查OBS文件大小，请使用符合服务大小限制的文件作为输入。
	ModelArts.4706	The Obs file is not exist.	OBS文件不存在。	请检查对应的文件是否存在。
	ModelArts.6201	The user account has been suspended. Please check if you have an outstanding balance.	用户账户被冻结。	请检查账户是否被冻结。请检查账户余额并完成充值。

版权所有 © 华为云计算技术有限公司

模块	错误码	错误信息	说明	建议解决方法
APIG	APIG.0101	The API does not exist or has not been published in the environment.	访问的API不存在或尚未在环境中发布。	• 请检查API的URL是否拼写正确，例如，URL中是否缺少project_id。 • 检查URI中的区域信息是否和调用服务配置的**终端节点**是否对应。  • HTTP请求方法（POST，GET等）是否正确。 • 具体的URI信息请在各接口的API页面查看。
	APIG.0201	Backend timeout.	请求超时。	• 请检查原调用请求是否过于频繁，如果是并发过大，可以通过重试机制解决，在代码里检查返回值，碰到这个并发错误可以延时一小段时间（如2-5s）重试请求；也可以后端检查上一个请求结果，上一个请求返回之后再发送下一个请求，避免请求过于频繁。 • 请检查图片是否过大，或网络延时过长，如果图片过大，在保证图片清晰度前提下可以对图片进行等比例压缩；如果网络延时较长，可以适当提升网络传输速度。

版权所有 © 华为云计算技术有限公司

模块	错误码	错误信息	说明	建议解决方法
	APIG.0301	Incorrect IAM authentication information.	IAM身份验证信息不正确： • decrypt token fail：token解析失败。 • token expires：token过期。 • verify aksk signature fail：AK/SK认证失败。 • x-auth-token not found：未找到x-auth-token参数。	• token解析失败，请检查获取token的方法，请求体信息是否填写正确，token是否正确；检查获取token的环境与调用的环境是否一致。 • token超时（token expires），请重新获取token，使用不过期的token。 • 请检查AK/SK是否正确（AK对应的SK错误，不匹配；AK/SK中多填了空格）。 • AK/SK频繁出现鉴权出错，连续错误5次以上，被锁定5分钟（5分钟内，则一直认为其是异常的鉴权请求），5分钟后解锁重新认证。 • 检查账号权限，是否欠费，被冻结等。 • 检查请求header参数X-Auth-Token是否拼写正确。
	APIG.0308	The throttling threshold has been reached: policy user over ratelimit,limit:XX,time:1 minute.	发送请求超过了服务的默认配置限流。	请参考**调用API时提示"APIG.0308"**方案进行解决。
其他	如若出现其他无法解决的错误码，请**联系华为云支持人员**。			

版权所有 © 华为云计算技术有限公司

7 附录

7.1 获取项目 ID

从控制台获取项目 ID

1. 登录**管理控制台**。
2. 鼠标移动到右上角的用户名上，在下拉列表中选择“我的凭证”。
3. 在“我的凭证 > API凭证”页面，可以查看用户名、账号名，在项目列表中查看项目。

图 7-1 查看项目 ID



多项目时，展开“所属区域”，从“项目ID”列获取子项目ID。

调用 API 获取项目 ID

获取项目ID的接口为“GET https://{endpoint}/v3/projects”，其中“{endpoint}”为IAM的终端节点。接口的认证鉴权请参见**认证鉴权**。

响应示例如下，例如，文字识别服务部署的区域为“cn-north-4”，响应消息体中查找“name”为“cn-north-4”，其中“projects”下的“id”即为项目ID。获取项目ID报错时，请检查{endpoint}是否填写正确。

```
GET https://iam.cn-north-4.myhuaweicloud.com/v3/projects
{
    "projects": [
        {
            "domain_id": "65382450e8f64ac0870cd180d14e684b",
            "is_domain": false,
            "parent_id": "65382450e8f64ac0870cd180d14e684b",
            "name": "cn-north-4",
```


版权所有 © 华为云计算技术有限公司

```
            "description": "",
            "links": {
                "next": null,
                "previous": null,
                "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22"
            },
            "id": "a4a5d4098fb4474fa22cd05f897d6b99",
            "enabled": true
        }
    ],
    "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects"
    }
}
```


7.2 获取账号 ID

在调用接口的时候，部分URL中需要填入账号ID（domain-id），所以需要先在管理控制台上获取到账号ID。账号ID获取步骤如下：

1. 注册并登录管理控制台。
2. 单击用户名，在下拉列表中单击“我的凭证”。
   在“我的凭证”页面的项目列表中查看账号ID。

图 7-2 查看账号 ID



7.3 配置 OBS 访问权限

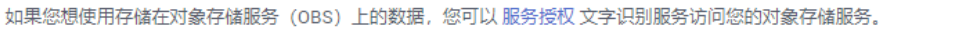
EI企业智能服务对于图片、语音等多媒体文件支持直接使用OBS服务的数据处理方式，以减少服务使用成本，降低服务的响应时长，提升服务使用的体验。

考虑到数据的安全，当对服务进行授权后，才可以使用授权方式的URL（https://<bucket-name>.<endpoint>/<object-name>）对服务进行访问。若未对服务授权，则无法直接获取到用户数据，需要用户开启公共读授权或者提供一个临时授权的URL。

对文字识别服务开启授权

如果您需要使用OBS中的数据，请开通对象存储服务OBS授权。进入文字识别控制台。打开对象存储服务OBS授权的按钮，完成授权操作。完成授权即可使用授权方式的URL对服务进行访问。

图 7-3 OBS 授权



版权所有 © 华为云计算技术有限公司

说明

不支持跨区域OBS，OBS的区域需要和服务保持一致。

OBS特性的使用限制请参考**约束与限制**。

开启公共读授权（可选）

配置公共读可参考《对象存储服务控制台指南》**配置标准桶策略**，创建桶策略为“**公共读**”。一般私密数据不建议用此方法。

“私有桶”策略（可选）

OCR识别OBS私有桶中的图片时需要保证调用OCR服务的账号和调用OBS服务的账号为同一个。

使用临时授权请求鉴权

开启公共读授权访问，虽然使用比较方便，但若对于敏感的信息，例如个人的私有数据，存在泄露风险。此场景下，可以考虑OBS提供的临时授权功能。

OBS服务支持用户对OBS服务中的对象构造一个特定URL，URL中会包含鉴权信息，任何用户都可以通过该URL访问OBS中的对象，但该URL只在Expires指定的失效时间内有效。该方式用于在不提供给其他人Secret Access Key的情况下，让其他人能够执行自己定义的操作。

进一步了解和使用OBS临时授权功能，请参见**《对象存储服务SDK参考》**对应语言的“授权访问”章节的相关内容，下载相关的SDK及示例代码，并进行相关的编码开发，以支持相关的URL获取。

版权所有 © 华为云计算技术有限公司