

ALIBABA CLOUD

阿里云

视觉智能开放平台
文字识别

文档版本：20200925

 阿里云

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
 危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 危险 重置操作将丢失用户配置数据。
 警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	 警告 重启操作将导致业务中断，恢复业务时间约十分钟。
 注意	用于警示信息、补充说明等，是用户必须了解的内容。	 注意 权重设置为0，该服务器不会再接受新请求。
 说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击设置> 网络> 设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击确定。
Courier字体	命令或代码。	执行 <code>cd /d C:/window</code> 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	<code>bae log list --instanceid</code> <i>Instance_ID</i>
[] 或者 [a b]	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
{ } 或者 {a b}	表示必选项，至多选择一个。	<code>switch {active stand}</code>

目录

1.文字识别介绍	06
2.个人类卡证识别	09
2.1. 身份证识别	09
2.2. 户口页识别	12
2.3. 银行卡识别	18
2.4. 名片识别	21
2.5. 二维码识别	24
2.6. 中国护照识别	28
2.7. 护照MRZ码识别	33
3.资产类证件识别	38
3.1. 营业执照识别	38
3.2. 公章识别	45
4.通用文字类识别	50
4.1. 通用识别	50
4.2. 表格识别	61
4.3. 验证码识别	84
4.4. 文档结构化还原识别	86
5.车辆交通类识别	89
5.1. 驾驶证识别	89
5.2. 行驶证识别	92
5.3. 车牌识别	96
5.4. VIN码识别	99
6.行业票证类识别	102
6.1. 出租车发票识别	102
6.2. 增值税发票识别	117
6.3. 火车票识别	128

6.4. 外卖单识别	130
7. 查询异步任务结果	139
8. 公共信息	145
8.1. 请求结构	145
8.2. 公共请求参数	145
8.3. 公共响应参数	146
9. 常见错误码	148

1.文字识别介绍

文字识别技术是基于阿里云深度学习技术，为您提供通用的印刷文字识别和文档结构化等能力。文字识别技术可以灵活应用于证件文字识别、发票文字识别、文档识别与整理等行业场景，满足认证、鉴权、票据流转审核等业务需求。

服务开通

请单击[立即开通](#)在对应产品页面开通该服务，具体开通方式请参见[开通视觉智能开放平台服务](#)。

能力介绍

目前阿里云视觉智能开放平台提供如下文字识别能力。

类别	能力	说明
个人类卡证识别	身份证识别	识别二代身份证关键字段内容，关键字段包括：姓名、性别、民族、身份证号、出生日期、地址信息、有效起始时间、签发机关，同时可输出身份证区域位置和人脸位置信息。
	户口页识别	支持户口本常住人口登记卡结构化识别，输出姓名、籍贯、民族、性别、身份证号、出生地、出生日期、户主或户主关系、承办人签章、注册章位置、其他印章位置，共11个关键字段信息。
	银行卡识别	输入主流银行卡图像，输出3个信息，包括：发卡行、银行卡号、有效日期。
	名片识别	识别各类名片关键字段内容。包括：姓名、职位、公司、部门、座机号、手机号、地址、邮箱，共8个关键字段信息。
	二维码识别	识别图像中是否含有二维码信息，输出图像中二维码包含的文本信息（每个二维码对应的url或文本），可支持图像中含有多个二维码识别。
	中国护照识别	识别中国护照关键字段内容，关键字段包括：中文姓名（带拼音）、护照号码、护照人身份ID、性别、英文姓名、出生日期、出生地（带拼音）、国籍、发证日期、结束日期、发证地址（带拼音）、MRZ码第一行、MRZ码第二行、护照类型。
	护照MRZ码识别	输入护照MRZ图像，输出11个信息，包括：类型、国家码、护照号、姓名、国籍、生日、性别、开始日期、结束日期、机器校验码1、机器校验码2，方便后续信息提取和证件审核。
资产类证件识别	营业执照识别	识别营业执照关键字段内容，包括：公司地址、营业范围、注册资本、注册日期、公司法人、公司名称、注册号、公司类型、公司营业期限日期，同时可输出营业执照上二维码、印章位置。
	公章识别	识别图像中公章的位置，输出公章的机关、团体、企事业单位名称。

类别	能力	说明
通用文字类识别	通用识别	识别图像中文字内容和文字区域坐标，适用于多场景图像文字识别。
	表格识别	自动识别表格位置、表格内容信息。适用于黑色的，横线、竖线都齐全的表格中内容识别。
	验证码识别	识别图像中验证码，输出验证码中的字母和数字。
	文档结构化还原识别	对输入文档内容进行解析，输出结构化样式（html或者json）。
车辆交通类识别	驾驶证识别	识别驾驶证首页和副页关键字段内容，包括：档案编号、姓名、有效期时长、性别、发证日期、驾驶证号、驾驶证准驾车型、有效期开始时间、地址，共9个关键字段信息。
	行驶证识别	识别行驶证首页和副页关键字段内容，输出品牌型号、车辆类型、车牌号码、检验记录、核定载质量、核定载人数等21个关键字段信息。
	车牌识别	准确识别出图像中车牌位置，输出车牌位置坐标、车牌类型、车牌号码、车牌号码置信度、车牌置信度，共5个关键字段信息。
	VIN码识别	识别车辆VIN码，输出车辆VIN码数值。
行业票证类识别	出租车发票识别	准确识别出全国各大城市出租车发票在图像中的位置，支持出租车发票结构化识别，输出发票号码、代码、车号、日期、时间、金额，共6个关键字段信息。
	火车票识别	支持火车票结构化识别，输出乘车时间、始发站、目的站、座位席别、乘车人姓名、车次号、票价、车厢及座次号，共8个关键字段信息。
	增值税发票识别	识别增值税发票（电子发票和纸质发票）关键字段内容，包括：校验码、复核人、开票人、发票代码、收款人等。
	外卖单识别	识别外卖单上的关键字段内容，输出商店名称、电话、包装费、配送费、商品合计、其他费用合计、顾客优惠合计、总件数、在线支付、订单编号、下单时间等，目前支持饿了么外卖单。

应用场景

文字识别应用场景如下：

- 注册登记审核

互联网时代，许多应用都需要通过证件审核，认证访客的真实信息，从而提供更安全和高级的服务。通过身份证、护照、银行卡等能力可以快速准确的识别上传的证件，高达99%的识别准确率，可以提升审核效率，降低人工审核录入成本。

- 报销凭证录入

解决各种金融，医疗以及企业日常报销等场景的票据识别问题，通过诸如增值税发票、出租车发票、火车票等细分场景里的票证识别能力，可以实现高效、快速、低成本的各种行业票剧等识别与录入能力。

平台将持续更新迭代更多视觉AI能力，请您持续关注[阿里云视觉智能开放平台](#)。

2. 个人类卡证识别

2.1. 身份证识别

本文档介绍身份证识别RecognizeIdentityCard的语法及示例。

功能描述

RecognizeIdentityCard可以识别二代身份证关键字段内容，关键字段包括：姓名、性别、民族、身份证号、出生日期、地址信息、有效起始时间、签发机关，同时可输出身份证区域位置和人脸位置信息。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeIdentityCard	要执行的操作。取值：RecognizeIdentityCard。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/7%E8%BA%AB%E4%BB%BD%E8%AF%81%E5%8E%9F%E5%9B%BE%20%281%29.jpeg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。
Side	String	是	face	身份证正反面类型。 <ul style="list-style-type: none"> • face：正面。 • back：反面。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
BackResult	Struct		反面照结果。
EndDate	String	19800101	有效期结束时间，格式：YYYYMMDD，例如19800101，即1980年01月01日。
Issue	String	杭州市公安局	签发机关。
StartDate	String	19700101	有效期起始时间，格式：YYYYMMDD，例如19800101，即1980年01月01日。
FrontResult	Struct		正面照结果。
Address	String	浙江省杭州市余杭区文一西路969号	地址信息。
BirthDate	String	20000101	出生日期，格式：YYYYMMDD，例如19800101，即1980年01月01日。
CardAreas	Array of CardArea		身份证区域位置，四个顶点表示，顺序是逆时针（左上、左下、右下、右上）。
X	Float	165	身份证区域横坐标。
Y	Float	657	身份证区域纵坐标。
FaceRectVertices	Array of FaceRectVertices		人脸位置，四个顶点表示。
X	Float	1024.6600341796875	人脸位置横坐标。
Y	Float	336.629638671875	人脸位置纵坐标。
FaceRectangle	Struct		人脸位置。
Angle	Float	-90	表示矩形顺时针旋转的度数，范围-180~180。

名称	类型	示例值	描述
Center	Struct		人脸矩形中心坐标。
X	Float	952	人脸矩形中心横坐标。
Y	Float	325.5	人脸矩形中心纵坐标。
Size	Struct		人脸矩形尺寸。
Height	Float	181.99	高度。
Width	Float	164.99	宽度。
Gender	String	男	性别。
IDNumber	String	1234567890	身份证号。
Name	String	张三	姓名。
Nationality	String	汉	民族。
RequestId	String	D6C24839-91A7-41DA-B31F-98F08EF80CC0	请求ID。

示例

请求示例

```
http(s)://[Endpoint]/?Action=RecognizeIdentityCard
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/7%E8%BA%AB%E4%BB%BD%E8%AF%81%E5%8E%9F%E5%9B%BE%20%281%29.jpeg
&Side=face
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>7E19E2CA-0BD2-4A3D-A96A-BA7CB5A8E5CD</RequestId>
<Data>
  <FrontResult>
    <Address>河北省邯郸市肥乡县肥乡镇</Address>
    <Gender>男</Gender>
    <Nationality>汉</Nationality>
    <Name>常涛</Name>
    <IDNumber>411526199706013217</IDNumber>
    <BirthDate>19970601</BirthDate>
  </FrontResult>
</Data>
```

JSON 格式

```
{
  "RequestId": "7E19E2CA-0BD2-4A3D-A96A-BA7CB5A8E5CD",
  "Data": {
    "FrontResult": {
      "Address": "河北省邯郸市肥乡县肥乡镇",
      "Gender": "男",
      "Nationality": "汉",
      "Name": "常涛",
      "IDNumber": "411526199706013217",
      "BirthDate": "19970601"
    }
  }
}
```

错误码

关于身份证识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会由系统自动清理删除。

2.2. 户口簿识别

本文档介绍户口簿识别RecognizeAccountPage的语法及示例。

功能描述

RecognizeAccountPage可以对户口本常住人口登记卡进行结构化识别，输出姓名、籍贯、民族、性别、身份证号、出生地、出生日期、户主或户主关系、承办人签章、注册章位置、其他印章位置，共11个关键字段信息。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeAccountPage	要执行的操作。取值：RecognizeAccountPage。
ImageURL	String	是	https://viapi-demo.oss-cn-shanghai.aliyuncs.com/viapi-demo/images/DetectImageElements/xxx.png	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Angle	Float	0	图片顺时针旋转角度，枚举类型：0度、90度、180度、270度。
BirthDate	String	2002年12月27日	出生日期，格式：xxxx年x月x日。识别失败时返回 FailInRecognition。

名称	类型	示例值	描述
BirthPlace	String	四川省雷波县	出生地，识别失败时返回 FailInRecognition 。
Gender	String	"Gender": "女",	性别，识别失败时返回 FailInRecognition 。
IDNumber	String	513437200212271325	公民身份证件编号，识别失败时返回 FailInRecognition 。
InvalidStampAreas	Array of InvalidStampArea		注销或作废印章位置列表（识别失败时，则不显示该字段）。
Height	Integer	10	注销或作废印章高度。
Left	Integer	10	注销或作废印章左上角x坐标。
Top	Integer	10	注销或作废印章左上角y坐标。
Width	Integer	10	注销或作废印章宽度。
Name	String	杨冬	姓名，识别失败时返回 FailInRecognition 。
Nationality	String	彝族	民族，识别失败时返回 FailInRecognition 。
NativePlace	String	四川省雷波县	籍贯，识别失败时返回 FailInRecognition 。
OtherStampAreas	Array of OtherStampArea		其他印章位置列表（识别失败时，则不显示该字段）。
Height	Integer	10	其他印章高度。
Left	Integer	10	其他印章左上角x坐标。
Top	Integer	10	其他印章左上角y坐标。

名称	类型	示例值	描述
Width	Integer	10	其他印章宽度。
RegisterStamp Areas	Array of RegisterStamp Area		大圆形登记注册章位置列表（识别失败时，则不显示该字段）。
Height	Integer	138	登记注册章高度。
Left	Integer	330	登记注册章左上角x坐标。
Top	Integer	2215	登记注册章左上角y坐标。
Width	Integer	152	登记注册章宽度。
Relation	String	孙女	户主或与户主关系，识别失败时返回 <code>FailInRecognition</code> 。
TitleArea	Struct		标题区域位置，识别失败时返回 <code>FailInDetection</code> 。
Height	Integer	30	标题区域高度。
Left	Integer	205	标题区域左上角x坐标。
Top	Integer	17	标题区域左上角y坐标。
Width	Integer	30	标题区域宽度。
UndertakeStampAreas	Array of UndertakeStampArea		承办人签章位置列表（识别失败时，则不显示该字段）。
Height	Integer	10	承办人签章高度。
Left	Integer	10	承办人签章左上角x坐标。
Top	Integer	10	承办人签章左上角y坐标。

名称	类型	示例值	描述
Width	Integer	10	承办人签章宽度。
RequestId	String	8D588F22-3E14-4674-9EAE-49AE1BA10EA2	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeAccountPage
&ImageURL=https://viapi-demo.oss-cn-shanghai.aliyuncs.com/viapi-demo/images/DetectImageElements/xxx.png
&<公共请求参数>
```

正常返回示例

XML 格式


```
<RequestId>8D588F22-3E14-4674-9EAE-49AE1BA10EA2</RequestId>
<Data>
  <NativePlace>四川省雷波县</NativePlace>
  <BirthPlace>四川省雷波县</BirthPlace>
  <RegisterStampAreas>
    <Left>330</Left>
    <Top>215</Top>
    <Height>138</Height>
    <Width>152</Width>
  </RegisterStampAreas>
  <Gender>女</Gender>
  <Nationality>彝族</Nationality>
  <Name>杨冬</Name>
  <Relation>孙女</Relation>
  <Angle>0</Angle>
  <TitleArea>
    <Left>205</Left>
    <Top>17</Top>
    <Height>30</Height>
    <Width>200</Width>
  </TitleArea>
  <BirthDate>2002年12月27日</BirthDate>
  <IDNumber>513437200212271325</IDNumber>
</Data>
```

JSON 格式

```
{
  "RequestId": "8D588F22-3E14-4674-9EAE-49AE1BA10EA2",
  "Data": {
    "NativePlace": "四川省雷波县",
    "BirthPlace": "四川省雷波县",
    "RegisterStampAreas": [
      {
        "Left": 330,
        "Top": 215,
        "Height": 138,
        "Width": 152
      }
    ],
    "Gender": "女",
    "Nationality": "彝族",
    "Name": "杨冬",
    "Relation": "孙女",
    "Angle": 0,
    "TitleArea": {
      "Left": 205,
      "Top": 17,
      "Height": 30,
      "Width": 200
    },
    "BirthDate": "2002年12月27日",
    "IDNumber": "513437200212271325"
  }
}
```

错误码

关于户口页识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

2.3. 银行卡识别

本文档介绍银行卡识别RecognizeBankCard的语法及示例。

功能描述

RecognizeBankCard可以检测输入的主流银行卡图像，输出3个信息，包括：发卡行、银行卡号、有效日期。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeBankCard	要执行的操作。取值：RecognizeBankCard。
ImageUrl	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/3%E9%93%B6%E8%A1%8C%E5%8D%A1.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的数据内容。
BankName	String	中国工商银行	银行名称，识别失败则返回空字符串。
CardNumber	String	6212262315007683105	银行卡号。

名称	类型	示例值	描述
ValidDate	String	07/26	有效日期，识别失败则返回空字符串。如有多个，则用逗号分隔，例如 03/17,04/05。
RequestId	String	D9C7521-0367-42EE-9646-FD066CCADB26	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeBankCard
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/3%E9%93%B6%E8%A
1%8C%E5%8D%A1.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>FD9C7521-0367-42EE-9646-FD066CCADB26</RequestId>
<Data>
  <CardNumber>6212262315007683105</CardNumber>
  <BankName>中国工商银行</BankName>
  <ValidDate>07/26</ValidDate>
</Data>
```

JSON 格式

```
{
  "RequestId": "FD9C7521-0367-42EE-9646-FD066CCADB26",
  "Data": {
    "CardNumber": "6212262315007683105",
    "BankName": "中国工商银行",
    "ValidDate": "07/26"
  }
}
```

错误码

关于银行卡识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

2.4. 名片识别

本文档介绍名片识别RecognizeBusinessCard的语法及示例

功能描述

RecognizeBusinessCard可以识别各类名片关键字段内容，包括：姓名、职位、公司、部门、座机号、手机号、地址、邮箱共8个关键字段信息。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeBusinessCard	要执行的操作。取值：RecognizeBusinessCard。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/8%E5%90%8D%E7%89%87%E6%96%87%E5%AD%97%E8%AF%86%E5%88%AB-%282%29.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的数据内容。
Addresses	List	上海市浦东新区66号	名片上的地址，可能为空。
CellPhoneNumbers	List	13588888888	名片上的手机号码，可能为空。
Companies	List	泰隆物业有限公司	名片上的公司名称，可能为空。
Departments	List	企业服务部	名片上的部门名称，可能为空。
Emails	List	tll@vip.163.com	名片上的邮箱地址，可能为空。
Name	String	田立龙	名片上的姓名。
OfficePhoneNumbers	List	02188888888	名片上的座机号码，可能为空。
Titles	List	部长	名片上的职位名称，可能为空。
RequestId	String	0068957A-C493-481F-BA14-C3F89BF75BD4	请求ID。

示例

请求示例

```
http(s)://[Endpoint]/?Action=RecognizeBusinessCard
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/8%E5%90%8D%E7%89%87%E6%96%87%E5%AD%97%E8%AF%86%E5%88%AB-%282%29.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>0068957A-C493-481F-BA14-C3F89BF75BD4</RequestId>
<Data>
  <Addresses>:上海市浦东新区66号</Addresses>
  <Departments>企业服务部</Departments>
  <Companies>泰隆物业有限公司</Companies>
  <Emails>tll@vip.163.com</Emails>
  <OfficePhoneNumbers>02188888888</OfficePhoneNumbers>
  <CellPhoneNumbers>13588888888</CellPhoneNumbers>
  <Titles>部长</Titles>
  <Name>田立龙</Name>
</Data>
```

JSON 格式

```
{
  "RequestId": "0068957A-C493-481F-BA14-C3F89BF75BD4",
  "Data": {
    "Addresses": [
      "上海市浦东新区66号"
    ],
    "Departments": [
      "企业服务部"
    ],
    "Companies": [
      "泰隆物业有限公司"
    ],
    "Emails": [
      "tll@vip.163.com"
    ],
    "OfficePhoneNumbers": [
      "02188888888"
    ],
    "CellPhoneNumbers": [
      "13588888888"
    ],
    "Titles": [
      "部长"
    ],
    "Name": "田立龙"
  }
}
```

错误码

关于名片识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

2.5. 二维码识别

本文档介绍二维码识别RecognizeQrCode的语法及示例。

功能描述

RecognizeQrCode可以识别图像中是否含有二维码信息，输出图像中二维码包含的文本信息（每个二维码对应的URL或文本），可支持图像中含有多个二维码识别。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：PNG、JPG、JPEG、BMP、GIF、WEBP。
- 图片大小：大小不超过10MB。如您有大图需求，请提工单申请。
- 图片分辨率：建议不低于256×256像素，分辨率过低可能会影响识别效果。
- URL地址中不能包含中文字符。

检测说明

- 最长检测时间为6秒，如果在该时间限制内没有完成检测，系统会返回超时错误码。
- 图片下载时间限制为3秒，如果下载时间超过3秒，则返回下载超时。
- 图片检测接口响应时间依赖图片的下载时间。请保证被检测图片所在的存储服务稳定可靠，建议您使用阿里云OSS存储或者CDN缓存等。

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeQrCode	要执行的操作。取值：RecognizeQrCode。
Tasks.N.ImageURL	String	是	https://timgsa.baidu.com/timg?imagexxx.jpg	待检测图像的URL地址。JSON数组中的每个元素是一个图片检测任务结构体（image表）。最多支持10个元素，即最多同时对10张图片进行检测。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Elements	Array of Element		识别返回的元素信息。

名称	类型	示例值	描述
ImageURL	String	https://tingsa.baidu.com/timg?image&quality=80&size=b9999_10000&sec=1581673386305&di=23d9a0a85bf0dbd3a6779d0bf878474a&imgtype=0&src=http%3A%2F%2Fpicapi.zhitud.com%2Fphoto%2F486%2F65%2F53ECA.jpg	对应请求中的图像URL地址。
Results	Array of Result		返回结果。调用成功时返回结果中包含一个或多个元素。
Label	String	qrcode	<p>检测结果的分类，取值包括：</p> <ul style="list-style-type: none"> • normal: 正常图片。 • qrcode: 含二维码的图片。 • programCode: 含小程序码的图片。 <p> 说明 默认不识别小程序码。如果需要识别，请提工单申请。</p>
QrCodesData	List	http://weixin.qq.com/r/_UwoMKDEDP1zreRT9xnr	图片中含有二维码时，返回图片中二维码包含的文本信息（每个二维码对应的URL或者文本）。
Rate	Float	99.91	结果为该分类的概率，取值范围为 [0.00-100.00]。值越高，表示属于该分类的概率越大。
Suggestion	String	review	<p>建议您执行的操作，取值包括：</p> <ul style="list-style-type: none"> • pass: 图片正常，无需进行其余操作。 • review: 检测结果不确定，需要进行人工审核。 • block: 图片违规，建议执行进一步操作（如直接删除或做限制处理）。
TaskId	String	img18Uu1GBIkOp4V1pJa0SLI\$-1s827C	该检测任务的ID。

名称	类型	示例值	描述
RequestId	String	A53DC437-F883-4968-86D5-EB21FB044692	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeQrCode
&Tasks.1.ImageURL=https://timgsa.baidu.com/timg?image&quality=80&size=b9999_10000&sec=1581673386305&di=23d9a0a85bf0dbd3a6779d0bf878474a&imgtype=0&src=http%3A%2F%2Fpicapi.zhitud.com%2Fphoto%2F486%2F65%2F53ECA.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>A53DC437-F883-4968-86D5-EB21FB044692</RequestId>
<Data>
  <Elements>
    <TaskId>img18Uu1GBIkOp4V1pJa0SLI$-1s827C</TaskId>
    <Results>
      <Suggestion>review</Suggestion>
      <Rate>99.91</Rate>
      <QrCodesData>http://weixin.qq.com/r/_UwoMKDEDP1zreRT9xnr</QrCodesData>
      <Label>qrcode</Label>
    </Results>
    <ImageUrl>https://timgsa.baidu.com/timg?image&amp;quality=80&amp;size=b9999_10000&amp;sec=1581673386305&amp;di=23d9a0a85bf0dbd3a6779d0bf878474a&amp;imgtype=0&amp;src=http%3A%2F%2Fpicapi.zhitud.com%2Fphoto%2F486%2F65%2F53ECA.jpg</ImageUrl>
  </Elements>
</Data>
```

JSON 格式

```
{
  "RequestId": "A53DC437-F883-4968-86D5-EB21FB044692",
  "Data": {
    "Elements": [
      {
        "TaskId": "img18Uu1GBIkOp4V1pJa0SLI$-1s827C",
        "Results": [
          {
            "Suggestion": "review",
            "Rate": 99.91,
            "QrCodesData": [
              "http://weixin.qq.com/r/_UwoMKDEDP1zreRT9xnr"
            ],
            "Label": "qrcode"
          }
        ],
        "ImageURL": "https://timgsa.baidu.com/timg?image&quality=80&size=b9999_10000&sec=1581673386305&di=23d9a0a85bf0dbd3a6779d0bf878474a&imgtype=0&src=http%3A%2F%2Fpicapi.zhitud.com%2Fphoto%2F486%2F65%2F53ECA.jpg"
      }
    ]
  }
}
```

错误码

关于二维码识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

2.6. 中国护照识别

本文介绍中国护照识别RecognizeChinapassport的语法及示例。

功能描述

RecognizeChinapassport可以识别中国护照关键字段内容，包括：中文姓名（带拼音）、护照号码、护照人身份ID、性别、英文姓名、出生日期、出生地（带拼音）、国籍、发证日期、结束日期、发证地址（带拼音）、MRZ码第一行、MRZ码第二行、护照类型。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG、PNG、BMP、GIF。
- 图片大小：图片二进制数据的base64编码后大小不超过2MB。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeChina passport	要执行的操作。取值： RecognizeChinapassport。
ImageURL	String	是	https://viapi- test.oss-cn- shanghai.aliyun cs.com/test- team/passport.j pg	图片URL地址。当前仅支持上海地域的 OSS链接，如何生成URL请参见 生成 URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Authority	String	公安部出入境管理局	签发机关。
BirthDate	String	19861030	出生日期。
BirthDay	String	861030	生日。
BirthPlace	String	广西	出生地。
BirthPlaceRaw	String	广西/GUANGXI	出生地带拼音。

名称	类型	示例值	描述
Type	String	P0	护照类型。
RequestId	String	UUID	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeChinapassport
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/passport.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

功能描述

RecognizePassportMRZ可以检测输入的护照MRZ图像，输出11个信息，包括：类型、国家码、护照号、姓名、国籍、生日、性别、开始日期、结束日期、机器校验码1、机器校验码2，方便后续信息提取和证件审核。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、PNG、JPG。
- 图片大小：大小不超过10M。
- 分辨率：不限制分辨率，但是分辨率太高可能会导致API推理时间超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizePassportMRZ	要执行的操作。取值：RecognizePassportMRZ。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/passportMRZ.jpg	图片URL地址。URL地址中不能包含中文字符。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的数据内容。
Regions	Array of Region		输出识别MRZ码结果。
BandBoxes	List	[49, 571, 998, 603]	识别内容框位置。格式为 [x1,y1,x2,y2]，x1和y1为矩形框左上角点坐标，x2和y2为矩形框右下角点坐标。

归一化字段	中文含义
date_of_expiry	结束日期
mrz1	机器校验码1
mrz2	机器校验码2

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizePassportMRZ
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/passportMRZ.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>230EA60A-B983-4829-814D-207ECEC7F0C6</RequestId>
<Data>
  <Regions>
    <Name>mrz1</Name>
    <RecognitionScore>0.9916993379592896</RecognitionScore>
    <Content>P<MYSCHAN<WAI<LENG<
    <BandBoxes>49</BandBoxes>
    <BandBoxes>571</BandBoxes>
    <BandBoxes>998</BandBoxes>
    <BandBoxes>603</BandBoxes>
    <DetectionScore>0.59131116</DetectionScore>
  </Regions>
</Data>
```

JSON 格式

3. 资产类证件识别

3.1. 营业执照识别

本文档介绍营业执照识别RecognizeBusinessLicense的语法及示例。

功能描述

RecognizeBusinessLicense可以识别营业执照关键字段内容，包括：公司地址、营业范围、注册资本、注册日期、公司法人、公司名称、注册号、公司类型、公司营业期限日期。同时可以输出营业执照上二维码、印章位置。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 请求格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求参数


名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeBusinessLicense	要执行的操作。取值：RecognizeBusinessLicense。
ImageURL	String	是	https://viapi-demo.oss-cn-shanghai.aliyuncs.com/viapi-demo/images/RecognizeBusinessLicense/xxx.png	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。

名称	类型	示例值	描述
Address	String	浙江省杭州市西湖区转塘科技经济区块888号888幢	公司地址。识别失败时返回 <code>FailInRecognition</code> 。
Angle	Float	0	输入图片的角度（顺时针旋转）。枚举值包括：0、90、180、270。
Business	String	鲜冻畜禽产品、水产品、新鲜果蔬、初级农产品、农副产品、预包装食品、工艺品（文物除外）、文化用品、办公用品、日用百货、电子产品、电脑及配件、计算机软件、通信设备、照相器材、家用电器、钟表眼镜及配件、包装材料、鲜花、木材、皮革制品、服装鞋帽、服装服饰、针纺织品、金银饰品、化妆品、体育用品、家具、宠物用品、厨房用品、卫生间用品、日用杂货、玉器、玩具、保健用品、装潢材料、建筑材料、化原料及产品、印刷设备、卫生洁具，陶瓷制品、橡胶制品、制冷设备、金属材料、电线电缆、机电设备、仪器仪表、健身器材、汽摩配件、一类医疗用品和器械、批发、代购代销、网上销售，自营和代理各类商品和技术的进出口业务，但国家限制或禁止公司经营的商品和技术除外；设计、制作、代理及发布国内外各类广告。	经营范围，识别失败时返回 <code>FailInRecognition</code> 。
Capital	String	壹百万元整	注册资本，识别失败时返回 <code>FailInRecognition</code> 。

名称	类型	示例值	描述
Emblem	Struct		国徽位置，识别失败时返回 <code>FailInDetection</code> 。
Height	Integer	163	区域高度。
Left	Integer	366	区域左上角x坐标。
Top	Integer	8	区域左上角y坐标。
Width	Integer	162	区域宽度。
EstablishDate	String	20150504	公司注册日期。例如证件上为“2014年04月16日”，则接口返回“20140416”。
LegalPerson	String	李四	公司法人。识别失败时返回 <code>FailInRecognition</code> 。
Name	String	某某电子商务有限公司	公司名称。识别失败时返回 <code>FailInRecognition</code> 。
QRCode	Struct		二维码位置，识别失败时返回 <code>FailInDetection</code> 。
Height	Integer	132	区域高度。
Left	Integer	156	区域左上角x坐标。
Top	Integer	914	区域左上角y坐标。
Width	Integer	126	区域宽度。
RegisterNumber	String	中国	注册号。识别失败时返回 <code>FailInRecognition</code> 。

名称	类型	示例值	描述
Stamp	Struct		印章位置，识别失败时返回 <code>FailInDetection</code> 。
Height	Integer	154	区域高度。
Left	Integer	650	区域左上角x坐标。
Top	Integer	1030	区域左上角y坐标。
Width	Integer	154	区域宽度。
Title	Struct		标题位置，识别失败时返回 <code>FailInDetection</code> 。
Height	Integer	10	区域高度。
Left	Integer	10	区域左上角x坐标。
Top	Integer	10	区域左上角y坐标。
Width	Integer	10	区域宽度。
Type	String	有限责任公司	公司类型。识别失败时返回 <code>FailInRecognition</code> 。
ValidPeriod	String	29991231	<p>公司营业期限终止日期。例如证件上为“2014年04月16日至2034年04月15日”，则接口返回“20340415”。</p> <p> 说明 当前算法将日期格式统一输出为“年月日”（例如“20391130”），并用“29991231”表示“长期”，若证件上没有营业期限，则默认其为“长期”返回“29991231”。</p>
RequestId	String	0898E8C2-4608-401B-9FA2-xxxx6FA59807	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeBusinessLicense
&ImageURL=https://viapi-demo.oss-cn-shanghai.aliyuncs.com/viapi-demo/images/RecognizeBusiness
License/xxx.png
&<公共请求参数>
```

正常返回示例

XML 格式

```

<Data>
  <Stamp>
    <Left>650</Left>
    <Top>1030</Top>
    <Height>154</Height>
    <Width>154</Width>
  </Stamp>
  <ValidPeriod>29991231</ValidPeriod>
  <Address>浙江省杭州市西湖区转塘科技经济区块888号888幢</Address>
  <Capital>壹百万元整</Capital>
  <LegalPerson>李四</LegalPerson>
  <EstablishDate>20150504</EstablishDate>
  <Name>某某电子商务有限公司</Name>
  <RegisterNumber>中国</RegisterNumber>
  <QRCode>
    <Left>156</Left>
    <Top>914</Top>
    <Height>132</Height>
    <Width>126</Width>
  </QRCode>
  <Type>有限责任公司</Type>
  <Angle>0</Angle>
  <Emblem>
    <Left>366</Left>
    <Top>8</Top>
    <Height>163</Height>
    <Width>162</Width>
  </Emblem>
  <Business>鲜冻畜禽产品、水产品、新鲜果蔬、初级农产品、农副产品、预包装食品、工艺品（文物除外）、文化用品、办公用品、日用百货、电子产品、电脑及配件、计算机软件、通信设备、照相器材、家用电器、钟表眼镜及配件、包装材料、鲜花、木材、皮革制品、服装鞋帽、服装服饰、针纺织品、金银饰品、化妆品、体育用品、家具、宠物用品、厨房用品、卫生间用品、日用杂货、玉器、玩具、保健用品、装潢材料、建筑材料、化原料及产品、印刷设备、卫生洁具、陶瓷制品、橡制品、制冷设备、金属材料、电线电缆、机电设备、仪器仪表、健身器材、汽摩配件、一类医疗用品和器成、批发、代购代销、网上销售，自营和代理各类商品和技术的进出口业务，但国家限制或禁止公司经营的商品和技术除外；设计、制作、代理及发布国内外各类广告。（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）</Business>
</Data>

```

JSON 格式

```
{
  "Data": {
    "Stamp": {
      "Left": 650,
      "Top": 1030,
      "Height": 154,
      "Width": 154
    },
    "ValidPeriod": "29991231",
    "Address": "浙江省杭州市西湖区转塘科技经济区块888号888幢)",
    "Capital": "壹百万元整",
    "LegalPerson": "李四",
    "EstablishDate": "20150504",
    "Name": "某某电子商务有限公司",
    "RegisterNumber": "中国",
    "QRCode": {
      "Left": 156,
      "Top": 914,
      "Height": 132,
      "Width": 126
    },
    "Type": "有限责任公司",
    "Angle": 0,
    "Emblem": {
      "Left": 366,
      "Top": 8,
      "Height": 163,
      "Width": 162
    },
    "Business": "鲜冻畜禽产品、水产品、新鲜果蔬、初级农产品、农副产品、预包装食品、工艺品（文物除外）、文化用品、办公用品、日用百货、电子产品、电脑及配件、计算机软件、通信设备、照相器材、家用电器、钟表眼镜及配件、包装材料、鲜花、木材、皮革制品、服装鞋帽、服装服饰、针纺织品、金银饰品、化妆品、体育用品、家具、宠物用品、厨房用品、卫生间用品、日用杂货、玉器、玩具、保健用品、装潢材料、建筑材料、化原料及产品、印刷设备、卫生洁具、陶瓷制品、橡制品、制冷设备、金属材料、电线电缆、机电设备、仪器仪表、健身器材、汽摩配件、一类医疗用品和器成、批发、代购代销、网上销售，自营和代理各类商品和技术的进出口业务，但国家限制或禁止公司经营的商品和技术除外；设计、制作、代理及发布国内外各类广告。（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）"
  }
}
```

错误码

关于营业执照识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

3.2. 公章识别

本文档介绍公章识别RecognizeStamp的语法及示例。

功能描述

RecognizeStamp可以识别图像中公章的位置，输出公章的机关、团体、企事业单位名称。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

图片限制

- 请求格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeStamp	要执行的操作。取值：RecognizeStamp。
ImageURL	String	是	https://viapi-demo.oss-cn-shanghai.aliyuncs.com/viapi-demo/images/RecognizeStamp/xxx.png	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Results	Array of Results		印章列表详细信息。
GeneralText	Array of GeneralText		印章其他区域文字列表。
Confidence	Float	0.81859749555587769	其他区域文字概率，范围 0~1。
Content	String	0857-88888888	其他区域文字内容。
Roi	Struct		印章文字区域。
Height	Integer	200	文字区域高度。
Left	Integer	716	文字区域左上角x坐标。
Top	Integer	345	文字区域左上角y坐标。
Width	Integer	192	文字区域宽度。
Text	Struct		印章具体文字。
Confidence	Float	0.85911655426025391	印章文字概率，范围 0~1。
Content	String	余杭区测试音乐餐吧	印章文字内容。
RequestId	String	D6C24839-91A7-41DA-B31F-98F08EF80CC0	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeStamp
&ImageURL=https://viapi-demo.oss-cn-shanghai.aliyuncs.com/viapi-demo/images/RecognizeStamp/xx.png
&<公共请求参数>
```

正常返回示例

XML 格式

```
<Data>
  <Results>
    <Text>
      <Confidence>0.8591165542602539</Confidence>
      <Content>余杭区测试音乐餐吧</Content>
    </Text>
    <Roi>
      <Left>716</Left>
      <Top>345</Top>
      <Height>200</Height>
      <Width>192</Width>
    </Roi>
    <GeneralText>
      <Confidence>0.8185974955558777</Confidence>
      <Content>0857-88888888</Content>
    </GeneralText>
    <GeneralText>
      <Confidence>0.7384993433952332</Confidence>
      <Content>测试音乐餐吧（公章）</Content>
    </GeneralText>
    <GeneralText>
      <Confidence>0.16586235165596008</Confidence>
      <Content>1月12日起</Content>
    </GeneralText>
  </Results>
</Data>
```

JSON 格式

```
{
  "Data": {
    "Results": [
      {
        "Text": {
          "Confidence": 0.85911655426025391,
          "Content": "余杭区测试音乐餐吧"
        },
        "Roi": {
          "Left": 716,
          "Top": 345,
          "Height": 200,
          "Width": 192
        },
        "GeneralText": [
          {
            "Confidence": 0.81859749555587769,
            "Content": "0857-88888888"
          },
          {
            "Confidence": 0.73849934339523315,
            "Content": "测试音乐餐吧（公章）"
          },
          {
            "Confidence": 0.16586235165596008,
            "Content": "1月12日起"
          }
        ]
      }
    ]
  }
}
```

错误码

关于公章识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

4.通用文字类识别

4.1. 通用识别

本文档介绍通用识别RecognizeCharacter的语法及示例。

功能描述

RecognizeCharacter可以识别图像中文字内容和文字区域坐标，适用于多场景图像文字识别。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 请求格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeCharacter	要执行的操作。取值：RecognizeCharacter。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/demo-center/ocr/RecognizeCharacter.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。
MinHeight	Integer	是	10	图片中文字的最小高度，单位像素。
OutputProbability	Boolean	是	true	是否输出文字框的概率。取值true或者false。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Results	Array of Result		返回识别信息。
Probability	Float	0.99908673763275146	文字内容的概率，取值范围为0~1。
Text	String	6月9日	文字内容。
TextRectangles	Struct		文字框区域位置。
Angle	Integer	-89	文字区域角度，角度范围 [-180, 180] 。
Height	Integer	58	文字区域高度。
Left	Integer	126	文字区域左上角x坐标。
Top	Integer	26	文字区域左上角y坐标。
Width	Integer	17	文字区域宽度。
RequestId	String	062A6BA0-40AF-47D4-B2A3-664B1009141F	请求ID。

示例

请求示例

```
http(s)://[Endpoint]/?Action=RecognizeCharacter
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/demo-center/ocr/RecognizeCharacter.jpg
&MinHeight=10
&OutputProbability=true
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>062A6BA0-40AF-47D4-B2A3-664B1009141F</RequestId>
```

```
<Data>
  <Results>
    <TextRectangles>
      <Left>126</Left>
      <Angle>-89</Angle>
      <Top>26</Top>
      <Height>58</Height>
      <Width>17</Width>
    </TextRectangles>
    <Probability>0.99908673763275146</Probability>
    <Text>6月9日</Text>
  </Results>
  <Results>
    <TextRectangles>
      <Left>124</Left>
      <Angle>-89</Angle>
      <Top>-22</Top>
      <Height>198</Height>
      <Width>19</Width>
    </TextRectangles>
    <Probability>0.7280038595199585</Probability>
    <Text>名企商机千亿市场发布会</Text>
  </Results>
  <Results>
    <TextRectangles>
      <Left>129</Left>
      <Angle>-89</Angle>
      <Top>77</Top>
      <Height>61</Height>
      <Width>15</Width>
    </TextRectangles>
    <Probability>0.99313169717788696</Probability>
    <Text>邀请函</Text>
  </Results>
  <Results>
    <TextRectangles>
      <Left>72</Left>
      <Angle>-89</Angle>
      <Top>102</Top>
      <Height>86</Height>
      <Width>16</Width>
```

```
<width>10</width>
</TextRectangles>
<Probability>0.72483104467391968</Probability>
<Text>尊敬的先生/女士嘉宾</Text>
</Results>
<Results>
  <TextRectangles>
    <Left>57</Left>
    <Angle>-89</Angle>
    <Top>147</Top>
    <Height>22</Height>
    <Width>9</Width>
  </TextRectangles>
  <Probability>0.62350982427597046</Probability>
  <Text>您好!</Text>
</Results>
<Results>
  <TextRectangles>
    <Left>34</Left>
    <Angle>0</Angle>
    <Top>167</Top>
    <Height>9</Height>
    <Width>204</Width>
  </TextRectangles>
  <Probability>0.32332202792167664</Probability>
  <Text>现诚邀请您在临“6月9日名企商机千亿市场发布会</Text>
</Results>
<Results>
  <TextRectangles>
    <Left>35</Left>
    <Angle>0</Angle>
    <Top>206</Top>
    <Height>14</Height>
    <Width>113</Width>
  </TextRectangles>
  <Probability>0.7724418044090271</Probability>
  <Text>01新采购新趋势</Text>
</Results>
<Results>
  <TextRectangles>
    <Left>69</Left>
```

```
<Angle>-89</Angle>
<Top>188</Top>
<Height>76</Height>
<Width>7</Width>
</TextRectangles>
<Probability>0.19863292574882507</Probability>
<Text>*IW PURCHASE 1 1**</Text>
</Results>
<Results>
<TextRectangles>
<Left>35</Left>
<Angle>0</Angle>
<Top>243</Top>
<Height>14</Height>
<Width>154</Width>
</TextRectangles>
<Probability>0.82449477910995483</Probability>
<Text>02采购供应链+两化融合</Text>
</Results>
<Results>
<TextRectangles>
<Left>34</Left>
<Angle>0</Angle>
<Top>259</Top>
<Height>7</Height>
<Width>133</Width>
</TextRectangles>
<Probability>0.23317055404186249</Probability>
<Text>PEDCORENERT SUPPLY CEAIR TWO FOSIO</Text>
</Results>
<Results>
<TextRectangles>
<Left>90</Left>
<Angle>-89</Angle>
<Top>223</Top>
<Height>125</Height>
<Width>14</Width>
</TextRectangles>
<Probability>0.72289299964904785</Probability>
<Text>03阳光效益转型</Text>
</Results>
```

```
<Results>
  <TextRectangles>
    <Left>115</Left>
    <Angle>-89</Angle>
    <Top>236</Top>
    <Height>202</Height>
    <Width>10</Width>
  </TextRectangles>
  <Probability>0.72888237237930298</Probability>
  <Text>主办单位:网里巴巴(中国)网络技术有限公司</Text>
</Results>
<Results>
  <TextRectangles>
    <Left>90</Left>
    <Angle>-89</Angle>
    <Top>290</Top>
    <Height>121</Height>
    <Width>9</Width>
  </TextRectangles>
  <Probability>0.48480159044265747</Probability>
  <Text>主办部门:大企业采购+诚信通</Text>
</Results>
<Results>
  <TextRectangles>
    <Left>72</Left>
    <Angle>-89</Angle>
    <Top>320</Top>
    <Height>85</Height>
    <Width>10</Width>
  </TextRectangles>
  <Probability>0.99824750423431396</Probability>
  <Text>时间:2017年6月9日</Text>
</Results>
<Results>
  <TextRectangles>
    <Left>115</Left>
    <Angle>-89</Angle>
    <Top>285</Top>
    <Height>182</Height>
    <Width>9</Width>
  </TextRectangles>
  <Probability>0.99824750423431396</Probability>
  <Text>时间:2017年6月9日</Text>
</Results>
```

```
</TextRectangles>
<Probability>0.87808668613433838</Probability>
<Text>地址:中国浙江省杭州市滨江区网商路699号</Text>
</Results>
<Results>
  <TextRectangles>
    <Left>65</Left>
    <Angle>-89</Angle>
    <Top>353</Top>
    <Height>72</Height>
    <Width>10</Width>
  </TextRectangles>
  <Probability>0.77262163162231445</Probability>
  <Text>阿里巴巴集团总部</Text>
</Results>
</Data>
```

JSON 格式

```
{
  "RequestId": "062A6BA0-40AF-47D4-B2A3-664B1009141F",
  "Data": {
    "Results": [
      {
        "TextRectangles": {
          "Left": 126,
          "Angle": -89,
          "Top": 26,
          "Height": 58,
          "Width": 17
        },
        "Probability": "0.99908673763275146",
        "Text": "6月9日"
      },
      {
        "TextRectangles": {
          "Left": 124,
          "Angle": -89,
          "Top": -22,
          "Height": 198,
          "Width": 19
        }
      }
    ]
  }
}
```



```
},
  "Probability": "0.7280038595199585",
  "Text": "名企商机千亿市场发布会"
},
{
  "TextRectangles": {
    "Left": 129,
    "Angle": -89,
    "Top": 77,
    "Height": 61,
    "Width": 15
  },
  "Probability": "0.99313169717788696",
  "Text": "邀请函"
},
{
  "TextRectangles": {
    "Left": 72,
    "Angle": -89,
    "Top": 102,
    "Height": 86,
    "Width": 10
  },
  "Probability": "0.72483104467391968",
  "Text": "尊敬的先生/女士嘉宾"
},
{
  "TextRectangles": {
    "Left": 57,
    "Angle": -89,
    "Top": 147,
    "Height": 22,
    "Width": 9
  },
  "Probability": "0.62350982427597046",
  "Text": "您好!"
},
{
  "TextRectangles": {
    "Left": 34,
    "Angle": 0,
```

```
"Top": 167,
"Height": 9,
"Width": 204
},
"Probability": "0.32332202792167664",
"Text": "现诚邀请您在临“6月9日名企商机千亿市场发布会”
},
{
  "TextRectangles": {
    "Left": 35,
    "Angle": 0,
    "Top": 206,
    "Height": 14,
    "Width": 113
  },
  "Probability": "0.7724418044090271",
  "Text": "01新采购新趋势"
},
{
  "TextRectangles": {
    "Left": 69,
    "Angle": -89,
    "Top": 188,
    "Height": 76,
    "Width": 7
  },
  "Probability": "0.19863292574882507",
  "Text": "*IW PURCHASE 1 1****"
},
{
  "TextRectangles": {
    "Left": 35,
    "Angle": 0,
    "Top": 243,
    "Height": 14,
    "Width": 154
  },
  "Probability": "0.82449477910995483",
  "Text": "02采购供应链+两化融合"
},
{
```

```
{
  "TextRectangles": {
    "Left": 34,
    "Angle": 0,
    "Top": 259,
    "Height": 7,
    "Width": 133
  },
  "Probability": "0.23317055404186249",
  "Text": "PEDCORENERT SUPPLY CEAIR TWO FOSIO"
},
{
  "TextRectangles": {
    "Left": 90,
    "Angle": -89,
    "Top": 223,
    "Height": 125,
    "Width": 14
  },
  "Probability": "0.72289299964904785",
  "Text": "03阳光效益转型"
},
{
  "TextRectangles": {
    "Left": 115,
    "Angle": -89,
    "Top": 236,
    "Height": 202,
    "Width": 10
  },
  "Probability": "0.72888237237930298",
  "Text": "主办单位:网里巴巴(中国)网络技术有限公司"
},
{
  "TextRectangles": {
    "Left": 90,
    "Angle": -89,
    "Top": 290,
    "Height": 121,
    "Width": 9
  },
}
```

```
"Probability": "0.48480159044265747",
  "Text": "主办部门:大企业采购+诚信通"
},
{
  "TextRectangles": {
    "Left": 72,
    "Angle": -89,
    "Top": 320,
    "Height": 85,
    "Width": 10
  },
  "Probability": "0.99824750423431396",
  "Text": "时间:2017年6月9日"
},
{
  "TextRectangles": {
    "Left": 115,
    "Angle": -89,
    "Top": 285,
    "Height": 182,
    "Width": 9
  },
  "Probability": "0.87808668613433838",
  "Text": "地址:中国浙江省杭州市滨江区网商路699号"
},
{
  "TextRectangles": {
    "Left": 65,
    "Angle": -89,
    "Top": 353,
    "Height": 72,
    "Width": 10
  },
  "Probability": "0.77262163162231445",
  "Text": "阿里巴巴集团总部"
}
]
}
```

错误码

关于通用识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

4.2. 表格识别

本文档介绍表格识别RecognizeTable的语法及示例。

功能描述

RecognizeTable可以自动识别表格位置、表格内容信息。适用于黑色的，横线、竖线都齐全的表格中内容识别。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 请求格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeTable	要执行的操作。取值：RecognizeTable。
AssureDirection	Boolean	是	false	图片方向是否确定为正向。 <ul style="list-style-type: none">• true: 确定• false: 不确定

名称	类型	是否必选	示例值	描述
HasLine	Boolean	是	false	是否无线条。 <ul style="list-style-type: none"> • true: 无线条或者只有横线没有竖线。 • false: 有线条。
ImageURL	String	是	https://viapi-demo.oss-cn-shanghai-internal.aliyuncs.com/viapi-demo/images/RecognizeTable/%E8%A1%A8%E6%A0%BC.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。
OutputFormat	String	是	json	输出格式。包括：html、json、xlsx。
SkipDetection	Boolean	是	false	是否跳过检测，取值true或false。
UseFinanceModel	Boolean	是	true	是否使用财务报表模型。取值true或false。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
FileContent	String	-	HTML字符串。
Tables	Array of Table		JSON格式的数据。
Head	List	14 淮南矿 PPN003	表头信息。
TableRows	Array of TableRow		JSON格式的表格数据。
TableColumns	Array of TableColumn		JSON格式的表格数据。
EndColumn	Integer	1	单元格所占的列数（colspan）为ex-sx。

名称	类型	示例值	描述
EndRow	Integer	1	单元格所占的行数（rowspan）为ey-sy。
Height	Integer	0	图片上单元格的高度。
StartColumn	Integer	0	单元格的起始列ID。
StartRow	Integer	0	单元格的起始行ID。
Texts	List	项目	文本，每行（row）文字为一个block。
Width	Integer	0	图片上单元格的宽度。
Tail	List	职工券	表尾信息。
RequestId	String	ECC70342-7CF1-4EC9-AE5A-41923D10D552	请求ID。

示例

请求示例

```

http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeTable
&AssureDirection=false
&HasLine=false
&ImageURL=https://viapi-demo.oss-cn-shanghai-internal.aliyuncs.com/viapi-demo/images/RecognizeTable/%E8%A1%A8%E6%A0%BC.jpg
&OutputFormat=json
&SkipDetection=false
&UseFinanceModel=true
&<公共请求参数>

```

正常返回示例

XML 格式

```

<RequestId>ECC70342-7CF1-4EC9-AE5A-41923D10D552</RequestId>
<Data>
  <Tables>
    <Head>14 淮南矿 PPN003</Head>
    <Head>14 淮南矿 PPN004</Head>

```

```
<Head>14 淮南矿 MTNO01</Head>
<Head>15 淮南矿 PPN001</Head>
<Tail>15 淮南矿 PPN002</Tail>
<Tail>15 淮南矿 PPN003</Tail>
<Tail>12 芜湖港</Tail>
<Tail>职工券</Tail>
<TableRows>
  <TableColumns>
    <EndRow>1</EndRow>
    <EndColumn>1</EndColumn>
    <Texts>项目</Texts>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>0</StartRow>
    <StartColumn>0</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>1</EndRow>
    <EndColumn>2</EndColumn>
    <Texts>目</Texts>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>0</StartRow>
    <StartColumn>1</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>1</EndRow>
    <EndColumn>3</EndColumn>
    <Texts>年末余额</Texts>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>0</StartRow>
    <StartColumn>2</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>1</EndRow>
    <EndColumn>4</EndColumn>
    <Texts>年初余额</Texts>
    <Height>0</Height>
    <Width>0</Width>
```



```

<StartRow>0</StartRow>
  <StartColumn>3</StartColumn>
</TableColumns>
</TableRows>
<TableRows>
  <TableColumns>
    <EndRow>2</EndRow>
    <EndColumn>1</EndColumn>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>1</StartRow>
    <StartColumn>0</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>2</EndRow>
    <EndColumn>2</EndColumn>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>1</StartRow>
    <StartColumn>1</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>2</EndRow>
    <EndColumn>3</EndColumn>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>1</StartRow>
    <StartColumn>2</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>2</EndRow>
    <EndColumn>4</EndColumn>
    <Texts>1,993,528,348.75</Texts>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>1</StartRow>
    <StartColumn>3</StartColumn>
  </TableColumns>
</TableRows>
<TableRows>
  <TableColumns>

```

```
<EndRow>3</EndRow>
<EndColumn>1</EndColumn>
<Height>0</Height>
<Width>0</Width>
<StartRow>2</StartRow>
<StartColumn>0</StartColumn>
</TableColumns>
<TableColumns>
  <EndRow>3</EndRow>
  <EndColumn>2</EndColumn>
  <Height>0</Height>
  <Width>0</Width>
  <StartRow>2</StartRow>
  <StartColumn>1</StartColumn>
</TableColumns>
<TableColumns>
  <EndRow>3</EndRow>
  <EndColumn>3</EndColumn>
  <Texts>298,063,697.36</Texts>
  <Height>0</Height>
  <Width>0</Width>
  <StartRow>2</StartRow>
  <StartColumn>2</StartColumn>
</TableColumns>
<TableColumns>
  <EndRow>3</EndRow>
  <EndColumn>4</EndColumn>
  <Texts>3,987,056,697.49</Texts>
  <Height>0</Height>
  <Width>0</Width>
  <StartRow>2</StartRow>
  <StartColumn>3</StartColumn>
</TableColumns>
</TableRows>
<TableRows>
  <TableColumns>
    <EndRow>4</EndRow>
    <EndColumn>1</EndColumn>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>3</StartRow>
```

```
</StartRow>3</StartRow>
  <StartColumn>0</StartColumn>
</TableColumns>
<TableColumns>
  <EndRow>4</EndRow>
  <EndColumn>2</EndColumn>
  <Height>0</Height>
  <Width>0</Width>
  <StartRow>3</StartRow>
  <StartColumn>1</StartColumn>
</TableColumns>
<TableColumns>
  <EndRow>4</EndRow>
  <EndColumn>3</EndColumn>
  <Texts>2,989,029,381.93</Texts>
  <Height>0</Height>
  <Width>0</Width>
  <StartRow>3</StartRow>
  <StartColumn>2</StartColumn>
</TableColumns>
<TableColumns>
  <EndRow>4</EndRow>
  <EndColumn>4</EndColumn>
  <Texts>297,152,184.91</Texts>
  <Height>0</Height>
  <Width>0</Width>
  <StartRow>3</StartRow>
  <StartColumn>3</StartColumn>
</TableColumns>
</TableRows>
<TableRows>
  <TableColumns>
    <EndRow>5</EndRow>
    <EndColumn>1</EndColumn>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>4</StartRow>
    <StartColumn>0</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>5</EndRow>
```

```
<EndColumn>2</EndColumn>
<Height>0</Height>
<Width>0</Width>
<StartRow>4</StartRow>
<StartColumn>1</StartColumn>
</TableColumns>
<TableColumns>
  <EndRow>5</EndRow>
  <EndColumn>3</EndColumn>
  <Texts>1,991,116,621.19</Texts>
  <Height>0</Height>
  <Width>0</Width>
  <StartRow>4</StartRow>
  <StartColumn>2</StartColumn>
</TableColumns>
<TableColumns>
  <EndRow>5</EndRow>
  <EndColumn>4</EndColumn>
  <Texts>2,979,665,189.68</Texts>
  <Height>0</Height>
  <Width>0</Width>
  <StartRow>4</StartRow>
  <StartColumn>3</StartColumn>
</TableColumns>
</TableRows>
<TableRows>
  <TableColumns>
    <EndRow>6</EndRow>
    <EndColumn>1</EndColumn>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>5</StartRow>
    <StartColumn>0</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>6</EndRow>
    <EndColumn>2</EndColumn>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>5</StartRow>
    <StartColumn>1</StartColumn>
```

```

</TableColumns>
<TableColumns>
  <EndRow>6</EndRow>
  <EndColumn>3</EndColumn>
  <Texts>994,500,600.03</Texts>
  <Height>0</Height>
  <Width>0</Width>
  <StartRow>5</StartRow>
  <StartColumn>2</StartColumn>
</TableColumns>
<TableColumns>
  <EndRow>6</EndRow>
  <EndColumn>4</EndColumn>
  <Texts>1,984,968,998.90</Texts>
  <Height>0</Height>
  <Width>0</Width>
  <StartRow>5</StartRow>
  <StartColumn>3</StartColumn>
</TableColumns>
</TableRows>
<TableRows>
  <TableColumns>
    <EndRow>7</EndRow>
    <EndColumn>1</EndColumn>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>6</StartRow>
    <StartColumn>0</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>7</EndRow>
    <EndColumn>2</EndColumn>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>6</StartRow>
    <StartColumn>1</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>7</EndRow>
    <EndColumn>3</EndColumn>
    <Texts>1,484,378,418.40</Texts>

```

```
<TEXT>1,474,270,410.47</TEXT>  
<Height>0</Height>  
<Width>0</Width>  
<StartRow>6</StartRow>  
<StartColumn>2</StartColumn>  
</TableColumns>  
<TableColumns>  
<EndRow>7</EndRow>  
<EndColumn>4</EndColumn>  
<Texts>991,495,233.24</Texts>  
<Height>0</Height>  
<Width>0</Width>  
<StartRow>6</StartRow>  
<StartColumn>3</StartColumn>  
</TableColumns>  
</TableRows>  
<TableRows>  
<TableColumns>  
<EndRow>8</EndRow>  
<EndColumn>1</EndColumn>  
<Height>0</Height>  
<Width>0</Width>  
<StartRow>7</StartRow>  
<StartColumn>0</StartColumn>  
</TableColumns>  
<TableColumns>  
<EndRow>8</EndRow>  
<EndColumn>2</EndColumn>  
<Height>0</Height>  
<Width>0</Width>  
<StartRow>7</StartRow>  
<StartColumn>1</StartColumn>  
</TableColumns>  
<TableColumns>  
<EndRow>8</EndRow>  
<EndColumn>3</EndColumn>  
<Texts>789,105,219.30</Texts>  
<Height>0</Height>  
<Width>0</Width>  
<StartRow>7</StartRow>  
<StartColumn>2</StartColumn>
```

```
</TableColumns>
<TableColumns>
  <EndRow>8</EndRow>
  <EndColumn>4</EndColumn>
  <Texts>1,496,800,438.30</Texts>
  <Height>0</Height>
  <Width>0</Width>
  <StartRow>7</StartRow>
  <StartColumn>3</StartColumn>
</TableColumns>
</TableRows>
<TableRows>
  <TableColumns>
    <EndRow>9</EndRow>
    <EndColumn>1</EndColumn>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>8</StartRow>
    <StartColumn>0</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>9</EndRow>
    <EndColumn>2</EndColumn>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>8</StartRow>
    <StartColumn>1</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>9</EndRow>
    <EndColumn>3</EndColumn>
    <Texts>61,590.70</Texts>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>8</StartRow>
    <StartColumn>2</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>9</EndRow>
    <EndColumn>4</EndColumn>
    <Texts>61,590.70</Texts>
```

```

    <Height>0</Height>
    <Width>0</Width>
    <StartRow>8</StartRow>
    <StartColumn>3</StartColumn>
  </TableColumns>
</TableRows>
<TableRows>
  <TableColumns>
    <EndRow>10</EndRow>
    <EndColumn>1</EndColumn>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>9</StartRow>
    <StartColumn>0</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>10</EndRow>
    <EndColumn>2</EndColumn>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>9</StartRow>
    <StartColumn>1</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>10</EndRow>
    <EndColumn>3</EndColumn>
    <Texts>10,551,553,801.53</Texts>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>9</StartRow>
    <StartColumn>2</StartColumn>
  </TableColumns>
  <TableColumns>
    <EndRow>10</EndRow>
    <EndColumn>4</EndColumn>
    <Texts>18,618,966,233.02</Texts>
    <Height>0</Height>
    <Width>0</Width>
    <StartRow>9</StartRow>
    <StartColumn>3</StartColumn>
  </TableColumns>

```



```
</TableColumns>
</TableRows>
</Tables>
</Data>
```

JSON 格式

```
{
  "RequestId": "ECC70342-7CF1-4EC9-AE5A-41923D10D552",
  "Data": {
    "Tables": [
      {
        "Head": [
          "14 淮南矿 PPN003",
          "14 淮南矿 PPN004",
          "14 淮南矿 MTNO01",
          "15淮南矿 PPN001"
        ],
        "Tail": [
          "15 淮南矿 PPNO02",
          "15淮南矿 PPN003",
          "12芜湖港",
          "职工券"
        ],
        "TableRows": [
          {
            "TableColumns": [
              {
                "EndRow": 1,
                "EndColumn": 1,
                "Texts": [
                  "项目"
                ],
                "Height": 0,
                "Width": 0,
                "StartRow": 0,
                "StartColumn": 0
              },
              {
                "EndRow": 1,
                "EndColumn": 2,
```

```
"Texts": [
  "目"
],
"Height": 0,
"Width": 0,
"StartRow": 0,
"StartColumn": 1
},
{
  "EndRow": 1,
  "EndColumn": 3,
  "Texts": [
    "年末余额"
  ],
  "Height": 0,
  "Width": 0,
  "StartRow": 0,
  "StartColumn": 2
},
{
  "EndRow": 1,
  "EndColumn": 4,
  "Texts": [
    "年初余额"
  ],
  "Height": 0,
  "Width": 0,
  "StartRow": 0,
  "StartColumn": 3
}
]
},
{
  "TableColumns": [
    {
      "EndRow": 2,
      "EndColumn": 1,
      "Height": 0,
      "Width": 0,
      "StartRow": 1,
      "StartColumn": 0
```

```
},
{
  "EndRow": 2,
  "EndColumn": 2,
  "Height": 0,
  "Width": 0,
  "StartRow": 1,
  "StartColumn": 1
},
{
  "EndRow": 2,
  "EndColumn": 3,
  "Height": 0,
  "Width": 0,
  "StartRow": 1,
  "StartColumn": 2
},
{
  "EndRow": 2,
  "EndColumn": 4,
  "Texts": [
    "1,993,528,348.75"
  ],
  "Height": 0,
  "Width": 0,
  "StartRow": 1,
  "StartColumn": 3
}
]
},
{
  "TableColumns": [
    {
      "EndRow": 3,
      "EndColumn": 1,
      "Height": 0,
      "Width": 0,
      "StartRow": 2,
      "StartColumn": 0
    }
  ]
},
}
```

```
{
  "EndRow": 3,
  "EndColumn": 2,
  "Height": 0,
  "Width": 0,
  "StartRow": 2,
  "StartColumn": 1
},
{
  "EndRow": 3,
  "EndColumn": 3,
  "Texts": [
    "298,063,697.36"
  ],
  "Height": 0,
  "Width": 0,
  "StartRow": 2,
  "StartColumn": 2
},
{
  "EndRow": 3,
  "EndColumn": 4,
  "Texts": [
    "3,987,056,697.49"
  ],
  "Height": 0,
  "Width": 0,
  "StartRow": 2,
  "StartColumn": 3
}
]
},
{
  "TableColumns": [
    {
      "EndRow": 4,
      "EndColumn": 1,
      "Height": 0,
      "Width": 0,
      "StartRow": 3,
      "StartColumn": 0
```

```
},
{
  "EndRow": 4,
  "EndColumn": 2,
  "Height": 0,
  "Width": 0,
  "StartRow": 3,
  "StartColumn": 1
},
{
  "EndRow": 4,
  "EndColumn": 3,
  "Texts": [
    "2,989,029,381.93"
  ],
  "Height": 0,
  "Width": 0,
  "StartRow": 3,
  "StartColumn": 2
},
{
  "EndRow": 4,
  "EndColumn": 4,
  "Texts": [
    "297,152,184.91"
  ],
  "Height": 0,
  "Width": 0,
  "StartRow": 3,
  "StartColumn": 3
}
]
},
{
  "TableColumns": [
    {
      "EndRow": 5,
      "EndColumn": 1,
      "Height": 0,
      "Width": 0,
      "StartRow": 4,
```

```
    "StartColumn": 0
  },
  {
    "EndRow": 5,
    "EndColumn": 2,
    "Height": 0,
    "Width": 0,
    "StartRow": 4,
    "StartColumn": 1
  },
  {
    "EndRow": 5,
    "EndColumn": 3,
    "Texts": [
      "1,991,116,621.19"
    ],
    "Height": 0,
    "Width": 0,
    "StartRow": 4,
    "StartColumn": 2
  },
  {
    "EndRow": 5,
    "EndColumn": 4,
    "Texts": [
      "2,979,665,189.68"
    ],
    "Height": 0,
    "Width": 0,
    "StartRow": 4,
    "StartColumn": 3
  }
]
},
{
  "TableColumns": [
    {
      "EndRow": 6,
      "EndColumn": 1,
      "Height": 0,
      "Width": 0
```

```
width : 0,
  "StartRow": 5,
  "StartColumn": 0
},
{
  "EndRow": 6,
  "EndColumn": 2,
  "Height": 0,
  "Width": 0,
  "StartRow": 5,
  "StartColumn": 1
},
{
  "EndRow": 6,
  "EndColumn": 3,
  "Texts": [
    "994,500,600.03"
  ],
  "Height": 0,
  "Width": 0,
  "StartRow": 5,
  "StartColumn": 2
},
{
  "EndRow": 6,
  "EndColumn": 4,
  "Texts": [
    "1,984,968,998.90"
  ],
  "Height": 0,
  "Width": 0,
  "StartRow": 5,
  "StartColumn": 3
}
]
},
{
  "TableColumns": [
    {
      "EndRow": 7,
      "EndColumn": 1,
```

```
"Height": 0,
"Width": 0,
"StartRow": 6,
"StartColumn": 0
},
{
  "EndRow": 7,
  "EndColumn": 2,
  "Height": 0,
  "Width": 0,
  "StartRow": 6,
  "StartColumn": 1
},
{
  "EndRow": 7,
  "EndColumn": 3,
  "Texts": [
    "1,494,278,418.49"
  ],
  "Height": 0,
  "Width": 0,
  "StartRow": 6,
  "StartColumn": 2
},
{
  "EndRow": 7,
  "EndColumn": 4,
  "Texts": [
    "991,495,233.24"
  ],
  "Height": 0,
  "Width": 0,
  "StartRow": 6,
  "StartColumn": 3
}
]
},
{
  "TableColumns": [
    {
      "EndRow": 8,
```



```
"EndColumn": 1,
"Height": 0,
"Width": 0,
"StartRow": 7,
"StartColumn": 0
},
{
"EndRow": 8,
"EndColumn": 2,
"Height": 0,
"Width": 0,
"StartRow": 7,
"StartColumn": 1
},
{
"EndRow": 8,
"EndColumn": 3,
"Texts": [
"789,105,219.30"
],
"Height": 0,
"Width": 0,
"StartRow": 7,
"StartColumn": 2
},
{
"EndRow": 8,
"EndColumn": 4,
"Texts": [
"1,496,800,438.30"
],
"Height": 0,
"Width": 0,
"StartRow": 7,
"StartColumn": 3
}
]
},
{
"TableColumns": [
```

```
{
  "EndRow": 9,
  "EndColumn": 1,
  "Height": 0,
  "Width": 0,
  "StartRow": 8,
  "StartColumn": 0
},
{
  "EndRow": 9,
  "EndColumn": 2,
  "Height": 0,
  "Width": 0,
  "StartRow": 8,
  "StartColumn": 1
},
{
  "EndRow": 9,
  "EndColumn": 3,
  "Texts": [
    "61,590.70"
  ],
  "Height": 0,
  "Width": 0,
  "StartRow": 8,
  "StartColumn": 2
},
{
  "EndRow": 9,
  "EndColumn": 4,
  "Texts": [
    "61,590.70"
  ],
  "Height": 0,
  "Width": 0,
  "StartRow": 8,
  "StartColumn": 3
}
]
},
{
```

```
"TableColumns": [  
  {  
    "EndRow": 10,  
    "EndColumn": 1,  
    "Height": 0,  
    "Width": 0,  
    "StartRow": 9,  
    "StartColumn": 0  
  },  
  {  
    "EndRow": 10,  
    "EndColumn": 2,  
    "Height": 0,  
    "Width": 0,  
    "StartRow": 9,  
    "StartColumn": 1  
  },  
  {  
    "EndRow": 10,  
    "EndColumn": 3,  
    "Texts": [  
      "10,551,553,801.53"  
    ],  
    "Height": 0,  
    "Width": 0,  
    "StartRow": 9,  
    "StartColumn": 2  
  },  
  {  
    "EndRow": 10,  
    "EndColumn": 4,  
    "Texts": [  
      "18,618,966,233.02"  
    ],  
    "Height": 0,  
    "Width": 0,  
    "StartRow": 9,  
    "StartColumn": 3  
  }  
]  
}
```

```
    ]  
  }  
]  
}  
}
```

错误码

关于表格识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

4.3. 验证码识别

本文档介绍验证码识别RecognizeVerificationcode的语法及示例。

功能描述

RecognizeVerificationcode可以识别图像中验证码，输出验证码中的字母和数字。

 **说明** 图像中的验证码必须是大小写英文字母或数字。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPRG、JPG、PNG。
- 图片大小：大小不超过10M。
- 图片分辨率：不限制图片分辨率，但是分辨率太高可能会导致API推理时间超时，推理超时时间5秒。
- URL地址中不能包含中文字符。

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
----	----	------	-----	----

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeVerificationcode	要执行的操作。取值：RecognizeVerificationcode。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/verificationcode.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Content	String	5246	识别的验证码内容。
RequestId	String	12F2C1FF-16D6-4B71-B957-F6C4B4761E95	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeVerificationcode
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/verificationcode.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>12F2C1FF-16D6-4B71-B957-F6C4B4761E95</RequestId>
<Data>
  <Content>5246</Content>
</Data>
```

JSON 格式

```
{
  "RequestId": "12F2C1FF-16D6-4B71-B957-F6C4B4761E95",
  "Data": {
    "Content": "5246"
  }
}
```

错误码

关于验证码识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会系统自动清理删除。

4.4. 文档结构化还原识别

本文档介绍文档结构化还原识别TrimDocument的语法及示例。

功能描述

TrimDocument可以对输入文档内容进行解析，输出结构化样式（html或者json）。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 文档类型：当前支持pdf类型。
- 文档大小：大小不超过10M。
- URL地址中不能包含中文字符。

调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	TrimDocument	要执行的操作。取值：TrimDocument。

名称	类型	是否必选	示例值	描述
FileType	String	是	pdf	输入文档类型，目前仅支持pdf。
FileURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test/ant_ai/ant_doc_scanner/01.pdf	输入文档URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。
OutputType	String	是	html	识别结果输出类型，体现在结果中content字段的值。可选json或html。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Content	String	参见返回示例。	解析文档得到的具体结构化内容。
RequestId	String	E700C1C1-57E4-413A-9987-56A5A0443CE4	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=TrimDocument
&FileType=pdf
&FileURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test/ant_ai/ant_doc_scanner/01.pdf
&OutputType=html
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>E700C1C1-57E4-413A-9987-56A5A0443CE4</RequestId>
```

JSON 格式

```
{  
  "RequestId": "E700C1C1-57E4-413A-9987-56A5A0443CE4"  
}
```

查询任务结果

该接口为异步接口，当前并未返回真实的请求结果，您需要通过返回的RequestId调用GetAsyncJobResult接口来获取该接口的真实请求结果。详情请参见[GetAsyncJobResult](#)。

错误码

关于文档结构化还原识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

5. 车辆交通类识别

5.1. 驾驶证识别

本文档介绍驾驶证识别RecognizeDriverLicense的语法及示例。

功能描述

RecognizeDriverLicense可以识别驾驶证首页和副页关键字段内容，包括：档案编号、姓名、有效期时长、性别、发证日期、驾驶证号、驾驶证准驾车型、有效期开始时间、地址，共9个关键字段信息。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeDriverLicense	要执行的操作。取值：RecognizeDriverLicense。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/demo-center/ocr/RecognizeDriverLicense.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。
Side	String	是	face	上传照片为驾驶证首页或者副页。 <ul style="list-style-type: none"> • face：首页 • back：副页

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
BackResult	Struct		驾驶证副页结果。
ArchiveNumber	String	130601473955	档案编号。
FaceResult	Struct		驾驶证首页结果。
Address	String	江苏省徐州市铜山区棠张镇田河村1队129号	地址。
EndDate	String	20190201	驾驶证有效期时长或有效期截止日期。具体视驾驶证数据格式而定，格式：YYYYMMDD，例如19800101，即1980年01月01日。
Gender	String	男	性别。
IssueDate	String	20130208	初次发证日期。格式：YYYYMMDD，例如19800101，即1980年01月01日。
LicenseNumber	String	320323199909099999	驾驶证号，识别失败时返回 NoResult 或 InvalidInput 。
Name	String	王昌	姓名字符串，识别失败时返回 NoResult 或 InvalidInput 。
StartDate	String	20130208	驾驶证有效期开始时间，格式：YYYYMMDD，例如19800101，即1980年01月01日。
VehicleType	String	C1	驾驶证准驾车型。
RequestId	String	374D8C7E-9ECC-4750-A87F-50571702F175	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeDriverLicense
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/demo-center/ocr/RecognizeDriverLicense.jpg
&Side=face
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>F778BC7D-63F3-4FEA-9647-14402B2460DE</RequestId>
<Data>
  <FaceResult>
    <StartDate>20130208</StartDate>
    <Address>江苏省徐州市铜山区棠张镇田河村1队129号</Address>
    <LicenseNumber>320323199909099999</LicenseNumber>
    <VehicleType>C1</VehicleType>
    <IssueDate>20130208</IssueDate>
    <Gender>男</Gender>
    <EndDate>20190201</EndDate>
    <Name>王昌</Name>
  </FaceResult>
</Data>
```

JSON 格式

```
{
  "RequestId": "F778BC7D-63F3-4FEA-9647-14402B2460DE",
  "Data": {
    "FaceResult": {
      "StartDate": "20130208",
      "Address": "江苏省徐州市铜山区棠张镇田河村1队129号",
      "LicenseNumber": "320323199909099999",
      "VehicleType": "C1",
      "IssueDate": "20130208",
      "Gender": "男",
      "EndDate": "20190201",
      "Name": "王昌"
    }
  }
}
```

错误码

关于驾驶证识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

5.2. 行驶证识别

本文档介绍行驶证识别RecognizeDrivingLicense的语法及示例。

功能描述

RecognizeDrivingLicense可以识别行驶证首页和副页关键字段内容，输出品牌型号、车辆类型、车牌号码、检验记录、核定载质量、核定载人数等21个关键字段信息。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeDrivingLicense	要执行的操作。取值：RecognizeDrivingLicense。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/4%E8%A1%8C%E9%A9%B6%E8%AF%81.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

名称	类型	是否必选	示例值	描述
Side	String	是	face	上传照片为行驶证首页或副页。 <ul style="list-style-type: none"> • face: 首页 • back: 副页

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
BackResult	Struct		行驶证副页结果。
ApprovedLoad	String	300	核定载质量。
ApprovedPassengerCapacity	String	5	核定载人数。
EnergyType	String	-	能源标志。
FileNumber	String	云AD8V02	档案编号。
GrossMass	String	2205	总质量。
InspectionRecord	String	检验有效期至2014年09月云A(01)	检验记录。
OverallDimension	String	4945x1845x1480	外廓尺寸。
PlateNumber	String	云AD8V02	号牌号码。
TractionMass	String	100	准牵引总质量。
UnladenMass	String	2000	整备质量。
FaceResult	Struct		行驶证首页结果。
Address	String	中牟县三刘寨村	地址。

名称	类型	示例值	描述
EngineNumber	String	111111	发动机号码。
IssueDate	String	20180313	发证日期，格式：YYYYMMDD，例如19800101，即1980年01月01日。
Model	String	大众汽车牌 SVW6666DFD	品牌型号。
Owner	String	郑昆中	所有人名称。
PlateNumber	String	豫A99999	车牌号码。
RegisterDate	String	20180312	注册日期，格式：YYYYMMDD，例如19800101，即1980年01月01日。
UseCharacter	String	非营运	使用性质。
VehicleType	String	小型普通客车	车辆类型。
Vin	String	SSVUDDTT2J202255 5	车辆识别代号。
RequestId	String	1DD989C1-4E08- 4E04-9D5D- 314901E91226	请求ID。

示例

请求示例

```
http(s)://[Endpoint]/?Action=RecognizeDrivingLicense
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/4%E8%A1%8C%E9%A
9%B6%E8%AF%81.jpg
&Side=face
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>1DD989C1-4E08-4E04-9D5D-314901E91226</RequestId>
<Data>
  <FaceResult>
    <PlateNumber>豫A99999</PlateNumber>
    <Owner>郑昆中</Owner>
    <EngineNumber>111111</EngineNumber>
    <Address>中牟县三刘寨村</Address>
    <VehicleType>小型普通客车</VehicleType>
    <Model>大众汽车牌SVW6666DFD</Model>
    <RegisterDate>20180312</RegisterDate>
    <IssueDate>20180313</IssueDate>
    <Vin>SSVUDDTT2J2022555</Vin>
    <UseCharacter>非营运</UseCharacter>
  </FaceResult>
</Data>
```

JSON 格式

```
{
  "RequestId": "1DD989C1-4E08-4E04-9D5D-314901E91226",
  "Data": {
    "FaceResult": {
      "PlateNumber": "豫A99999",
      "Owner": "郑昆中",
      "EngineNumber": "111111",
      "Address": "中牟县三刘寨村",
      "VehicleType": "小型普通客车",
      "Model": "大众汽车牌SVW6666DFD",
      "RegisterDate": "20180312",
      "IssueDate": "20180313",
      "Vin": "SSVUDDTT2J2022555",
      "UseCharacter": "非营运"
    }
  }
}
```

错误码

关于行驶证识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

5.3. 车牌识别

本文档介绍车牌识别RecognizeLicensePlate的语法及示例。

功能描述

RecognizeLicensePlate可以准确识别出图像中车牌位置，输出车牌位置坐标、车牌类型、车牌号码、车牌号码置信度、车牌置信度，共5个关键字段信息。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为10秒。
- URL地址中不能包含中文字符。

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeLicensePlate	要执行的操作。取值：RecognizeLicensePlate。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/3%E8%BD%A6%E7%89%8C%E8%AF%86%E5%88%AB.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Plates	Array of Plate		车牌面板详细信息。
Confidence	Float	0.99745339155197144	车牌号码的置信度，取值范围0~1。
PlateNumber	String	粤BP57E7	车牌号码。
PlateType	String	小型汽车	车牌类型。包括：小型汽车、新能源车、大型汽车、挂车、教练车、警车、军车、使领馆车、港澳车。
PlateTypeConfidence	Float	1	车牌类型的置信度，取值范围0~1。
Roi	Struct		车牌位置。
H	Integer	53	车牌号高度。
W	Integer	141	车牌号宽度。
X	Integer	294	车牌号左上角x坐标。
Y	Integer	256	车牌号左上角y坐标。
RequestId	String	3F10DAC3-CF4A-487C-BF33-3B8EB9AA12F2	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeLicensePlate
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/3%E8%BD%A6%E7%89%8C%E8%AF%86%E5%88%AB.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>3F10DAC3-CF4A-487C-BF33-3B8EB9AA12F2</RequestId>
<Data>
  <Plates>
    <PlateNumber>粤BP57E7</PlateNumber>
    <Confidence>0.99745339155197144</Confidence>
    <PlateType>小型汽车</PlateType>
    <PlateTypeConfidence>1</PlateTypeConfidence>
    <Roi>
      <W>141</W>
      <H>53</H>
      <X>294</X>
      <Y>256</Y>
    </Roi>
  </Plates>
</Data>
```

JSON 格式

```
{
  "RequestId": "3F10DAC3-CF4A-487C-BF33-3B8EB9AA12F2",
  "Data": {
    "Plates": [
      {
        "PlateNumber": "粤BP57E7",
        "Confidence": "0.99745339155197144",
        "PlateType": "小型汽车",
        "PlateTypeConfidence": 1,
        "Roi": {
          "W": 141,
          "H": 53,
          "X": 294,
          "Y": 256
        }
      }
    ]
  }
}
```

错误码

关于车牌识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

5.4. VIN码识别

本文档介绍VIN码识别RecognizeVINCode的语法及示例。

功能描述

RecognizeVINCode可以识别车辆VIN码，输出车辆VIN码数值。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeVINCode	要执行的操作。取值：RecognizeVINCode。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/2VIN%E7%A0%81%E8%AF%86%E5%88%AB.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
VinCode	String	LGXC14AA4B1106539	识别的车辆VIN码。
RequestId	String	911FC8CF-CC27-477E-BE3B-7ED77DF4DFE0	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeVINCode
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/2VIN%E7%A0%81%E8
%AF%86%E5%88%AB.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>911FC8CF-CC27-477E-BE3B-7ED77DF4DFE0</RequestId>
<Data>
  <VinCode>LGXC14AA4B1106539</VinCode>
</Data>
```

JSON 格式

```
{
  "RequestId": "911FC8CF-CC27-477E-BE3B-7ED77DF4DFE0",
  "Data": {
    "VinCode": "LGXC14AA4B1106539"
  }
}
```

错误码

关于VIN码识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

6. 行业票证类识别

6.1. 出租车发票识别

本文档介绍出租车发票识别RecognizeTaxiInvoice的语法及示例。

功能描述

RecognizeTaxiInvoice可以准确识别出全国各大城市出租车发票在图像中的位置，支持出租车发票结构化识别，输出发票号码、代码、车号、日期、时间、金额，共6个关键字段信息。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

图片限制

- 请求格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeTaxiInvoice	要执行的操作。取值：RecognizeTaxiInvoice。
ImageURL	String	是	https://viapi-demo.oss-cn-shanghai.aliyuncs.com/viapi-demo/images/RecognizeTaxiInvoice/TaxiInvoice.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的数据内容。

名称	类型	示例值	描述
Invoices	Array of Invoice		发票列表详细信息。
InvoiceRoi	Struct		发票位置。
H	Float	3625	发票高度。
W	Float	1773	发票宽度。
X	Float	513	发票左上角x坐标。
Y	Float	302	发票左上角y坐标。
Items	Array of Item		每张发票的文字列表。
ItemRoi	Struct		发票每个字段对应位置信息。
Angle	Float	-90	角度信息。坐标的原点在左上角，与x轴平行的方向角度为0。逆时针旋转角度为负，顺时针旋转角度为正。角度是水平轴（x轴）顺时针旋转，与碰到的第一个边的夹角度数。角度范围-180°~180°。
Center	Struct		位置框中心点。
X	Float	1593	位置框中心点x坐标。
Y	Float	1360	位置框中心点y坐标。
Size	Struct		位置框长宽。
H	Float	81.999984741210938	位置框高度。
W	Float	887.9998779296875	位置框宽度。
Text	String	86655664	发票每个字段信息。

名称	类型	示例值	描述
RotateType	Integer	0	发票旋转角度。 <ul style="list-style-type: none"> 0: 不需要旋转。 1: 顺时针旋转90。 2: 顺时针旋转180。 3: 顺时针旋转270。
RequestId	String	B2BBBD26-1D3E-4CFA-A80B-6A9266B8D125	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeTaxiInvoice
&ImageURL=https://viapi-demo.oss-cn-shanghai.aliyuncs.com/viapi-demo/images/RecognizeTaxiInvoice/TaxiInvoice.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>B2BBBD26-1D3E-4CFA-A80B-6A9266B8D125</RequestId>
<Data>
  <Invoices>
    <RotateType>0</RotateType>
    <InvoiceRoi>
      <W>1773</W>
      <H>3625</H>
      <X>513</X>
      <Y>302</Y>
    </InvoiceRoi>
    <Items>
      <ItemRoi>
        <Center>
          <X>1436</X>
          <Y>968.5</Y>
        </Center>
        <Angle>-90</Angle>
        <Size>
          <W>89</W>
```



```
<H>1110</H>
</Size>
</ItemRoi>
<Text>132011781210</Text>
</Items>
<Items>
<ItemRoi>
<Center>
<X>1246.5</X>
<Y>1239</Y>
</Center>
<Angle>-90</Angle>
<Size>
<W>92</W>
<H>743</H>
</Size>
</ItemRoi>
<Text>13742573</Text>
</Items>
<Items>
<ItemRoi>
<Center>
<X>1593</X>
<Y>1360</Y>
</Center>
<Angle>0</Angle>
<Size>
<W>887.9998779296875</W>
<H>81.99998474121094</H>
</Size>
</ItemRoi>
<Text>013560325483</Text>
</Items>
<Items>
<ItemRoi>
<Center>
<X>1744</X>
<Y>1507.5</Y>
</Center>
<Angle>0</Angle>
```

```
<Size>
  <W>585.9998779296875</W>
  <H>80.99998474121094</H>
</Size>
</ItemRoi>
<Text>86655664</Text>
</Items>
<Items>
  <ItemRoi>
    <Center>
      <X>1778</X>
      <Y>1668</Y>
    </Center>
    <Angle>-90</Angle>
    <Size>
      <W>80</W>
      <H>524</H>
    </Size>
  </ItemRoi>
  <Text>A-B6167</Text>
</Items>
<Items>
  <ItemRoi>
    <Center>
      <X>1629.5</X>
      <Y>1840</Y>
    </Center>
    <Angle>-90</Angle>
    <Size>
      <W>81.99998474121094</W>
      <H>828.9998779296875</H>
    </Size>
  </ItemRoi>
  <Text>2018年02月28日</Text>
</Items>
<Items>
  <ItemRoi>
    <Center>
      <X>1864</X>
      <Y>2009.5</Y>
    </Center>
```

```
<Angle>-90</Angle>
<Size>
  <W>91</W>
  <H>390</H>
</Size>
</ItemRoi>
<Text>22:24</Text>
</Items>
<Items>
  <ItemRoi>
    <Center>
      <X>1862</X>
      <Y>2180</Y>
    </Center>
    <Angle>-90</Angle>
    <Size>
      <W>96</W>
      <H>384</H>
    </Size>
  </ItemRoi>
  <Text>22:57</Text>
</Items>
<Items>
  <ItemRoi>
    <Center>
      <X>1730</X>
      <Y>2348</Y>
    </Center>
    <Angle>-90</Angle>
    <Size>
      <W>96</W>
      <H>608</H>
    </Size>
  </ItemRoi>
  <Text>3.60元/公里</Text>
</Items>
<Items>
  <ItemRoi>
    <Center>
      <X>1813</X>
      <Y>2530</Y>
```

```
</Center>
<Angle>-90</Angle>
<Size>
  <W>92</W>
  <H>466</H>
</Size>
</ItemRoi>
<Text>40.9公里</Text>
</Items>
<Items>
  <ItemRoi>
    <Center>
      <X>1741</X>
      <Y>2697</Y>
    </Center>
    <Angle>0</Angle>
    <Size>
      <W>613.9998779296875</W>
      <H>91.99998474121094</H>
    </Size>
  </ItemRoi>
  <Text>00:04.03</Text>
</Items>
<Items>
  <ItemRoi>
    <Center>
      <X>1785.5</X>
      <Y>2868.5</Y>
    </Center>
    <Angle>-90</Angle>
    <Size>
      <W>91</W>
      <H>519</H>
    </Size>
  </ItemRoi>
  <Text>129.00元</Text>
</Items>
<Items>
  <ItemRoi>
    <Center>
```

```
<X>1850.5</X>
<Y>3026</Y>
</Center>
<Angle>0</Angle>
<Size>
  <W>400.99993896484375</W>
  <H>105.99998474121094</H>
</Size>
</ItemRoi>
<Text>0.00元</Text>
</Items>
<Items>
  <ItemRoi>
    <Center>
      <X>1848.5</X>
      <Y>3187</Y>
    </Center>
    <Angle>0</Angle>
    <Size>
      <W>396.99993896484375</W>
      <H>99.99998474121094</H>
    </Size>
  </ItemRoi>
  <Text>0.00元</Text>
</Items>
<Items>
  <ItemRoi>
    <Center>
      <X>1953</X>
      <Y>3368</Y>
    </Center>
    <Angle>0</Angle>
    <Size>
      <W>166</W>
      <H>78</H>
    </Size>
  </ItemRoi>
  <Text>*</Text>
</Items>
<Items>
  <ItemRoi>
```

```
<Center>
  <X>1954</X>
  <Y>3539</Y>
</Center>
<Angle>0</Angle>
<Size>
  <W>160</W>
  <H>72</H>
</Size>
</ItemRoi>
<Text>**</Text>
</Items>
<Items>
  <ItemRoi>
    <Center>
      <X>1945</X>
      <Y>3704.5</Y>
    </Center>
    <Angle>0</Angle>
    <Size>
      <W>158</W>
      <H>71</H>
    </Size>
  </ItemRoi>
  <Text>**</Text>
</Items>
</Invoices>
</Data>
```

JSON 格式

```
{
  "RequestId": "B2BBBD26-1D3E-4CFA-A80B-6A9266B8D125",
  "Data": {
    "Invoices": [
      {
        "RotateType": 0,
        "InvoiceRoi": {
          "W": 1773,
          "H": 3625,
          "X": 513,
```

```
"Y": 302
},
"Items": [
  {
    "ItemRoi": {
      "Center": {
        "X": 1436,
        "Y": 968.5
      },
      "Angle": -90,
      "Size": {
        "W": 89,
        "H": 1110
      }
    },
    "Text": "132011781210"
  },
  {
    "ItemRoi": {
      "Center": {
        "X": 1246.5,
        "Y": 1239
      },
      "Angle": -90,
      "Size": {
        "W": 92,
        "H": 743
      }
    },
    "Text": "13742573"
  },
  {
    "ItemRoi": {
      "Center": {
        "X": 1593,
        "Y": 1360
      },
      "Angle": 0,
      "Size": {
        "W": 887.9998779296875,
```

```
"H": 81.999984741210938
}
},
"Text": "013560325483"
},
{
  "ItemRoi": {
    "Center": {
      "X": 1744,
      "Y": 1507.5
    },
    "Angle": 0,
    "Size": {
      "W": 585.9998779296875,
      "H": 80.999984741210938
    }
  },
  "Text": "86655664"
},
{
  "ItemRoi": {
    "Center": {
      "X": 1778,
      "Y": 1668
    },
    "Angle": -90,
    "Size": {
      "W": 80,
      "H": 524
    }
  },
  "Text": "A-B6167"
},
{
  "ItemRoi": {
    "Center": {
      "X": 1629.5,
      "Y": 1840
    },
    "Angle": -90,
    "Size": {
```



```
    "W": 81.999984741210938,  
    "H": 828.9998779296875  
  }  
},  
"Text": "2018年02月28日"  
},  
{  
  "ItemRoi": {  
    "Center": {  
      "X": 1864,  
      "Y": 2009.5  
    },  
    "Angle": -90,  
    "Size": {  
      "W": 91,  
      "H": 390  
    }  
  },  
  "Text": "22:24"  
},  
{  
  "ItemRoi": {  
    "Center": {  
      "X": 1862,  
      "Y": 2180  
    },  
    "Angle": -90,  
    "Size": {  
      "W": 96,  
      "H": 384  
    }  
  },  
  "Text": "22:57"  
},  
{  
  "ItemRoi": {  
    "Center": {  
      "X": 1730,  
      "Y": 2348  
    },  
    "Angle": -90.
```

```
    "Size": {
      "W": 96,
      "H": 608
    }
  },
  "Text": "3.60元/公里"
},
{
  "ItemRoi": {
    "Center": {
      "X": 1813,
      "Y": 2530
    },
    "Angle": -90,
    "Size": {
      "W": 92,
      "H": 466
    }
  },
  "Text": "40.9公里"
},
{
  "ItemRoi": {
    "Center": {
      "X": 1741,
      "Y": 2697
    },
    "Angle": 0,
    "Size": {
      "W": 613.9998779296875,
      "H": 91.999984741210938
    }
  },
  "Text": "00:04.03"
},
{
  "ItemRoi": {
    "Center": {
      "X": 1785.5,
      "Y": 2868.5
```

```
},
  "Angle": -90,
  "Size": {
    "W": 91,
    "H": 519
  }
},
  "Text": "129.00元"
},
{
  "ItemRoi": {
    "Center": {
      "X": 1850.5,
      "Y": 3026
    },
    "Angle": 0,
    "Size": {
      "W": 400.99993896484375,
      "H": 105.99998474121094
    }
  },
  "Text": "0.00元"
},
{
  "ItemRoi": {
    "Center": {
      "X": 1848.5,
      "Y": 3187
    },
    "Angle": 0,
    "Size": {
      "W": 396.99993896484375,
      "H": 99.999984741210938
    }
  },
  "Text": "0.00元"
},
{
  "ItemRoi": {
    "Center": {
      "X": 1953,
```

```
    "Y": 3368
  },
  "Angle": 0,
  "Size": {
    "W": 166,
    "H": 78
  }
},
"Text": ""
},
{
  "ItemRoi": {
    "Center": {
      "X": 1954,
      "Y": 3539
    },
    "Angle": 0,
    "Size": {
      "W": 160,
      "H": 72
    }
  },
  "Text": ""
},
{
  "ItemRoi": {
    "Center": {
      "X": 1945,
      "Y": 3704.5
    },
    "Angle": 0,
    "Size": {
      "W": 158,
      "H": 71
    }
  },
  "Text": ""
}
]
}
```

```
},  
}  
}
```

错误码

关于出租车发票识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

6.2. 增值税发票识别

本文档为您介绍增值税发票识别RecognizeVATInvoice的语法及示例。

功能描述

RecognizeVATInvoice可以识别增值税发票（电子发票和纸质发票）关键字段内容，包括：校验码、复核人、开票人、发票代码、收款人等。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：PDF、JPG。
- 图片大小：大小不超过10M。
- 图片分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeVATInvoice	要执行的操作。取值：RecognizeVATInvoice。
FileType	String	是	jpg	输入的图片格式。

名称	类型	是否必选	示例值	描述
FileURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test/ant_ai/vat_invoice/2019041500152001020003743286_0.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Box	Struct		发票关键字位置框。
Checkers	List	589,1003,662,1033	复核人位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
Clerks	List	986,1003,1060,1033	开票人位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
InvoiceAmounts	List	1364, 776,1438,804	价税合计位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
InvoiceCodes	List	1378,41,1520,68	发票代码位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
InvoiceDates	List	1376,115,1596,145	开票日期位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
InvoiceFakeCodes	List	1376,153,1640,181	校验码位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。

名称	类型	示例值	描述
InvoiceNoes	List	1377,78,1478,105	发票号码位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
PayeeAddresses	List	355,909,734,939	购买方地址及电话位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
PayeeBankNames	List	354,947,938,977	购买方开户行及账号位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
PayeeNames	List	356,833,633,865	购买方名称位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
PayeeRegisterNoes	List	356,873,571,902	销售方纳税人识别号位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
Payees	List	189,1003,264,1033	收款人位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
PayerAddresses	List	0,0,0,0	销售方地址及电话位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
PayerBankNames	List	0,0,0,0	销售方开户行及账号位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
PayerNames	List	354,222,700,255	购买方名称位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
PayerRegisterNoes	List	358,262,567,290	购买方纳税人识别号位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。

名称	类型	示例值	描述
SumAmounts	List	32,774,629,805	价税合计位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
TaxAmounts	List	1606,721,1658,748	合计税额位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
WithoutTaxAmounts	List	1265,721,1339,749	合计金额位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
Content	Struct		合计金额识别发票各项结果。
AntiFakeCode	String	02702870934284730434	校验码。
Checker	String	赵弯弯	复核人。
Clerk	String	赵弯弯	开票人。
InvoiceAmount	String	200.00	价税合计（小写）。
InvoiceCode	String	031001600311	发票代码。
InvoiceDate	String	20190415	开票日期。
InvoiceNo	String	80625433	发票号码。
Payee	String	赵弯弯	收款人。
PayeeAddress	String	上海虹桥机场迎宾二路161号22342185	销售方地址、电话。
PayeeBankName	String	上海浦东发展银行空港支行076389-98910158000000030-22	销售方开户行及账号。
PayeeName	String	上海机场（集团）有限公司	销售方名称。

名称	类型	示例值	描述
PayeeRegisterNo	String	91310000132284295X	销售方纳税人识别号。
PayerAddress	String	xxx	购买方地址、电话。
PayerBankName	String	xxx	购买方开户行及账号。
PayerName	String	百特医疗用品贸易(上海)有限公司	购买方名称。
PayerRegisterNo	String	91310000607402073L	购买方纳税人识别号。
SumAmount	String	200.00	价税合计(大写)。
TaxAmount	String	9.52	合计税额。
WithoutTaxAmount	String	190.48	合计金额。
RequestId	String	20926BD9-7B2A-48D3-8B05-99F380791271	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeVATInvoice
&FileType=jpg
&FileURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test/ant_ai/vat_invoice/20190415001520010
20003743286_0.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>20926BD9-7B2A-48D3-8B05-99F380791271</RequestId>
<Data>
  <Content>
    <Checker>赵弯弯</Checker>
    <PayeeAddress>上海虹桥机场迎宾二路161号22342185</PayeeAddress>
    <PayeeName>上海机场(集团)有限公司</PayeeName>
    <PayerAddress>
```

```
<PayerAddress/>
<AntiFakeCode>02702870934284730434</AntiFakeCode>
<PayerBankName/>
<InvoiceNo>80625433</InvoiceNo>
<PayerRegisterNo>91310000607402073L</PayerRegisterNo>
<PayerName>百特医疗用品贸易（上海）有限公司</PayerName>
<WithoutTaxAmount>190.48</WithoutTaxAmount>
<InvoiceAmount>200.00</InvoiceAmount>
<TaxAmount>9.52</TaxAmount>
<PayeeBankName>上海浦东发展银行空港支行076389-9891015800000030-22</PayeeBankName>
<Clerk>赵弯弯</Clerk>
<InvoiceCode>031001600311</InvoiceCode>
<PayeeRegisterNo>91310000132284295X</PayeeRegisterNo>
<InvoiceDate>20190415</InvoiceDate>
<SumAmount>200.00</SumAmount>
<Payee>赵弯弯</Payee>
</Content>
<Box>
  <PayerRegisterNoes>358</PayerRegisterNoes>
  <PayerRegisterNoes>262</PayerRegisterNoes>
  <PayerRegisterNoes>567</PayerRegisterNoes>
  <PayerRegisterNoes>290</PayerRegisterNoes>
  <PayerAddresses>0</PayerAddresses>
  <PayerAddresses>0</PayerAddresses>
  <PayerAddresses>0</PayerAddresses>
  <PayerAddresses>0</PayerAddresses>
  <InvoiceAmounts>1364</InvoiceAmounts>
  <InvoiceAmounts>776</InvoiceAmounts>
  <InvoiceAmounts>1438</InvoiceAmounts>
  <InvoiceAmounts>804</InvoiceAmounts>
  <InvoiceDates>1376</InvoiceDates>
  <InvoiceDates>115</InvoiceDates>
  <InvoiceDates>1596</InvoiceDates>
  <InvoiceDates>145</InvoiceDates>
  <PayeeAddresses>355</PayeeAddresses>
  <PayeeAddresses>909</PayeeAddresses>
  <PayeeAddresses>734</PayeeAddresses>
  <PayeeAddresses>939</PayeeAddresses>
  <Clerks>986</Clerks>
  <Clerks>1003</Clerks>
  <Clerks>1060</Clerks>
```

<Clerks>1033</Clerks>
<WithoutTaxAmounts>1265</WithoutTaxAmounts>
<WithoutTaxAmounts>721</WithoutTaxAmounts>
<WithoutTaxAmounts>1339</WithoutTaxAmounts>
<WithoutTaxAmounts>749</WithoutTaxAmounts>
<Checkers>589</Checkers>
<Checkers>1003</Checkers>
<Checkers>662</Checkers>
<Checkers>1033</Checkers>
<InvoiceCodes>1378</InvoiceCodes>
<InvoiceCodes>41</InvoiceCodes>
<InvoiceCodes>1520</InvoiceCodes>
<InvoiceCodes>68</InvoiceCodes>
<TaxAmounts>1606</TaxAmounts>
<TaxAmounts>721</TaxAmounts>
<TaxAmounts>1658</TaxAmounts>
<TaxAmounts>748</TaxAmounts>
<SumAmounts>532</SumAmounts>
<SumAmounts>774</SumAmounts>
<SumAmounts>629</SumAmounts>
<SumAmounts>805</SumAmounts>
<PayerNames>354</PayerNames>
<PayerNames>222</PayerNames>
<PayerNames>700</PayerNames>
<PayerNames>255</PayerNames>
<InvoiceFakeCodes>1376</InvoiceFakeCodes>
<InvoiceFakeCodes>153</InvoiceFakeCodes>
<InvoiceFakeCodes>1640</InvoiceFakeCodes>
<InvoiceFakeCodes>181</InvoiceFakeCodes>
<PayeeNames>356</PayeeNames>
<PayeeNames>833</PayeeNames>
<PayeeNames>633</PayeeNames>
<PayeeNames>865</PayeeNames>
<PayeeBankNames>354</PayeeBankNames>
<PayeeBankNames>947</PayeeBankNames>
<PayeeBankNames>938</PayeeBankNames>
<PayeeBankNames>977</PayeeBankNames>
<Payees>189</Payees>
<Payees>1003</Payees>
<Payees>264</Payees>
<Payees>1033</Payees>

```

<InvoiceNoes>1377</InvoiceNoes>
<InvoiceNoes>78</InvoiceNoes>
<InvoiceNoes>1478</InvoiceNoes>
<InvoiceNoes>105</InvoiceNoes>
<PayerBankNames>0</PayerBankNames>
<PayerBankNames>0</PayerBankNames>
<PayerBankNames>0</PayerBankNames>
<PayerBankNames>0</PayerBankNames>
<PayeeRegisterNoes>356</PayeeRegisterNoes>
<PayeeRegisterNoes>873</PayeeRegisterNoes>
<PayeeRegisterNoes>571</PayeeRegisterNoes>
<PayeeRegisterNoes>902</PayeeRegisterNoes>
</Box>
</Data>

```

JSON 格式

```

{
  "RequestId": "20926BD9-7B2A-48D3-8B05-99F380791271",
  "Data": {
    "Content": {
      "Checker": "赵弯弯",
      "PayeeAddress": "上海虹桥机场迎宾二路161号22342185",
      "PayeeName": "上海机场（集团）有限公司",
      "PayerAddress": "",
      "AntiFakeCode": "02702870934284730434",
      "PayerBankName": "",
      "InvoiceNo": "80625433",
      "PayerRegisterNo": "91310000607402073L",
      "PayerName": "百特医疗用品贸易（上海）有限公司",
      "WithoutTaxAmount": "190.48",
      "InvoiceAmount": "200.00",
      "TaxAmount": "9.52",
      "PayeeBankName": "上海浦东发展银行空港支行076389-98910158000000030-22",
      "Clerk": "赵弯弯",
      "InvoiceCode": "031001600311",
      "PayeeRegisterNo": "91310000132284295X",
      "InvoiceDate": "20190415",
      "SumAmount": "200.00",
      "Payee": "赵弯弯"
    }
  }
}

```

```
”  
"Box": {  
  "PayerRegisterNoes": [  
    358,  
    262,  
    567,  
    290  
  ],  
  "PayerAddresses": [  
    0,  
    0,  
    0,  
    0  
  ],  
  "InvoiceAmounts": [  
    1364,  
    776,  
    1438,  
    804  
  ],  
  "InvoiceDates": [  
    1376,  
    115,  
    1596,  
    145  
  ],  
  "PayeeAddresses": [  
    355,  
    909,  
    734,  
    939  
  ],  
  "Clerks": [  
    986,  
    1003,  
    1060,  
    1033  
  ],  
  "WithoutTaxAmounts": [  
    1265,  
    721,  
  ]  
}
```

```
1339,  
749  
],  
"Checkers": [  
589,  
1003,  
662,  
1033  
],  
"InvoiceCodes": [  
1378,  
41,  
1520,  
68  
],  
"TaxAmounts": [  
1606,  
721,  
1658,  
748  
],  
"SumAmounts": [  
532,  
774,  
629,  
805  
],  
"PayerNames": [  
354,  
222,  
700,  
255  
],  
"InvoiceFakeCodes": [  
1376,  
153,  
1640,  
181  
],  
"PayeeNames": [  
356,
```

```
833,  
633,  
865  
],  
"PayeeBankNames": [  
354,  
947,  
938,  
977  
],  
"Payees": [  
189,  
1003,  
264,  
1033  
],  
"InvoiceNoes": [  
1377,  
78,  
1478,  
105  
],  
"PayerBankNames": [  
0,  
0,  
0,  
0  
],  
"PayeeRegisterNoes": [  
356,  
873,  
571,  
902  
]  
}  
}  
}
```

错误码

关于增值税发票识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

6.3. 火车票识别

本文档介绍火车票识别RecognizeTrainTicket的语法及示例。

功能描述

RecognizeTrainTicket可以对火车票进行结构化识别，输出乘车时间、始发站、目的站、座位席别、乘车人姓名、车次号、票价、车厢及座次号，共8个关键字段信息。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeTrainTicket	要执行的操作。取值：RecognizeTrainTicket。
ImageURL	String	是	https://viapi-demo.oss-cn-shanghai.aliyuncs.com/viapi-demo/images/DetectImageElements/xxx.png	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Date	String	2016年10月01日 08:16	乘车日期时间。
DepartureStation	String	濮阳站	始发站点。
Destination	String	郑州站	目的站点。
Level	String	新空调硬座	座位席别。
Name	String	张三	乘车人姓名。
Number	String	K5372	车次号。
Price	Float	40.5	票价。
Seat	String	16车081号	座位车厢及座次号。
RequestId	String	BE4B73EA-30A0- 4573-A548- 3A101B34641A	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeTrainTicket
&ImageURL=https://viapi-demo.oss-cn-shanghai.aliyuncs.com/viapi-demo/images/DetectImageElements/xxx.png
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>BE4B73EA-30A0-4573-A548-3A101B34641A</RequestId>
<Data>
  <Destination>郑州站</Destination>
  <Seat>16车081号</Seat>
  <Number>K5372</Number>
  <Price>40.5</Price>
  <Level>新空调硬座</Level>
  <Date>2016年10月01日08:16</Date>
  <Name>张三</Name>
  <DepartureStation>濮阳站</DepartureStation>
</Data>
```

JSON 格式

```
{
  "RequestId": "BE4B73EA-30A0-4573-A548-3A101B34641A",
  "Data": {
    "Destination": "郑州站",
    "Seat": "16车081号",
    "Number": "K5372",
    "Price": 40.5,
    "Level": "新空调硬座",
    "Date": "2016年10月01日08:16",
    "Name": "张三",
    "DepartureStation": "濮阳站"
  }
}
```

错误码

关于火车票识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会系统自动清理删除。

6.4. 外卖单识别

本文介绍外卖单识别RecognizeTakeoutOrder的语法及示例。

功能描述

RecognizeTakeoutOrder可以识别外卖单上的关键字段内容，输出商店名称、电话、包装费、配送费、商品合计、其他费用合计、顾客优惠合计、总件数、在线支付、订单编号、下单时间等。目前支持饿了么外卖单。

前提条件

请确保您已开通[文字识别服务](#)，若未开通服务请[立即开通](#)。

前提条件

- 图片格式：JPEG、JPG、PNG。
- 图片大小：大小不超过10M。
- 图片分辨率：不限制图片分辨率，但是分辨率太高可能会导致API推理超时，推理超时时间为5秒。
- URL地址中不能包含中文字符。

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RecognizeTakeoutOrder	要执行的操作。取值：RecognizeTakeoutOrder。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/takeoutorder.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Elements	Array of Element		识别出的外卖单具体内容。
Boxes	List	[2177, 2082, 2321, 2152]	字段位置框信息，格式为【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。

名称	类型	示例值	描述
Name	String	delivery_fee	外卖单上关键字段。包括： <ul style="list-style-type: none"> • shop_name: 商店名称 • telephone_number: 电话 • packaging_fee: 包装费 • delivery_fee: 配送费 • total_goods: 商品合计 • total_other: 其他费用合计 • total_benefits: 顾客优惠合计 • total_number: 总件数 • online_pay: 在线支付 • order_number: 订单编号 • order_time: 下单时间
Score	Float	0.68	识别出的字段信息的置信度。
Value	String	4.00	具体字段对应的内容。
RequestId	String	3068BB25-663A-4BEC-9481-B970FFA5256C	请求ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeTakeoutOrder
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/takeoutorder.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>3068BB25-663A-4BEC-9481-B970FFA5256C</RequestId>
<Data>
  <Elements>
    <Score>0.9974356293678284</Score>
    <Value>4.00</Value>
    <Name>delivery_fee</Name>
    <Boxes>2190</Boxes>
    <Boxes>2022</Boxes>
    <Boxes>2337</Boxes>
```

```
<Boxes>2098</Boxes>
</Elements>
<Elements>
  <Score>0.9995468854904175</Score>
  <Value>36.00</Value>
  <Name>online_pay</Name>
  <Boxes>2157</Boxes>
  <Boxes>2531</Boxes>
  <Boxes>2340</Boxes>
  <Boxes>2616</Boxes>
</Elements>
<Elements>
  <Score>0.9989166855812073</Score>
  <Value>3056887498797982898</Value>
  <Name>order_number</Name>
  <Boxes>1597</Boxes>
  <Boxes>3653</Boxes>
  <Boxes>2310</Boxes>
  <Boxes>3755</Boxes>
</Elements>
<Elements>
  <Score>0.9980564713478088</Score>
  <Value>12-1720:25</Value>
  <Name>order_time</Name>
  <Boxes>1598</Boxes>
  <Boxes>3762</Boxes>
  <Boxes>2015</Boxes>
  <Boxes>3853</Boxes>
</Elements>
<Elements>
  <Score>0.9959266781806946</Score>
  <Value>0.00</Value>
  <Name>packaging_fee</Name>
  <Boxes>2187</Boxes>
  <Boxes>1934</Boxes>
  <Boxes>2336</Boxes>
  <Boxes>2010</Boxes>
</Elements>
<Elements>
  <Score>0.9957205653190613</Score>
  <Value>越峰便利店(鲁迅中路店)</Value>
```

```
<Name>shop_name</Name>
<Boxes>1260</Boxes>
<Boxes>847</Boxes>
<Boxes>1996</Boxes>
<Boxes>924</Boxes>
</Elements>
<Elements>
  <Score>0.9995644092559814</Score>
  <Value>13588503737</Value>
  <Name>telephone_number</Name>
  <Boxes>1258</Boxes>
  <Boxes>925</Boxes>
  <Boxes>1625</Boxes>
  <Boxes>991</Boxes>
</Elements>
<Elements>
  <Score>0.9992033243179321</Score>
  <Value>-2.00</Value>
  <Name>total_benefits</Name>
  <Boxes>2159</Boxes>
  <Boxes>2366</Boxes>
  <Boxes>2336</Boxes>
  <Boxes>2439</Boxes>
</Elements>
<Elements>
  <Score>0.9994907975196838</Score>
  <Value>34.00</Value>
  <Name>total_goods</Name>
  <Boxes>2157</Boxes>
  <Boxes>2189</Boxes>
  <Boxes>2339</Boxes>
  <Boxes>2270</Boxes>
</Elements>
<Elements>
  <Score>0.9997580051422119</Score>
  <Value>3</Value>
  <Name>total_number</Name>
  <Boxes>1520</Boxes>
  <Boxes>2530</Boxes>
  <Boxes>1564</Boxes>
  <Boxes>2612</Boxes>
```

```
<Boxes>2612</Boxes>
</Elements>
<Elements>
  <Score>0.9937386512756348</Score>
  <Value>4.00</Value>
  <Name>total_other</Name>
  <Boxes>2191</Boxes>
  <Boxes>2281</Boxes>
  <Boxes>2337</Boxes>
  <Boxes>2353</Boxes>
</Elements>
</Data>
```

JSON 格式

```
{
  "RequestId": "3068BB25-663A-4BEC-9481-B970FFA5256C",
  "Data": {
    "Elements": [
      {
        "Score": "0.9974356293678284",
        "Value": "4.00",
        "Name": "delivery_fee",
        "Boxes": [
          2190,
          2022,
          2337,
          2098
        ]
      },
      {
        "Score": "0.9995468854904175",
        "Value": "36.00",
        "Name": "online_pay",
        "Boxes": [
          2157,
          2531,
          2340,
          2616
        ]
      }
    ]
  }
}
```

```
{
  "Score": "0.9989166855812073",
  "Value": "3056887498797982898",
  "Name": "order_number",
  "Boxes": [
    1597,
    3653,
    2310,
    3755
  ]
},
{
  "Score": "0.9980564713478088",
  "Value": "12-1720:25",
  "Name": "order_time",
  "Boxes": [
    1598,
    3762,
    2015,
    3853
  ]
},
{
  "Score": "0.9959266781806946",
  "Value": "0.00",
  "Name": "packaging_fee",
  "Boxes": [
    2187,
    1934,
    2336,
    2010
  ]
},
{
  "Score": "0.9957205653190613",
  "Value": "越峰便利店(鲁迅中路店)",
  "Name": "shop_name",
  "Boxes": [
    1260,
    847,
    1996,
```



```
924
]
},
{
  "Score": "0.9995644092559814",
  "Value": "13588503737",
  "Name": "telephone_number",
  "Boxes": [
    1258,
    925,
    1625,
    991
  ]
},
{
  "Score": "0.9992033243179321",
  "Value": "-2.00",
  "Name": "total_benefits",
  "Boxes": [
    2159,
    2366,
    2336,
    2439
  ]
},
{
  "Score": "0.9994907975196838",
  "Value": "34.00",
  "Name": "total_goods",
  "Boxes": [
    2157,
    2189,
    2339,
    2270
  ]
},
{
  "Score": "0.9997580051422119",
  "Value": "3",
  "Name": "total_number",
  "Boxes": [
```

```
    "Boxes": [
      1520,
      2530,
      1564,
      2612
    ]
  },
  {
    "Score": "0.9937386512756348",
    "Value": "4.00",
    "Name": "total_other",
    "Boxes": [
      2191,
      2281,
      2337,
      2353
    ]
  }
]
```

错误码

关于外卖单识别的错误码，详情请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

7. 查询异步任务结果

本文档介绍查询异步任务结果GetAsyncJobResult的语法及示例。

功能描述

异步接口调用API接口后，返回的并不是真正的请求结果，您需要保存返回结果中的RequestId，然后调用GetAsyncJobResult来获取真正的请求结果。

以下均为异步接口，均需要调用GetAsyncJobResult来获取真实的请求结果。

- 文字识别
 - TrimDocument
- 视频理解
 - DetectVideoShot
 - GenerateVideoCover
- 视频增强
 - AdjustVideoColor
 - EraseVideoSubtitles
 - EraseVideoLogo
 - AbstractEcommerceVideo
 - AbstractFilmVideo
 - SuperResolveVideo
- 视频分割
 - SegmentVideoBody

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	GetAsyncJobResult	系统规定参数。取值：GetAsyncJobResult。
JobId	String	是	E75FE679-0303-4DD1-8252-1143B4FA8A27	异步接口返回的RequestId，输入后可以查询异步接口的真实请求结果。

返回数据

名称	类型	示例值	描述
----	----	-----	----

名称	类型	示例值	描述
Data	Struct		返回的数据内容。
ErrorCode	String	InvalidParameter	异步任务错误码。
ErrorMessage	String	非法入参	异步任务错误码信息。
JobId	String	49E2CC28-ED1D-4CC5-854D-7D0AE2B20976	异步任务Id。
Result	String	<pre>{ "Content": "\<div> <h2> 2017 年 3 月 40 多家陶企上榜失信被执行人名单 </h2> <div> 1 月 7 日，陶卫网记者根据最高人民法院及各地法院发布的失信被执行人信息统计，2019 年 12 月全国各地有 112 家陶瓷企业被列入“失信被执行人”名单，名单涉及 21 个省（市）。此次名单中，广东省的失信陶企多达 28 家。 </div> <div> 据统计，这些“失信被执行人”的上榜原因主要为债务纠纷，欠款不还，大部分为拖欠供应商货款、工资、员工赔偿、银行贷款以及融资租赁租金等，也有涉及运输、燃气等费用，小部分企业涉及承担债务连带清偿责任。 </div> <table border=1> <tr> <td> 公司名称 </td> <td> 最新失信发布时间 </td> <td> 累计失信次数 </td> </tr> <tr> <td> 潮州市澳士通陶瓷实业有限公司 </td> <td> 3 月 15 日 </td> <td> 1 次 </td> </tr> <tr> <td> 肇庆市德圣陶瓷有限公司 </td> <td> 3 月 14</pre>	异步任务返回的真实请求结果。

名称	类型	示例值	描述
		<pre> <table> <tr> <td> 6 次 </td> </tr> <tr> <td> 清远市港龙陶 瓷有限公司 </td> <td> 3 月 10 日 </td> <td> 28 次 </td> </tr> <tr> <td> 广西金沙江陶 瓷有限公司 </td> <td> 3 月 14 日 </td> <td> 3 次 </td> </tr> <tr> <td> 宜丰县凯扬陶 瓷发展有限公司 </td> <td> 3 月 29 日 </td> <td> 11 次 </td> </tr> <tr> <td> 江西领先精工 陶瓷发展有限公司 </td> <td> 3 月 29 日 </td> <td> 23 次 </td> </tr> <tr> <td> 高安弘建陶瓷 有限公司 </td> <td> 3 月 28 日 </td> <td> 5 次 </td> </tr> </table> </pre>	
Status	String	PROCESS_SUCCESS	<p>异步任务状态。包括：</p> <ul style="list-style-type: none"> • QUEUING：任务排队中 • PROCESSING：异步处理中 • PROCESS_SUCCESS：处理成功 • PROCESS_FAILED：处理失败 • TIMEOUT_FAILED：任务超时未处理完成 • LIMIT_RETRY_FAILED：超过最大重试次数
RequestId	String	A1F44EC4-118D-4A03-B213-F908F36F7DAA	当前请求的request id。

示例

请求示例

```

http(s)://[Endpoint]/?Action=GetAsyncJobResult
&JobId=E75FE679-0303-4DD1-8252-1143B4FA8A27
&<公共请求参数>

```

正常返回示例

XML 格式

```

<RequestId>C6F44EF2-21EE-4CD7-B0A3-56F36ED91D48</RequestId>
<Data>
  <Status>PROCESS_SUCCESS</Status>
  <JobId>E700C1C1-57E4-413A-9987-56A5A0443CE4</JobId>
  <Result>{"Content": "<div &gt; &lt;h2 &gt; 2017 年 3 月 40 多家陶企上榜失信被执行人名单 &lt;/h2&gt;&lt;div &gt; 1 月 7 日，陶卫网记者根据最高人民法院及各地法院发布的失信被执行人信息统计，2019 年 12 月全国各地有 112 家陶瓷企业被列入“失信被执行人”名单，名单涉及 21 个省（市）。此次名单中，广东省的失信陶企多达 28 家。&lt;/div&gt;&lt;div &gt; 据统计，这些“失信被执行人”的上榜原因主要为债务纠纷，欠款不还，大部分为拖欠供应商货款、工资、员工赔偿、银行贷款以及融资租赁租金等，也有涉及运输、燃气等费用，小部分企业涉及承担债务连带清偿责任。&lt;/div&gt;&lt;table border=1&gt; &lt;tr&gt;&lt;td &gt; 公司名称 &lt;/td&gt;&lt;td &gt; 最新失信发布时间 &lt;/td&gt;&lt;td &gt; 累计失信次数 &lt;/td&gt; &lt;/tr&gt; &lt;tr&gt;&lt;td &gt; 潮州市澳士通陶瓷实业有限公司 &lt;/td&gt;&lt;td &gt; 3 月 15 日 &lt;/td&gt;&lt;td &gt; 1 次 &lt;/td&gt; &lt;/tr&gt; &lt;tr&gt;&lt;td &gt; 肇庆市德圣陶瓷有限公司 &lt;/td&gt;&lt;td &gt; 3 月 14 日 &lt;/td&gt;&lt;td &gt; 6 次 &lt;/td&gt; &lt;/tr&gt; &lt;tr&gt;&lt;td &gt; 清远市港龙陶瓷有限公司 &lt;/td&gt;&lt;td &gt; 3 月 10 日 &lt;/td&gt;&lt;td &gt; 28 次 &lt;/td&gt; &lt;/tr&gt; &lt;tr&gt;&lt;td &gt; 广西金沙江陶瓷有限公司 &lt;/td&gt;&lt;td &gt; 3 月 14 日 &lt;/td&gt;&lt;td &gt; 3 次 &lt;/td&gt; &lt;/tr&gt; &lt;tr&gt;&lt;td &gt; 宜丰县凯扬陶瓷发展有限公司 &lt;/td&gt;&lt;td &gt; 3 月 29 日 &lt;/td&gt;&lt;td &gt; 11 次 &lt;/td&gt; &lt;/tr&gt; &lt;tr&gt;&lt;td &gt; 江西领先精工陶瓷发展有限公司 &lt;/td&gt;&lt;td &gt; 3 月 29 日 &lt;/td&gt;&lt;td &gt; 23 次 &lt;/td&gt; &lt;/tr&gt; &lt;tr&gt;&lt;td &gt; 高安弘建陶瓷有限公司 &lt;/td&gt;&lt;td &gt; 3 月 28 日 &lt;/td&gt;&lt;td &gt; 5 次 &lt;/td&gt; &lt;/tr&gt;&lt;/table&gt;"}</Result>
</Data>

```

JSON 格式

```
{
  "RequestId": "C6F44EF2-21EE-4CD7-B0A3-56F36ED91D48",
  "Data": {
    "Status": "PROCESS_SUCCESS",
    "JobId": "E700C1C1-57E4-413A-9987-56A5A0443CE4",
    "Result": "{\\\"Content\\\":\\\"<div><h2>2017年3月40多家陶企上榜失信被执行人名单</h2><div>1月7日，陶卫网记者根据最高人民法院及各地法院发布的失信被执行人信息统计，2019年12月全国各地有112家陶瓷企业被列入“失信被执行人”名单，名单涉及21个省（市）。此次名单中，广东省的失信陶企多达28家。</div><div>据统计，这些“失信被执行人”的上榜原因主要为债务纠纷，欠款不还，大部分为拖欠供应商货款、工资、员工赔偿、银行贷款以及融资租赁租金等，也有涉及运输、燃气等费用，小部分企业涉及承担债务连带清偿责任。</div><table border=1> <tr><td>公司名称</td><td>最新失信发布时间</td><td>累计失信次数</td> </tr> <tr><td>潮州市澳士通陶瓷实业有限公司</td><td>3月15日</td><td>1次</td> </tr> <tr><td>肇庆市德圣陶瓷有限公司</td><td>3月14日</td><td>6次</td> </tr> <tr><td>清远市港龙陶瓷有限公司</td><td>3月10日</td><td>28次</td> </tr> <tr><td>广西金沙江陶瓷有限公司</td><td>3月14日</td><td>3次</td> </tr> <tr><td>宜丰县凯扬陶瓷发展有限公司</td><td>3月29日</td><td>11次</td> </tr> <tr><td>江西领先精工陶瓷发展有限公司</td><td>3月29日</td><td>23次</td> </tr> <tr><td>高安弘建陶瓷有限公司</td><td>3月28日</td><td>5次</td> </tr></table>\\\"}"
  }
}
```

Result反序列化

将上面Result反序列化展示如下。

```
{
  "Content": "<div><h2>2017年3月40多家陶企上榜失信被执行人名单</h2><div>1月7日，陶卫网记者根据最高人民法院及各地法院发布的失信被执行人信息统计，2019年12月全国各地有112家陶瓷企业被列入“失信被执行人”名单，名单涉及21个省（市）。此次名单中，广东省的失信陶企多达28家。</div><div>据统计，这些“失信被执行人”的上榜原因主要为债务纠纷，欠款不还，大部分为拖欠供应商货款、工资、员工赔偿、银行贷款以及融资租赁租金等，也有涉及运输、燃气等费用，小部分企业涉及承担债务连带清偿责任。</div><table border=1> <tr><td>公司名称</td><td>最新失信发布时间</td><td>累计失信次数</td> </tr> <tr><td>潮州市澳士通陶瓷实业有限公司</td><td>3月15日</td><td>1次</td> </tr> <tr><td>肇庆市德圣陶瓷有限公司</td><td>3月14日</td><td>6次</td> </tr> <tr><td>清远市港龙陶瓷有限公司</td><td>3月10日</td><td>28次</td> </tr> <tr><td>广西金沙江陶瓷有限公司</td><td>3月14日</td><td>3次</td> </tr> <tr><td>宜丰县凯扬陶瓷发展有限公司</td><td>3月29日</td><td>11次</td> </tr> <tr><td>江西领先精工陶瓷发展有限公司</td><td>3月29日</td><td>23次</td> </tr> <tr><td>高安弘建陶瓷有限公司</td><td>3月28日</td><td>5次</td> </tr></table>"
}
```

错误码

具体请参见[常见错误码](#)。

SDK参考

阿里云视觉AI能力支持使用SDK调用，具体可参见[SDK总览](#)下载安装。

安全声明

- 请确保上传的图片或文件来源符合相应的法律法规。
- 通过体验调试上传的临时文件有效期为1小时，在24小时后会被系统自动清理删除。

8. 公共信息

8.1. 请求结构

文字识别服务支持基于URL发送HTTP/HTTPS请求。请求参数需要包含在URL中，请求及返回结果都使用UTF-8 字符集编码。

以下为一条未编码的URL请求示例：

```
https://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeBankCard<公共请求参数>
```

- `https`：指定了请求通信协议。
- `ocr.cn-shanghai.aliyuncs.com`：指定了文字识别服务的接入地址（Endpoint）。
- `cn-shanghai`：区域（RegionId），请参见[访问域名查看支持的域名](#)。
- `Action=RecognizeBankCard`：指定了要调用的API。
- `<公共请求参数>`：API接口中使用了公共请求头（Common Request Headers），该内容可以被所有的视觉智能服务请求使用。详细说明请参见[公共请求参数](#)。

8.2. 公共请求参数

阿里云视觉智能开放平台是基于HTTP协议的RESTful API类型接口，每个API都必须提供这些公共请求参数。

请求示例

```
http://ocr.cn-shanghai.aliyuncs.com/?Signature=NAxwl1W9ROkidJfGeZrsKUXw%2BQ****  
&AccessKeyId=testId  
&Action=RecognizeBankCard  
&Format=JSON  
&RegionId=cn-shanghai  
&SignatureMethod=HMAC-SHA1  
&SignatureNonce=313ef0fb-0393-464f-a6e2-59d9ca2585b1  
&SignatureVersion=1.0  
&Timestamp=2019-10-12T08%3A18%3A18Z  
&Version=2019-12-30
```

请求参数

名称	类型	是否必填	说明
Signature	String	是	请求签名，即最终生成的签名结果值。如何生成请求签名，请参见 请求签名 。

名称	类型	是否必填	说明
AccessKeyId	String	是	访问服务使用的密钥ID。具体请参见 创建AccessKey
Action	String	是	API接口名称。
Format	String	否	返回消息的格式。取值：JSON（默认值）或XML。
RegionId	String	否	API支持的区域。例如：cn-shanghai。
SignatureMethod	String	是	签名方式。取值：HMAC-SHA1。
SignatureNonce	String	是	签名唯一随机数。用于防止网络重放攻击，建议您在不同请求使用不同的随机数。JAVA语言建议用 <code>java.util.UUID.randomUUID()</code> 生成。
SignatureVersion	String	是	签名算法版本。取值：1.0。
Timestamp	String	是	请求时间戳，为日期格式。使用UTC时间按照 ISO8601标准，格式为YYYY-MM-DDThh:mm:ssZ。示例：2019-10-13T10:00:00Z 表示北京时间 2019 年 10 月 13 日 18 点 00 分 00 秒。
Version	String	是	API版本号，格式为 YYYY-MM-DD。取值：2019-12-30。

8.3. 公共响应参数

阿里云视觉智能开放平台是基于HTTP协议的RESTful API类型接口，所有API响应均提供一组公共响应头。您每次发送的接口调用请求，无论成功与否，系统都会返回一个唯一识别码RequestId。

请求示例

```
https://ocr.cn-shanghai.aliyuncs.com/
?Format=xml
&Version=2019-12-30
&Signature=Pc5WB8gokVn0xfeu%2FZV%2BiNM1dg****
&SignatureMethod=HMAC-SHA1
&SignatureNonce=15215528852396
&SignatureVersion=1.0
&AccessKeyId=key-test
&Timestamp=2014-10-10T12:00:00Z
```

成功结果示例

调用API服务后，系统采用统一格式返回数据。返回2xx的HTTP状态码表示调用成功，调用成功返回的数据格式主要有XML和JSON两种。外部系统可以在请求时传入参数来制定返回的数据格式，默认为XML格式。为了便于查看，本文档中的返回示例做了格式化处理，实际返回结果没有进行换行、缩进等处理。

- JSON格式

```
{
  "RequestId": "4C467B38-3910-447D-87BC-AC049166F216",
  /* 返回结果数据 */
}
```

- XML格式

```
<?xml version="1.0" encoding="utf-8"?>
<!--结果的根结点-->
<接口名称+Response>
  <!--返回请求标签-->
  <RequestId>4C467B38-3910-447D-87BC-AC049166F216</RequestId>
  <!--返回结果数据-->
</接口名称+Response>
```

错误结果示例

当调用出错时，HTTP请求会返回一个4xx或5xx的HTTP状态码。返回的消息体中包含具体的错误码及错误信息，还包含一个全局唯一的请求ID（RequestId）和一个该次请求访问的站点ID（HostId）。您可以根据返回的错误码来定位错误原因。若无法在调用方定位错误原因，请联系阿里云客服人员，并提供该HostId和RequestId，以便能尽快帮您解决问题。

- JSON格式

```
{
  "RequestId": "7463B73D-35CC-4D19-A010-6B8D65D242EF",
  "HostId": "ocr.cn-shanghai.aliyuncs.com",
  "Code": "UnsupportedOperation",
  "Message": "The specified action is not supported."
}
```

- XML格式

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>
  <RequestId>8906582E-6722-409A-A6C4-0E7863B733A5</RequestId>
  <HostId>ocr.cn-shanghai.aliyuncs.com</HostId>
  <Code>UnsupportedOperation</Code>
  <Message>The specified action is not supported.</Message>
</Error>
```

9. 常见错误码

当API请求发生错误时，服务端会返回错误信息。本文档为您介绍API返回的错误码信息。

文字识别错误码

httpcode	ErrorCode	错误中文解释
400	InvalidImage.Category	输入图像与当前服务不符，如行驶证服务输入的图像不是行驶证。
400	InvalidImage.Content	输入图像内容不合法。例如： <ul style="list-style-type: none"> 无效的base64编码图像。 输入的image是空字符串。 无效的image content。
400	InvalidImage.Decode	图像解码失败。输入的image不是合法的base64字符串。 <ul style="list-style-type: none"> 请删除base64字符串多余的头： data:image/jpg;base64。 base64编码的数据长度必须是4的倍数，如果长度不够，需要在末尾加上 = 补齐。 输入的image不是合法的base64字符串，存在非法字符。
400	InvalidImage.Download	URL下载图像失败。
400	InvalidImage.Gif	GIF图像打开失败、读取失败、解码失败、获取颜色映射表错误、图像为空。
400	InvalidImage.URL	无效的图片URL。
400	InvalidImage.Unsafe	URL不安全。
400	IllegalUrlParameter	URL不合法。
400	MissingImageURL	缺少图像URL参数。
400	InvalidSide	参数side无效。
400	Throttling.User	您这个时段的流量已经超限。如果不能满足现有业务要求可以提工单申请。
400	InvalidParameter	参数值校验不通过。请使用请求参数构造规范化的请求字符串。详细信息请参见 请求签名 。
400	InvalidImage.Region	图片所在地域错误。
400	InvalidResult	OCR识别失败、OCR识别异常。

httpcode	ErrorCode	错误中文解释
400	InvalidTimeStamp.Expired	用户时间和服务器时间差异超过15分钟，请检查您的时间戳设置，确保用户时间和服务器时间的差值在15分钟内。
400	InvalidParameter.Toolarge	待检测内容过大，请确保待检测内容在API的限制范围内。
400	InvalidOutputFormat	无效的输出格式。
400	MissingOutputFormat	缺少输出格式。
400	InvalidImageType	无效的图片类型。
400	MissingUseFinanceModel	缺少UseFinanceModel参数信息。
400	MissingAssureDirection	缺少AssureDirection参数信息。
400	MissingSide	缺少side信息。
400	MissingSkipDetection	缺少SkipDetection参数。
400	MissingMinHeight	缺少MinHeight参数。
400	MissingAccessKeyId	缺少AK信息。具体请参见 创建AccessKey 。
400	SignatureNonceUsed	SignatureNonce重复。每次请求的SignatureNonce在15分钟内不能重复。
400	SignatureDoesNotMatch	签名不匹配。请检查AccessKeyId和AccessKeySecret是否正确；检查签名方法是否正确。关于签名的详细信息请参见 请求签名 。
403	Unauthorized	请求权限不够，请为账号授权。
403	ParameterError	参数错误，请检查参数值。
403	UnsupportedHTTPMethod	HTTP请求方法不支持。
403	InvalidApi.NotPurchase	服务未开通，请先开通服务。具体请参见 开通视觉智能开放平台服务 。
404	InvalidAccessKeyId.NotFound	AccessKeyId找不到。请检查调用时是否使用了正确的AccessKey。具体请参见 创建AccessKey 。
408	Timeout	请求超时。
415	InvalidImage.UnsupportedMediaType	输入的image解码失败。

httpcode	ErrorCode	错误中文解释
500	InternalServerError	内部异常。
503	ServiceUnavailable	服务不可用。

公共错误码

API的公共错误码请参见[API错误中心](#)。

处理建议

请求服务出现错误码提示，请参见错误码描述查看具体错误原因，并进行错误处理。针对一些常见高频的错误类型，此处提供一些错误处理建议供您参考。

- URL类错误码

此类错误码一般都是文件URL不符合要求，例如URL不安全、无效。建议确认当前URL是否正确，或者您可以重新生成URL，具体请参见[生成URL](#)。

- 图像加载错误码

此类错误也可以看做是图像URL错误，例如图像下载错误、地域错误。当前我们仅支持 华东2（上海） 地域，如果您通过OSS生成URL，那么您在OSS上创建的Bucket所属地域应该为 华东2（上海） 。

- 参数类错误码

此类错误码一般是输入参数不符合要求，例如参数缺失、参数类型错误等。建议您检查输入参数是否符合要求，您可以参考对应API文档中的请求参数表格进行检查确认。

- 系统服务类错误码

如果出现此类错误码，建议您重新发起服务请求，如果问题依然存在，请在钉钉群组（32665857）中反馈问题。