ALIBABA CLOUD

阿里云

视觉智能开放平台

文字识别

文档版本：20220310

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 `cd /d C:/window` 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	`bae log list --instanceid` *Instance_ID*
[] 或者 [a|b]	表示可选项，至多选择一个。	`ipconfig [-all|-t]`
{} 或者 {a|b}	表示必选项，至多选择一个。	`switch {active|stand}`

目录

1.文字识别介绍 06
2.计费介绍 09
3.个人类卡证识别 25
3.1. 身份证识别 25
3.2. 户口页识别 31
3.3. 银行卡识别 36
3.4. 名片识别 39
3.5. 二维码识别 41
3.6. 中国护照识别 45
3.7. 护照MRZ码识别 49
4.资产类证件识别 55
4.1. 营业执照识别 55
4.2. 公章识别 61
5.通用文字类识别 65
5.1. 通用文字识别 65
5.2. 表格识别 69
5.3. 文档结构化还原识别 89
5.4. 证件翻拍识别 92
5.5. 门头照识别 95
5.6. PDF识别 101
5.7. 视频文字识别 106
6.车辆交通类识别 111
6.1. 驾驶证识别 111
6.2. 行驶证识别 115
6.3. 车牌识别 119
6.4. VIN码识别 123

7.行业票证类识别 126
7.1. 出租车发票识别 126
7.2. 增值税发票识别 137
7.3. 火车票识别 145
7.4. 外卖单识别 147
7.5. 定额发票识别 152
7.6. 增值税发票卷票识别 158
8.查询异步任务结果 166
9.公共信息 172
9.1. 请求结构 172
9.2. 公共请求参数 172
9.3. 公共响应参数 173
10.常见错误码 175

1.文字识别介绍

文字识别技术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新手指南。

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	通用文字识别	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。
	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）。
	证件翻拍识别	可以识别中国身份证照片是否为屏幕翻拍图。
	门头照识别	可以识别门头招牌的图像，并且过滤出门头信息，例如门头照、Logo、店铺地址、联系方式等。
	PDF识别	可以对PDF上的文字进行结构化识别。
	视频文字识别	可以实现对输入视频的结构化处理，返回对应的文字内容、文字区域坐标及时间等信息。
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	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。
	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。

类别	能力	说明
	定额发票识别	可以对定额发票上的发票号码、发票代码、发票金额进行结构化识别。
	增值税发票卷票识别	支持对卷票上包括价税合计、发票代码、发票号码、合计税额、合计金额、密码区、开票日期、税率、购买方识别号、销售方识别号等结构化识别。

应用场景

文字识别应用场景如下：

- 注册登记审核

  通过身份证、护照识别、银行卡识别等能力，在互联网金融、公共网上办事大厅等场景，能够快速准确识别录入用户身份、账户信息，准确率高达99%，从而降低用户输入成本，提高人工审核效率，有效控制业务风险。

- 报销凭证录入

  应用于企业税务核算、金融保费核发及内部报销等场景，实现对各类税务发票、医疗发票、交通票据等的自动识别和快速录入，能够有效减少人力成本，实现报销的自动化。

更多产品动态，请您持续关注阿里云视觉智能开放平台。

2.计费介绍

本文主要介绍文字识别收费能力的计费方式及报价。其余未收费能力当前还处于公测阶段，可免费使用。

车牌识别

车牌识别能力支持**按量付费**和**预付费资源包**方式进行计费。

- **按量付费**

调用量	价格
0 < 调用量（次/月）≤10,000	0.15元/次
10,000 < 调用量（次/月）≤100,000	0.1元/次
100,000 < 调用量（次/月）≤1,000,000	0.06元/次
调用量（次/月）> 1,000,000	0.02元/次

> **注意**
> - 如果您未购买资源包，系统将会按照**按量付费**的价格计费，请留意并及时支付账单，避免因欠费等原因影响您的业务。欠费24小时后会冻结调用该能力，支付欠费账单后可恢复使用。
> - 调用失败不计费。
> - 价格费用计算详情请参见费用计算及购买说明。

- **预付费资源包**

阿里云视觉智能开放平台提供首次购买1000点规格享全单0.01元的优惠活动。资源包支持叠加购买，当资源包点数耗尽或过期后，将按照**按量付费**进行计费。您可根据实际情况按需购买资源包。资源包有效期**1年**，价格如下所示。

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
1000点	1000次	1点/次	120元	83.29元	首次享0.01元购买
10,000点	10,000次		800元	550元	无
100,000点	100,000次		5000元	3300元	无
500,000点	500,000次		20,000元	13,750元	无
1,000,000点	1,000,000次		30,000元	16,500元	无
5,000,000点	5,000,000次		100,000元	55,000元	无
10,000,000点	10,000,000次		200,000元	80,000元	无

注意

- 资源包不可跨类目使用。例如，使用车牌识别和物体检测两个能力，需要同时购买车牌识别和物体检测两个资源包。
- 价格费用计算详情请参见费用计算及购买说明。
- 如果因业务需求购买更多QPS，请提交工单或通过咨询服务联系我们。

驾驶证识别

驾驶证识别能力支持**按量付费**和**预付费资源包**方式进行计费。

- **按量付费**

调用量	价格
0 < 调用量（次/月）≤10,000	0.0825元/次
10,000 < 调用量（次/月）≤100,000	0.0495元/次
100,000 < 调用量（次/月）≤500,000	0.0415元/次
500,000 < 调用量（次/月）≤1,000,000	0.0248元/次
调用量（次/月）> 1,000,000	0.009元/次

注意

- 如果您未购买资源包，系统将会按照**按量付费**的价格计费，请留意并及时支付账单，避免因欠费等原因影响您的业务。欠费24小时后会冻结调用该能力，支付欠费账单后可恢复使用。
- 调用失败不计费。
- 价格费用计算详情请参见费用计算及购买说明。

- **预付费资源包**

阿里云视觉智能开放平台提供首次购买1000点规格享全单0.01元的优惠活动。资源包支持叠加购买，当资源包点数耗尽或过期后，将按照**按量付费**进行计费。您可根据实际情况按需购买资源包。资源包有效期1年，价格如下所示。

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
1000点	1000次	1点/次	120元	83.29元	首次享0.01元购买
10,000点	10,000次		800元	550元	无
100,000点	100,000次		5000元	3300元	无
500,000点	500,000次		20,000元	13,750元	无
1,000,000点	1,000,000次		30,000元	16,500元	无
5,000,000点	5,000,000次		100,000元	55,000元	无

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
10,000,000点	10,000,000次		200,000元	80,000元	无

注意

- 资源包不可跨类目使用。例如，使用车牌识别和物体检测两个能力，需要同时购买车牌识别和物体检测两个资源包。
- 价格费用计算详情请参见费用计算及购买说明。
- 如果因业务需求购买更多QPS，请提交工单或通过咨询服务联系我们。

行驶证识别

行驶证识别能力支持**按量付费**和**预付费资源包**方式进行计费。

- **按量付费**

调用量	价格
0 < 调用量（次/月）≤10,000	0.0825元/次
10,000 < 调用量（次/月）≤100,000	0.0495元/次
100,000 < 调用量（次/月）≤500,000	0.0415元/次
500,000 < 调用量（次/月）≤1,000,000	0.0248元/次
调用量（次/月）> 1,000,000	0.009元/次

注意

- 如果您未购买资源包，系统将会按照**按量付费**的价格计费，请留意并及时支付账单，避免因欠费等原因影响您的业务。欠费24小时后会冻结调用该能力，支付欠费账单后可恢复使用。
- 调用失败不计费。
- 价格费用计算详情请参见费用计算及购买说明。

- **预付费资源包**

阿里云视觉智能开放平台提供首次购买1000点规格享全单0.01元的优惠活动。资源包支持叠加购买，当资源包点数耗尽或过期后，将按照**按量付费**进行计费。您可根据实际情况按需购买资源包。资源包有效期1年，价格如下所示。

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
1000点	1000次		120元	83.29元	首次享0.01元购买
10,000点	10,000次		800元	550元	无
100,000点	100,000次		5000元	3300元	无

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
500,000点	500,000次	1点/次	20,000元	13,750元	无
1,000,000点	1,000,000次		30,000元	16,500元	无
5,000,000点	5,000,000次		100,000元	55,000元	无
10,000,000点	10,000,000次		200,000元	80,000元	无

注意

- 资源包不可跨类目使用。例如，使用车牌识别和物体检测两个能力，需要同时购买车牌识别和物体检测两个资源包。
- 价格费用计算详情请参见费用计算及购买说明。
- 如果因业务需求购买更多QPS，请提交工单或通过咨询服务联系我们。

通用文字识别

通用文字识别能力支持**按量付费**和**预付费资源包**方式进行计费。

- **按量付费**

调用量	价格
0 < 调用量（次/月）≤10,000	0.0825元/次
10,000 < 调用量（次/月）≤100,000	0.0495元/次
100,000 < 调用量（次/月）≤500,000	0.0415元/次
500,000 < 调用量（次/月）≤1,000,000	0.0248元/次
调用量（次/月）> 1,000,000	0.009元/次

注意

- 如果您未购买资源包，系统将会按照**按量付费**的价格计费，请留意并及时支付账单，避免因欠费等原因影响您的业务。欠费24小时后会冻结调用该能力，支付欠费账单后可恢复使用。
- 调用失败不计费。
- 价格费用计算详情请参见费用计算及购买说明。

- **预付费资源包**

阿里云视觉智能开放平台提供首次购买1000点规格享全单0.01元的优惠活动。资源包支持叠加购买，当资源包点数耗尽或过期后，将按照**按量付费**进行计费。您可根据实际情况按需购买资源包。资源包有效期**1年**，价格如下所示。

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
1000点	1000次	1点/次	120元	83.29元	首次享0.01元购买
10,000点	10,000次		800元	550元	无
100,000点	100,000次		5000元	3300元	无
500,000点	500,000次		20,000元	13,750元	无
1,000,000点	1,000,000次		30,000元	16,500元	无
5,000,000点	5,000,000次		100,000元	55,000元	无
10,000,000点	10,000,000次		200,000元	80,000元	无

注意

- 资源包不可跨类目使用。例如，使用车牌识别和物体检测两个能力，需要同时购买车牌识别和物体检测两个资源包。
- 价格费用计算详情请参见费用计算及购买说明。
- 如果因业务需求购买更多QPS，请提交工单或通过咨询服务联系我们。

身份证识别

身份证识别能力支持**按量付费**和**预付费资源包**方式进行计费。

- **按量付费**

调用量	价格
0 < 调用量（次/月）≤10,000	0.0825元/次
10,000 < 调用量（次/月）≤100,000	0.0495元/次
100,000 < 调用量（次/月）≤500,000	0.0415元/次
500,000 < 调用量（次/月）≤1,000,000	0.0248元/次
调用量（次/月）> 1,000,000	0.009元/次

注意

- 如果您未购买资源包，系统将会按照**按量付费**的价格计费，请留意并及时支付账单，避免因欠费等原因影响您的业务。欠费24小时后会冻结调用该能力，支付欠费账单后可恢复使用。
- 调用失败不计费。
- 价格费用计算详情请参见费用计算及购买说明。

- **预付费资源包**

阿里云视觉智能开放平台提供首次购买1000点规格享全单0.01元的优惠活动。资源包支持叠加购买，当资源包点数耗尽或过期后，将按照**按量付费**进行计费。您可根据实际情况按需购买资源包。资源包有效期**1年**，价格如下所示。

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
1000点	1000次	1点/次	120元	83.29元	首次享0.01元购买
10,000点	10,000次		800元	550元	无
100,000点	100,000次		5000元	3300元	无
500,000点	500,000次		20,000元	13,750元	无
1,000,000点	1,000,000次		30,000元	16,500元	无
5,000,000点	5,000,000次		100,000元	55,000元	无
10,000,000点	10,000,000次		200,000元	80,000元	无

注意

- 资源包不可跨类目使用。例如，使用车牌识别和物体检测两个能力，需要同时购买车牌识别和物体检测两个资源包。
- 价格费用计算详情请参见费用计算及购买说明。
- 如果因业务需求购买更多QPS，请提交工单或通过咨询服务联系我们。

增值税发票识别

增值税发票识别能力支持**按量付费**和**预付费资源包**方式进行计费。

- **按量付费**

调用量	价格
0 < 调用量（次/月）≤10,000	0.225元/次
10,000 < 调用量（次/月）≤100,000	0.09元/次
100,000 < 调用量（次/月）≤500,000	0.054元/次
500,000 < 调用量（次/月）≤1,000,000	0.045元/次
调用量（次/月）> 1,000,000	0.036元/次

注意

- 如果您未购买资源包，系统将会按照**按量付费**的价格计费，请留意并及时支付账单，避免因欠费等原因影响您的业务。欠费24小时后会冻结调用该能力，支付欠费账单后可恢复使用。
- 调用失败不计费。
- 价格费用计算详情请参见费用计算及购买说明。

- **预付费资源包**

阿里云视觉智能开放平台提供首次购买1000点规格享全单0.01元的优惠活动。资源包支持叠加购买，当资源包点数耗尽或过期后，将按照**按量付费**进行计费。您可根据实际情况按需购买资源包。资源包有效期**1年**，价格如下所示。

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
1000点	459次	2.18点/次	120元	83.29元	首次享0.01元购买
10,000点	4587次		800元	550元	无
100,000点	45,871次		5000元	3300元	无
500,000点	229,358次		20,000元	13,750元	无
1,000,000点	458,716次		30,000元	16,500元	无
5,000,000点	2,293,578次		100,000元	55,000元	无
10,000,000点	4,587,156次		200,000元	80,000元	无

注意

- 资源包不可跨类目使用。例如，使用车牌识别和物体检测两个能力，需要同时购买车牌识别和物体检测两个资源包。
- 价格费用计算详情请参见费用计算及购买说明。
- 如果因业务需求购买更多QPS，请提交工单或通过咨询服务联系我们。

VIN码识别

VIN码识别能力支持**按量付费**和**预付费资源包**方式进行计费。

- **按量付费**

调用量	价格
0 < 调用量（次/月）≤10,000	0.048元/次
10,000 < 调用量（次/月）≤100,000	0.0144元/次
100,000 < 调用量（次/月）≤500,000	0.009元/次
500,000 < 调用量（次/月）≤1,000,000	0.0084元/次

调用量	价格
调用量（次/月）＞1,000,000	0.0068元/次

注意

- 如果您未购买资源包，系统将会按照**按量付费**的价格计费，请留意并及时支付账单，避免因欠费等原因影响您的业务。欠费24小时后会冻结调用该能力，支付欠费账单后可恢复使用。
- 调用失败不计费。
- 价格费用计算详情请参见费用计算及购买说明。

- **预付费资源包**

阿里云视觉智能开放平台提供首次购买1000点规格享全单0.01元的优惠活动。资源包支持叠加购买，当资源包点数耗尽或过期后，将按照**按量付费**进行计费。您可根据实际情况按需购买资源包。资源包有效期**1年**，价格如下所示。

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
1000点	2500次	0.4点/次	120元	83.29元	首次享0.01元购买
10,000点	25,000次		800元	550元	无
100,000点	250,000次		5000元	3300元	无
500,000点	1,250,000次		20,000元	13,750元	无
1,000,000点	2,500,000次		30,000元	16,500元	无
5,000,000点	12,500,000次		100,000元	55,000元	无
10,000,000点	25,000,000次		200,000元	80,000元	无

注意

- 资源包不可跨类目使用。例如，使用车牌识别和物体检测两个能力，需要同时购买车牌识别和物体检测两个资源包。
- 价格费用计算详情请参见费用计算及购买说明。
- 如果因业务需求购买更多QPS，请提交工单或通过咨询服务联系我们。

银行卡识别

银行卡识别能力支持**按量付费**和**预付费资源包**方式进行计费。

- **按量付费**

调用量	价格
0＜调用量（次/月）≤10,000	0.0825元/次

调用量	价格
10,000 < 调用量（次/月）≤100,000	0.0495元/次
100,000 < 调用量（次/月）≤500,000	0.0415元/次
500,000 < 调用量（次/月）≤1,000,000	0.0248元/次
调用量（次/月）> 1,000,000	0.009元/次

注意

- 如果您未购买资源包，系统将会按照**按量付费**的价格计费，请留意并及时支付账单，避免因欠费等原因影响您的业务。欠费24小时后会冻结调用该能力，支付欠费账单后可恢复使用。
- 调用失败不计费。
- 价格费用计算详情请参见费用计算及购买说明。

- **预付费资源包**

阿里云视觉智能开放平台提供首次购买1000点规格享全单0.01元的优惠活动。资源包支持叠加购买，当资源包点数耗尽或过期后，将按照**按量付费**进行计费。您可根据实际情况按需购买资源包。资源包有效期1年，价格如下所示。

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
1000点	1000次	1点/次	120元	83.29元	首次享0.01元购买
10,000点	10,000次		800元	550元	无
100,000点	100,000次		5000元	3300元	无
500,000点	500,000次		20,000元	13,750元	无
1,000,000点	1,000,000次		30,000元	16,500元	无
5,000,000点	5,000,000次		100,000元	55,000元	无
10,000,000点	10,000,000次		200,000元	80,000元	无

注意

- 资源包不可跨类目使用。例如，使用车牌识别和物体检测两个能力，需要同时购买车牌识别和物体检测两个资源包。
- 价格费用计算详情请参见费用计算及购买说明。
- 如果因业务需求购买更多QPS，请提交工单或通过咨询服务联系我们。

营业执照识别

营业执照识别能力支持**按量付费**和**预付费资源包**方式进行计费。

- **按量付费**

调用量	价格
0 < 调用量（次/月）≤10,000	0.0825元/次
10,000 < 调用量（次/月）≤100,000	0.0495元/次
100,000 < 调用量（次/月）≤500,000	0.0415元/次
500,000 < 调用量（次/月）≤1,000,000	0.0248元/次
调用量（次/月）> 1,000,000	0.009元/次

注意

- 如果您未购买资源包，系统将会按照**按量付费**的价格计费，请留意并及时支付账单，避免因欠费等原因影响您的业务。欠费24小时后会冻结调用该能力，支付欠费账单后可恢复使用。
- 调用失败不计费。
- 价格费用计算详情请参见费用计算及购买说明。

- **预付费资源包**

阿里云视觉智能开放平台提供首次购买1000点规格享全单0.01元的优惠活动。资源包支持叠加购买，当资源包点数耗尽或过期后，将按照**按量付费**进行计费。您可根据实际情况按需购买资源包。资源包有效期1年，价格如下所示。

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
1000点	1000次	1点/次	120元	83.29元	首次享0.01元购买
10,000点	10,000次		800元	550元	无
100,000点	100,000次		5000元	3300元	无
500,000点	500,000次		20,000元	13,750元	无
1,000,000点	1,000,000次		30,000元	16,500元	无
5,000,000点	5,000,000次		100,000元	55,000元	无
10,000,000点	10,000,000次		200,000元	80,000元	无

注意

- 资源包不可跨类目使用。例如，使用车牌识别和物体检测两个能力，需要同时购买车牌识别和物体检测两个资源包。
- 价格费用计算详情请参见费用计算及购买说明。
- 如果因业务需求购买更多QPS，请提交工单或通过咨询服务联系我们。

二维码识别

二维码识别能力支持**按量付费**和**预付费资源包**方式进行计费。

- **按量付费**

调用量	价格
0 < 调用量（次/月）≤10,000	0.024元/次
10,000 < 调用量（次/月）≤100,000	0.0072元/次
100,000 < 调用量（次/月）≤500,000	0.0045元/次
500,000 < 调用量（次/月）≤1,000,000	0.0042元/次
调用量（次/月）> 1,000,000	0.0034元/次

注意

- 如果您未购买资源包，系统将会按照**按量付费**的价格计费，请留意并及时支付账单，避免因欠费等原因影响您的业务。欠费24小时后会冻结调用该能力，支付欠费账单后可恢复使用。
- 调用失败不计费。
- 价格费用计算详情请参见费用计算及购买说明。

- **预付费资源包**

阿里云视觉智能开放平台提供首次购买1000点规格享全单0.01元的优惠活动。资源包支持叠加购买，当资源包点数耗尽或过期后，将按照**按量付费**进行计费。您可根据实际情况按需购买资源包。资源包有效期**1年**，价格如下所示。

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
1000点	5000次	0.2点/次	120元	83.29元	首次享0.01元购买
10,000点	50,000次		800元	550元	无
100,000点	500,000次		5000元	3300元	无
500,000点	2,500,000次		20,000元	13,750元	无
1,000,000点	5,000,000次		30,000元	16,500元	无
5,000,000点	25,000,000次		100,000元	55,000元	无
10,000,000点	50,000,000次		200,000元	80,000元	无

注意

- 资源包不可跨类目使用。例如，使用车牌识别和物体检测两个能力，需要同时购买车牌识别和物体检测两个资源包。
- 价格费用计算详情请参见费用计算及购买说明。
- 如果因业务需求购买更多QPS，请提交工单或通过咨询服务联系我们。

PDF识别

PDF识别能力支持**按量付费**和**预付费资源包**方式进行计费。

- **按量付费**

调用量	价格
0 < 调用量（次/月）≤10,000	0.225元/次
10,000 < 调用量（次/月）≤100,000	0.09元/次
100,000 < 调用量（次/月）≤500,000	0.054元/次
500,000 < 调用量（次/月）≤1,000,000	0.045元/次
调用量（次/月）> 1,000,000	0.036元/次

注意

- 如果您未购买资源包，系统将会按照**按量付费**的价格计费，请留意并及时支付账单，避免因欠费等原因影响您的业务。欠费24小时后会冻结调用该能力，支付欠费账单后可恢复使用。
- 调用失败不计费。
- 价格费用计算详情请参见费用计算及购买说明。

- **预付费资源包**

阿里云视觉智能开放平台提供首次购买1000点规格享全单0.01元的优惠活动。资源包支持叠加购买，当资源包点数耗尽或过期后，将按照**按量付费**进行计费。您可根据实际情况按需购买资源包。资源包有效期1年，价格如下所示。

资源包规格	可调用次数	抵扣系数	标准价格	价格	优惠政策
1000点	459次	2.18点/次	120元	83.29元	首次享0.01元购买
10,000点	4,587次		800元	550元	无
100,000点	45,871次		5000元	3300元	无
500,000点	229,358次		20,000元	13,750元	无
1,000,000点	458,716次		30,000元	16,500元	无
5,000,000点	2,293,578次		100,000元	55,000元	无

资源包规格	可调用次数	抵扣系数	标准价格	价格	优惠政策
10,000,000点	4,587,156次		200,000元	80,000元	无

注意

- 资源包不可跨类目使用。例如，使用车牌识别和物体检测两个能力，需要同时购买车牌识别和物体检测两个资源包。
- 价格费用计算详情请参见费用计算及购买说明。
- 如果因业务需求购买更多QPS，请提交工单或通过咨询服务联系我们。

定额发票识别

定额发票识别能力支持**按量付费**和**预付费资源包**方式进行计费。

- **按量付费**

调用量	价格
0 < 调用量（次/月）≤10,000	0.0825元/次
10,000 < 调用量（次/月）≤100,000	0.0495元/次
100,000 < 调用量（次/月）≤500,000	0.0415元/次
500,000 < 调用量（次/月）≤1,000,000	0.0248元/次
调用量（次/月）> 1,000,000	0.009元/次

注意

- 如果您未购买资源包，系统将会按照**按量付费**的价格计费，请留意并及时支付账单，避免因欠费等原因影响您的业务。欠费24小时后会冻结调用该能力，支付欠费账单后可恢复使用。
- 调用失败不计费。
- 价格费用计算详情请参见费用计算及购买说明。

- **预付费资源包**

阿里云视觉智能开放平台提供首次购买1000点规格享全单0.01元的优惠活动。资源包支持叠加购买，当资源包点数耗尽或过期后，将按照**按量付费**进行计费。您可根据实际情况按需购买资源包。资源包有效期1年，价格如下所示。

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
1000点	1000次		120元	83.29元	首次享0.01元购买
10,000点	10,000次		800元	550元	无
100,000点	100,000次		5000元	3300元	无

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
500,000点	500,000次	1点/次	20,000元	13,750元	无
1,000,000点	1,000,000次		30,000元	16,500元	无
5,000,000点	5,000,000次		100,000元	55,000元	无
10,000,000点	10,000,000次		200,000元	80,000元	无

注意

- 资源包不可跨类目使用。例如，使用车牌识别和物体检测两个能力，需要同时购买车牌识别和物体检测两个资源包。
- 价格费用计算详情请参见费用计算及购买说明。
- 如果因业务需求购买更多QPS，请提交工单或通过咨询服务联系我们。

增值税发票卷票识别

增值税发票卷票识别能力支持**按量付费**和**预付费资源包**方式进行计费。

- **按量付费**

调用量	价格
0 < 调用量（次/月）≤10,000	0.0825元/次
10,000 < 调用量（次/月）≤100,000	0.0495元/次
100,000 < 调用量（次/月）≤500,000	0.0415元/次
500,000 < 调用量（次/月）≤1,000,000	0.0248元/次
调用量（次/月）> 1,000,000	0.009元/次

注意

- 如果您未购买资源包，系统将会按照**按量付费**的价格计费，请留意并及时支付账单，避免因欠费等原因影响您的业务。欠费24小时后会冻结调用该能力，支付欠费账单后可恢复使用。
- 调用失败不计费。
- 价格费用计算详情请参见费用计算及购买说明。

- **预付费资源包**

阿里云视觉智能开放平台提供首次购买1000点规格享全单0.01元的优惠活动。资源包支持叠加购买，当资源包点数耗尽或过期后，将按照**按量付费**进行计费。您可根据实际情况按需购买资源包。资源包有效期**1年**，价格如下所示。

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策

资源包规格	可调用次数	抵扣系数	标准价格	优惠价格	优惠政策
1000点	1000次	1点/次	120元	83.29元	首次享0.01元购买
10,000点	10,000次		800元	550元	无
100,000点	100,000次		5000元	3300元	无
500,000点	500,000次		20,000元	13,750元	无
1,000,000点	1,000,000次		30,000元	16,500元	无
5,000,000点	5,000,000次		100,000元	55,000元	无
10,000,000点	10,000,000次		200,000元	80,000元	无

注意

- 资源包不可跨类目使用。例如，使用车牌识别和物体检测两个能力，需要同时购买车牌识别和物体检测两个资源包。
- 价格费用计算详情请参见费用计算及购买说明。
- 如果因业务需求购买更多QPS，请提交工单或通过咨询服务联系我们。

视频文字识别

视频文字识别能力支持**按量付费**方式进行计费。每自然月**前300秒**调用免费。

采集规格	分辨率规格	输入分辨率	分辨率取值范围	价格
1帧/秒	SD	分辨率≤854×480像素	Max（宽，高）≤854像素，且Min（宽，高）≤480像素	0.6元/分钟
	HD	854×480 < 分辨率≤1280×720像素	854像素 < Max（宽，高）≤1280像素，且Min（宽，高）≤720像素；或480像素 < Min（宽，高）≤720像素，且Max（宽，高）≤1280像素	0.9元/分钟
	2K	1280×720 < 分辨率≤2560×1440像素	1280像素 < Max（宽，高）≤2560像素，且Min（宽，高）≤1440像素；或720像素 < Min（宽，高）≤1440像素，且Max（宽，高）≤2560像素	1.8元/分钟
	4K	分辨率 > 2560×1440像素	Max（宽，高）> 2560像素，且Min（宽，高）> 1440像素	3.6元/分钟

注意

- 对于输入视频图像最长边和最短边均需要满足对应规格的分辨率要求。例如，分辨率为1080×760像素的视频图像会被系统判断为2K规格。系统将根据输入图像分辨率自动判断规格，按照SD、HD、2K或4K各规格进行计费。
- 按照视频**输入时长**进行计费，时长最小计量单位为秒，时长不足1秒按照1秒进行计费。
- 该能力为异步能力，调用失败不计费，通过RequestId查询结果不计费。
- 价格费用计算详情请参见费用计算及购买说明。

咨询服务

如果您有任何购买问题需要咨询阿里云视觉智能开放平台，欢迎各位企业用户、开发商、服务商或者开发者通过钉钉搜索群号23109592，加入阿里云视觉智能开放平台咨询群，与平台取得联系。

3.个人类卡证识别

3.1. 身份证识别

本文介绍身份证识别RecognizeIdentityCard的语法及示例。

功能描述

身份证识别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。

应用场景

远程注册：识别用户提交的身份证内容，自动完成用户身份信息填充。

特色优势

识别完整：支持识别身份证内各项内容。

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

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：不超过3 MB。
- 图像分辨率：大于15×15像素，小于4096×4096像素。
- URL地址中不能包含中文字符。

计费说明

关于身份证识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口，如需免费体验调试请前往体验中心。

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	系统规定参数。取值：RecognizeIdentityCard。

名称	类型	是否必选	示例值	描述
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeIdentityCard/sfz1.jpg	图像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。 • face：人像面。 • back：国徽面。

返回数据

名称	类型	示例值	描述
RequestId	String	D3F5BA69-79C4-46A4-B02B-58C4EEBC4C33	请求ID。
Data	Object		返回的结果数据内容。
BackResult	Object		国徽面结果。
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	19970101	有效期起始时间，格式：YYYYMMDD，例如19800101，即1980年01月01日。
FrontResult	Object		人像面结果。
FaceRectangle	Object		人脸位置。该字段已废弃。
Size	Object		人脸矩形尺寸。该字段已废弃。
Width	Float	118.16333770751953	宽度。该字段已废弃。

名称	类型	示例值	描述
Height	Float	97.06315612792968 8	高度。该字段已废弃。
Angle	Float	- 87.71058654785156 2	表示矩形顺时针旋转的度数，范围-180~180。该字段已废弃。
Center	Object		人脸矩形中心坐标。该字段已废弃。
Y	Float	225.2064361572265 6	人脸矩形中心纵坐标。该字段已废弃。
X	Float	475.5939025878906 2	人脸矩形中心横坐标。该字段已废弃。
BirthDate	String	19960111	出生日期，格式：YYYYMMDD，例如19800101，即1980年01月01日。
Gender	String	男	性别。
Address	String	浙江省杭州市余杭区余杭小区6栋602	地址信息。
FaceRectVertices	Array of FaceRectVertice		人脸位置，四个顶点表示。该字段已废弃。
Y	Float	164.2332153320312 5	人脸位置纵坐标。该字段已废弃。
X	Float	429.4612426757812 5	人脸位置横坐标。该字段已废弃。
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）。
Y	Float	81	身份证区域纵坐标。
X	Float	40	身份证区域横坐标。
Nationality	String	汉	民族。
Name	String	张三	姓名。
IDNumber	String	310228199601115411	身份证号。

名称	类型	示例值	描述

示例

请求示例

```
http(s)://[Endpoint]/?Action=RecognizeIdentityCard
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeIdentityCard/sfz1.jpg
&Side=face
&公共请求参数
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RequestId>D3F5BA69-79C4-46A4-B02B-58C4EEBC4C33</RequestId>
<Data>
    <FrontResult>
        <Address>浙江省杭州市余杭区余杭小区6栋602</Address>
        <FaceRectVertices>
            <X>521.7265625</X>
            <Y>286.1796569824219</Y>
        </FaceRectVertices>
        <FaceRectVertices>
            <X>424.7409362792969</X>
            <Y>282.30224609375</Y>
        </FaceRectVertices>
        <FaceRectVertices>
            <X>429.46124267578125</X>
            <Y>164.23321533203125</Y>
        </FaceRectVertices>
        <FaceRectVertices>
            <X>526.4468994140625</X>
            <Y>168.11062622070312</Y>
        </FaceRectVertices>
        <FaceRectangle>
            <Center>
                <X>475.5939025878906</X>
                <Y>225.20643615722656</Y>
            </Center>
            <Angle>-87.71058654785156</Angle>
            <Size>
                <Height>97.06315612792969</Height>
                <Width>118.16333770751953</Width>
            </Size>
        </FaceRectangle>
        <CardAreas>
            <X>40</X>
```

```
            <Y>81</Y>
        </CardAreas>
        <CardAreas>
            <X>588</X>
            <Y>80</Y>
        </CardAreas>
        <CardAreas>
            <X>588</X>
            <Y>400</Y>
        </CardAreas>
        <CardAreas>
            <X>40</X>
            <Y>401</Y>
        </CardAreas>
        <Gender>男</Gender>
        <Nationality>汉</Nationality>
        <Name>张三</Name>
        <IDNumber>310228199601115411</IDNumber>
        <BirthDate>19960111</BirthDate>
    </FrontResult>
</Data>
```


`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "D3F5BA69-79C4-46A4-B02B-58C4EEBC4C33",
  "Data" : {
    "FrontResult" : {
      "Address" : "浙江省杭州市余杭区余杭小区6栋602",
      "FaceRectVertices" : [ {
        "X" : 521.7265625,
        "Y" : 286.1796569824219
      }, {
        "X" : 424.7409362792969,
        "Y" : 282.30224609375
      }, {
        "X" : 429.46124267578125,
        "Y" : 164.23321533203125
      }, {
        "X" : 526.4468994140625,
        "Y" : 168.11062622070312
      } ],
      "FaceRectangle" : {
        "Center" : {
          "X" : 475.5939025878906,
          "Y" : 225.20643615722656
        },
        "Angle" : -87.71058654785156,
        "Size" : {
          "Height" : 97.06315612792969,
          "Width" : 118.16333770751953
        }
      },
      "CardAreas" : [ {
        "X" : 40,
        "Y" : 81
      }, {
        "X" : 588,
        "Y" : 80
      }, {
        "X" : 588,
        "Y" : 400
      }, {
        "X" : 40,
        "Y" : 401
      } ],
      "Gender" : "男",
      "Nationality" : "汉",
      "Name" : "张三",
      "IDNumber" : 310228199601115400,
      "BirthDate" : 19960111
    }
  }
}
```

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小时后会被系统自动清理删除。

3.2. 户口页识别

本文介绍户口页识别RecognizeAccountPage的语法及示例。

功能描述

户口页识别能力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

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：不超过3 MB。
- 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	系统规定参数。取值：**RecognizeAccountPage**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeAccountPage/hky1.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
RequestId	String	8D588F22-3E14-4674-9EAE-49AE1BA10EA2	请求ID。
Data	Object		返回的结果数据内容。
Gender	String	"Gender": "女",	性别，识别失败时返回 `FailInRecognition` 。
TitleArea	Object		标题区域位置，识别失败时返回 `FailInDetection` 。
Top	Integer	17	标题区域左上角y坐标。
Width	Integer	30	标题区域宽度。
Height	Integer	30	标题区域高度。
Left	Integer	205	标题区域左上角x坐标。
Angle	Float	0	图片顺时针旋转角度，枚举类型：0度、90度、180度、270度。
RegisterStampAreas	Array of RegisterStampArea		大圆形登记注册章位置列表（识别失败时， 则不显示该字段）。
Top	Integer	2215	登记注册章左上角y坐标。
Width	Integer	152	登记注册章宽度。
Height	Integer	138	登记注册章高度。
Left	Integer	330	登记注册章左上角x坐标。

名称	类型	示例值	描述
Nationality	String	彝族	民族，识别失败时返回 `FailInRecognition` 。
BirthPlace	String	四川省雷波县	出生地，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）。
Top	Integer	10	其他印章左上角y坐标。
Width	Integer	10	其他印章宽度。
Height	Integer	10	其他印章高度。
Left	Integer	10	其他印章左上角x坐标。
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）。
Top	Integer	10	承办人签章左上角y坐标。
Width	Integer	10	承办人签章宽度。
Height	Integer	10	承办人签章高度。
Left	Integer	10	承办人签章左上角x坐标。
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` 。
Relation	String	孙女	户主或与户主关系，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` 。

名称	类型	示例值	描述
Name	String	杨冬	姓名，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）。
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。
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。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeAccountPage
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeAccountPage/hky1.jpg
&<公共请求参数>
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<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

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "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。

3.3. 银行卡识别

本文介绍银行卡识别RecognizeBankCard的语法及示例。

功能描述

银行卡识别能力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

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：不超过3 MB。
- 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。

计费说明

关于银行卡识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口，如需免费体验调试请前往体验中心。

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	系统规定参数。取值：**RecognizeBankCard**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeBankCard/yhk3.jpg	图像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
RequestId	String	D9C7521-0367-42EE-9646-FD066CCADB26	请求ID。
Data	Object		返回的数据内容。
CardNumber	String	6212262315007683105	银行卡号。
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` 。

名称	类型	示例值	描述
BankName	String	中国工商银行	银行名称，识别失败则返回空字符串。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeBankCard
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeBank
Card/yhk3.jpg
&<公共请求参数>
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<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

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "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。

3.4. 名片识别

本文介绍名片识别RecognizeBusinessCard的语法及示例。

功能描述

名片识别能力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

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：不超过3 MB。
- 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。
- 名片完整且占据图像的60%以上，角度正向且文字清晰。
- 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	系统规定参数。取值：**RecognizeBusinessCard**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeBusinessCard/RecognizeBusinessCard1.jpg	图像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
RequestId	String	0068957A-C493-481F-BA14-C3F89BF75BD4	请求ID。

名称	类型	示例值	描述
Data	Object		返回的数据内容。
Companies	Array of String	美丽广告有限公司	名片上的公司名称，可能为空。
Titles	Array of String	设计经理	名片上的职位名称，可能为空。
Emails	Array of String	meili@123.com	名片上的邮箱地址，可能为空。
Departments	Array of String	设计部	名片上的部门名称，可能为空。
OfficePhoneNumbers	Array of String	02188888888	名片上的座机号码，可能为空。
Name	String	孙梦	名片上的姓名。
CellPhoneNumbers	Array of String	13900001234	名片上的手机号码，可能为空。
Addresses	Array of String	杭州市西湖区湖滨123号	名片上的地址，可能为空。

示例

请求示例

```
http(s)://[Endpoint]/?Action=RecognizeBusinessCard
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeBusi
nessCard/RecognizeBusinessCard1.jpg
&<公共请求参数>
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeBusinessCardResponse>
    <RequestId>0068957A-C493-481F-BA14-C3F89BF75BD4</RequestId>
    <Data>
        <Companies>美丽广告有限公司</Companies>
        <Titles>设计经理</Titles>
        <Emails>meili@123.com</Emails>
        <Departments>设计部</Departments>
        <OfficePhoneNumbers>02188888888</OfficePhoneNumbers>
        <Name>孙梦</Name>
        <CellPhoneNumbers>13900001234</CellPhoneNumbers>
        <Addresses>杭州市西湖区湖滨123号</Addresses>
    </Data>
</RecognizeBusinessCardResponse>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "0068957A-C493-481F-BA14-C3F89BF75BD4",
  "Data" : {
    "Companies" : [ "美丽广告有限公司" ],
    "Titles" : [ "设计经理" ],
    "Emails" : [ "meili@123.com" ],
    "Departments" : [ "设计部" ],
    "OfficePhoneNumbers" : [ "02188888888" ],
    "Name" : "孙梦",
    "CellPhoneNumbers" : [ "13900001234" ],
    "Addresses" : [ "杭州市西湖区湖滨123号" ]
  }
}
```

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。

3.5. 二维码识别

本文介绍二维码识别RecognizeQrCode的语法及示例。

功能描述

二维码识别能力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

- 图像格式：PNG、JPG、JPEG、BMP、GIF、WEBP。
- 图像大小：不超过10 MB。如您有大图需求，请提工单申请。
- 图像分辨率：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。

计费说明

关于二维码识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口，如需免费体验调试请前往体验中心。

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	系统规定参数。取值：**RecognizeQrCode**。
Tasks.N.ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeQrCode/RecognizeQrCode6.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
RequestId	String	A53DC437-F883-4968-86D5-EB21FB044692	请求ID。
Data	Object		返回的结果数据内容。
Elements	Array of Element		识别返回的元素信息。
ImageURL	String	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeQrCode/RecognizeQrCode6.jpg	对应请求中的图像URL地址。
TaskId	String	img5iGtwVIxQzc4Nqy$L84yHd-1v****	该检测任务的ID。
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。
Suggestion	String	review	建议您执行的操作，取值包括： • 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）。
QrCodesData	Array of String	https://yqh.aliyun.com/live/detail/23798	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）。
Label	String	qrcode	检测结果的分类，取值包括： • normal：正常图片。 • qrcode：含二维码的图片。 • programCode：含小程序码的图片。 **说明** 默认不识别小程序码。如果需要识别，请提工单申请。

名称	类型	示例值	描述
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。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeQrCode
&Tasks=[{"ImageURL":"http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/Re
cognizeQrCode/RecognizeQrCode6.jpg"}]
&公共请求参数
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeQrCodeResponse>
    <RequestId>A53DC437-F883-4968-86D5-EB21FB044692</RequestId>
    <Data>
        <Elements>
            <ImageURL>http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/R
ecognizeQrCode/RecognizeQrCode6.jpg</ImageURL>
            <TaskId>img5iGtwVIxQzc4Nqy$L84yHd-1v****</TaskId>
            <Results>
                <Suggestion>review</Suggestion>
                <QrCodesData>https://yqh.aliyun.com/live/detail/23798</QrCodesData>
                <Label>qrcode</Label>
                <Rate>99.91</Rate>
            </Results>
        </Elements>
    </Data>
</RecognizeQrCodeResponse>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "A53DC437-F883-4968-86D5-EB21FB044692",
  "Data" : {
    "Elements" : [ {
      "ImageURL" : "http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/Rec
ognizeQrCode/RecognizeQrCode6.jpg",
      "TaskId" : "img5iGtwVIxQzc4Nqy$L84yHd-1v****",
      "Results" : [ {
        "Suggestion" : "review",
        "QrCodesData" : [ "https://yqh.aliyun.com/live/detail/23798" ],
        "Label" : "qrcode",
        "Rate" : 99.91
      } ]
    } ]
  }
}
```

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。

3.6. 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

中国护照识别能力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

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：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	RecognizeChinapassport	系统规定参数。取值：**RecognizeChinapassport**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeChinapassport/hz1.jpg	图像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
RequestId	String	1DCA8C4A-DF5A-4402-8220-836C5DFC4221	请求ID。
Data	Object		返回的结果数据内容。
Type	String	PO	护照类型。
Sex	String	F	性别。 • M：male（男性） • F：female（女性）
Authority	String	公安部出入境管理局	签发机关。
IssuePlaceRaw	String	上海/SHANGHAI	发证地址带拼音。
Success	Boolean	true	识别是否成功。
LineOne	String	POCHNLI<<SI<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<	MRZ码第二行。

名称	类型	示例值	描述
ExpiryDate	String	20231018	到期日期。
BirthPlace	String	上海	出生地。
PassportNo	String	E20354xxxx	护照号码。
BirthPlaceRaw	String	上海/SHANGHAI	出生地带拼音。
IssueDate	String	20131019	发证日期。
SourceCountry	String	CHN	国籍。
BirthDate	String	19880820	出生日期。
NameChineseRaw	String	李四LISI	中文姓名包含拼音。
IssuePlace	String	上海	发证地址。
NameChinese	String	李四	中文姓名。
LineZero	String	E203545580CHN8610304M2305019MNPELOLIOKLPA938	MRZ码第一行。
Country	String	CHN	国籍。
BirthDay	String	880820	生日。
ExpiryDay	String	231018	到期日期。
Name	String	LI.SI	英文姓名。
PersonId	String	MNPELOLIOKLPA9	护照人身份ID。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeChinapassport
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeChin
apassport/hz1.jpg
&<公共请求参数>
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeChinapassportResponse>
    <RequestId>1DCA8C4A-DF5A-4402-8220-836C5DFC4221</RequestId>
    <Data>
        <Type>P0</Type>
        <Sex>F</Sex>
        <Authority>公安部出入境管理局</Authority>
        <IssuePlaceRaw>上海/SHANGHAI</IssuePlaceRaw>
        <Success>true</Success>
        <LineOne>POCHNLI&lt;&lt;SI&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&
lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;</LineOne>
        <ExpiryDate>20231018</ExpiryDate>
        <BirthPlace>上海</BirthPlace>
        <PassportNo>E20354xxxx</PassportNo>
        <BirthPlaceRaw>上海/SHANGHAI</BirthPlaceRaw>
        <IssueDate>20131019</IssueDate>
        <SourceCountry>CHN</SourceCountry>
        <BirthDate>19880820</BirthDate>
        <NameChineseRaw>李四LISI</NameChineseRaw>
        <IssuePlace>上海</IssuePlace>
        <NameChinese>李四</NameChinese>
        <LineZero>E203545580CHN8610304M2305019MNPELOLIOKLPA938</LineZero>
        <Country>CHN</Country>
        <BirthDay>880820</BirthDay>
        <ExpiryDay>231018</ExpiryDay>
        <Name>LI.SI</Name>
        <PersonId>MNPELOLIOKLPA9</PersonId>
    </Data>
</RecognizeChinapassportResponse>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "1DCA8C4A-DF5A-4402-8220-836C5DFC4221",
  "Data" : {
    "Type" : "P0",
    "Sex" : "F",
    "Authority" : "公安部出入境管理局",
    "IssuePlaceRaw" : "上海/SHANGHAI",
    "Success" : true,
    "LineOne" : "POCHNLI<<SI<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<",
    "ExpiryDate" : "20231018",
    "BirthPlace" : "上海",
    "PassportNo" : "E20354xxxx",
    "BirthPlaceRaw" : "上海/SHANGHAI",
    "IssueDate" : "20131019",
    "SourceCountry" : "CHN",
    "BirthDate" : "19880820",
    "NameChineseRaw" : "李四LISI",
    "IssuePlace" : "上海",
    "NameChinese" : "李四",
    "LineZero" : "E203545580CHN8610304M2305019MNPELOLIOKLPA938",
    "Country" : "CHN",
    "BirthDay" : "880820",
    "ExpiryDay" : "231018",
    "Name" : "LI.SI",
    "PersonId" : "MNPELOLIOKLPA9"
  }
}
```

错误码

关于中国护照识别的错误码，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。

3.7. 护照MRZ码识别

本文介绍护照MRZ码识别RecognizePassportMRZ的语法及示例。

功能描述

护照MRZ码识别能力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

- 图像格式：JPEG、PNG、JPG。
- 图像大小：不超过10 MB。
- 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	系统规定参数。取值：**RecognizePassportMRZ**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeChinapassport/hz2.jpg	图像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
RequestId	String	DC6370E9-4BE5-56A7-9BD9-E1BF538CDD23	请求ID。
Data	Object		返回的数据内容。
Regions	Array of Region		输出识别MRZ码结果。
RecognitionScore	Float	0.9916993379592896	识别结果置信度。
DetectionScore	Float	0.59131116	识别位置框置信度。
Name	String	mrz1	护照字段名称，具体请参见如下护照字段含义表格。

名称	类型	示例值	描述
Content	String	POCHNZHANG<<SAN<<<<<<<<<<<<<<<<<<<<<<<<<<<<<	识别内容。
BandBoxes	Array of Float	[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。

护照字段含义如下。

归一化字段	中文含义
type	类型
country_code	国家码
passport_no	护照号
name	姓名
surname	姓
given_name	名
nationality	国籍
birth	生日
sex	性别
date_of_issue	开始日期
date_of_expiry	结束日期
mrz1	机器校验码1

归一化字段	中文含义
mrz2	机器校验码2

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizePassportMRZ
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeChin
apassport/hz2.jpg
&公共请求参数
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizePassportMRZResponse>
    <RequestId>DC6370E9-4BE5-56A7-9BD9-E1BF538CDD23</RequestId>
    <Data>
        <Regions>
            <Content>ZHANG SAN</Content>
            <Name>name</Name>
        </Regions>
        <Regions>
            <Content>E080808081</Content>
            <Name>passport_no</Name>
        </Regions>
        <Regions>
            <Content>M</Content>
            <Name>sex</Name>
        </Regions>
        <Regions>
            <Content>CHN</Content>
            <Name>country_code</Name>
        </Regions>
        <Regions>
            <Content>E080808081CHN9201129M24091187766ZXCOWE&lt;&lt;&lt;&lt;90</Content>
            <RecognitionScore>0.9890465140342712</RecognitionScore>
            <BandBoxes>123</BandBoxes>
            <BandBoxes>454</BandBoxes>
            <BandBoxes>822</BandBoxes>
            <BandBoxes>473</BandBoxes>
            <Name>mrz2</Name>
            <DetectionScore>0.5469832</DetectionScore>
        </Regions>
        <Regions>
            <Content>POCHNZHANG&lt;&lt;SAN&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&
lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;</Content>
            <RecognitionScore>0.9894542098045349</RecognitionScore>
            <BandBoxes>138</BandBoxes>
            <BandBoxes>422</BandBoxes>
            <BandBoxes>812</BandBoxes>
            <BandBoxes>444</BandBoxes>
            <Name>mrz1</Name>
            <DetectionScore>0.51930517</DetectionScore>
        </Regions>
        <Regions>
            <Content>PO</Content>
            <Name>type</Name>
        </Regions>
    </Data>
</RecognizePassportMRZResponse>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "DC6370E9-4BE5-56A7-9BD9-E1BF538CDD23",
  "Data" : {
    "Regions" : [ {
      "Content" : "ZHANG SAN",
      "Name" : "name"
    }, {
      "Content" : "E080808081",
      "Name" : "passport_no"
    }, {
      "Content" : "M",
      "Name" : "sex"
    }, {
      "Content" : "CHN",
      "Name" : "country_code"
    }, {
      "Content" : "E080808081CHN9201129M24091187766ZXCOWE<<<<90",
      "RecognitionScore" : 0.9890465140342712,
      "BandBoxes" : [ 123, 454, 822, 473 ],
      "Name" : "mrz2",
      "DetectionScore" : 0.5469832
    }, {
      "Content" : "POCHNZHANG<<SAN<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<",
      "RecognitionScore" : 0.9894542098045349,
      "BandBoxes" : [ 138, 422, 812, 444 ],
      "Name" : "mrz1",
      "DetectionScore" : 0.51930517
    }, {
      "Content" : "PO",
      "Name" : "type"
    } ]
  }
}
```

错误码

关于护照MRZ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。

4.资产类证件识别

4.1. 营业执照识别

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

功能描述

营业执照识别能力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、统一社会信用代码、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

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：不超过3 MB。
- 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。
- 请求格式：JPEG、JPG、PNG、BMP、GIF。

计费说明

关于营业执照识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口，如需免费体验调试请前往体验中心。

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	系统规定参数。取值：**RecognizeBusinessLicense**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeBusinessLicense/RecognizeBusinessLicense1.jpg	图像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
RequestId	String	F34D031B-02BD-4A59-BA35-EE068DD6F6E6	请求ID。
Data	Object		返回的结果数据内容。
Type	String	有限责任公司	公司类型。识别失败时返回 `FailInRecognition` 。
Stamp	Object		印章位置。识别失败时返回 `FailInDetection` 。
Top	Integer	1030	区域左上角y坐标。
Width	Integer	154	区域宽度。
Height	Integer	154	区域高度。
Left	Integer	650	区域左上角x坐标。
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”。
ValidPeriod	String	29991231	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”。 **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”。
Business	String	网络技术服务；网站建设；销售：I类医疗器械、保健用品（不含保健食品）等	经营范围。识别失败时返回 `FailInRecognition` 。

名称	类型	示例值	描述
Angle	Float	0	输入图片的角度（顺时针旋转）。枚举值包括：0、90、180、270。
RegisterNumber	String	91500108320423****	统一社会信用代码。识别失败时返回 `FailInRecognition` 。
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 `FailInRecognition` 。
Capital	String	壹百万元整	注册资本。识别失败时返回 `FailInRecognition` 。
Title	Object		标题位置。识别失败时返回 `FailInDetection` 。
Top	Integer	10	区域左上角y坐标。
Width	Integer	10	区域宽度。
Height	Integer	10	区域高度。
Left	Integer	10	区域左上角x坐标。
Emblem	Object		国徽位置。识别失败时返回 `FailInDetection` 。
Top	Integer	8	区域左上角y坐标。
Width	Integer	162	区域宽度。
Height	Integer	163	区域高度。
Left	Integer	366	区域左上角x坐标。
Name	String	某某电子商务有限公司	公司名称。识别失败时返回 `FailInRecognition` 。

名称	类型	示例值	描述
QRCode	Object		二维码位置。识别失败时返回 `FailInDetection` 。
Top	Integer	914	区域左上角y坐标。
Width	Integer	126	区域宽度。
Height	Integer	132	区域高度。
Left	Integer	156	区域左上角x坐标。
LegalPerson	String	李四	公司法人。识别失败时返回 `FailInRecognition` 。

示例

请求示例

```
http(s)://[Endpoint]/?Action=RecognizeBusinessLicense
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeBusi
nessLicense/RecognizeBusinessLicense1.jpg
&公共请求参数
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RequestId>F34D031B-02BD-4A59-BA35-EE068DD6F6E6</RequestId>
<Data>
    <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>
    <Stamp>
        <Left>971</Left>
        <Top>572</Top>
        <Height>197</Height>
        <Width>191</Width>
    </Stamp>
    <Title>
        <Left>447</Left>
        <Top>177</Top>
        <Height>80</Height>
        <Width>354</Width>
    </Title>
    <Capital>叁佰万元整</Capital>
    <LegalPerson>李四</LegalPerson>
    <EstablishDate>20150504</EstablishDate>
    <Name>某某电子商务有限公司</Name>
    <RegisterNumber>91500108320423****</RegisterNumber>
    <Type>有限责任公司</Type>
    <QRCode>
        <Left>848</Left>
        <Top>199</Top>
        <Height>88</Height>
        <Width>91</Width>
    </QRCode>
    <Angle>0</Angle>
    <Emblem>
        <Left>550</Left>
        <Top>6</Top>
        <Height>150</Height>
        <Width>137</Width>
    </Emblem>
    <Business>网络技术服务;网站建设;销售:I类医疗器械、保健用品(不含保健食品);医药技术咨询(不含诊疗活动及药品销售)、健康知识及其信息咨询(不含诊疗活动)、健康管理咨询(不含诊疗活动)、医院管理咨询(不含诊疗活动及药品销售):健康保健项目策划、设计;化妆品、饰、工艺礼品(不含象牙及其制品)、家具护理用品、劳防用品的批发;货物进出口(法律、法规禁止的项目除外;法律、法规限制的项目取得许可后方可经营);品牌推广;旅游信息咨询;翻译服务;票务代理;设计、制作、代理、发布广告;会议及展览展示服务。(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

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "F34D031B-02BD-4A59-BA35-EE068DD6F6E6",
  "Data" : {
    "ValidPeriod" : 29991231,
    "Address" : "浙江省杭州市西湖区转塘科技经济区块888号888幢",
    "Stamp" : {
      "Left" : 971,
      "Top" : 572,
      "Height" : 197,
      "Width" : 191
    },
    "Title" : {
      "Left" : 447,
      "Top" : 177,
      "Height" : 80,
      "Width" : 354
    },
    "Capital" : "叁佰万元整",
    "LegalPerson" : "李四",
    "EstablishDate" : 20150504,
    "Name" : "某某电子商务有限公司",
    "RegisterNumber" : "91500108320423****",
    "Type" : "有限责任公司",
    "QRCode" : {
      "Left" : 848,
      "Top" : 199,
      "Height" : 88,
      "Width" : 91
    },
    "Angle" : 0,
    "Emblem" : {
      "Left" : 550,
      "Top" : 6,
      "Height" : 150,
      "Width" : 137
    },
    "Business" : "网络技术服务;网站建设;销售:I类医疗器械、保健用品(不含保健食品);医药技术咨询(不含诊疗活动及药品销售)、健康知识及其信息咨询(不含诊疗活动)、健康管理咨询(不含诊疗活动)、医院管理咨询(不含诊疗活动及药品销售):健康保健项目策划、设计;化妆品、饰、工艺礼品(不含象牙及其制品)、家具护理用品、劳防用品的批发;货物进出口(法律、法规禁止的项目除外;法律、法规限制的项目取得许可后方可经营);品牌推广;旅游信息咨询;翻译服务;票务代理;设计、制作、代理、发布广告;会议及展览展示服务。(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。

4.2. 公章识别

本文介绍公章识别RecognizeStamp的语法及示例。

功能描述

公章识别能力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。

输入限制

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：不超过3 MB。
- 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	系统规定参数。取值：**RecognizeStamp**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeStamp/RecognizeStamp1.jpg	图像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

名称	类型	示例值	描述
RequestId	String	D6C24839-91A7-41DA-B31F-98F08EF80CC0	请求ID。
Data	Object		返回的结果数据内容。
Results	Array of Results		印章列表详细信息。
Text	Object		印章具体文字。
Confidence	Float	0.9978633522987366	印章文字概率，范围 0~1。
Content	String	杭州市科技有限公司	印章文字内容。
Roi	Object		印章文字区域。
Top	Integer	187	文字区域左上角y坐标。
Width	Integer	262	文字区域宽度。
Height	Integer	264	文字区域高度。
Left	Integer	189	文字区域左上角x坐标。
GeneralText	Array of GeneralText		印章其他区域文字列表。
Confidence	Float	0.9650014638900757	其他区域文字概率，范围 0~1。
Content	String	合同专用章	其他区域文字内容。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeStamp
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeStamp/RecognizeStamp1.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeStampResponse>
    <RequestId>D6C24839-91A7-41DA-B31F-98F08EF80CC0</RequestId>
    <Data>
        <Results>
            <Text>
                <Confidence>0.99786335</Confidence>
                <Content>杭州市科技有限公司</Content>
            </Text>
            <Roi>
                <Top>187</Top>
                <Width>262</Width>
                <Height>264</Height>
                <Left>189</Left>
            </Roi>
            <GeneralText>
                <Confidence>0.96500146</Confidence>
                <Content>合同专用章</Content>
            </GeneralText>
        </Results>
    </Data>
</RecognizeStampResponse>
```

JSON 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "D6C24839-91A7-41DA-B31F-98F08EF80CC0",
  "Data" : {
    "Results" : [ {
      "Text" : {
        "Confidence" : 0.99786335,
        "Content" : "杭州市科技有限公司"
      },
      "Roi" : {
        "Top" : 187,
        "Width" : 262,
        "Height" : 264,
        "Left" : 189
      },
      "GeneralText" : [ {
        "Confidence" : 0.96500146,
        "Content" : "合同专用章"
      } ]
    } ]
  }
}
```

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。

5.通用文字类识别

5.1. 通用文字识别

本文介绍通用文字识别RecognizeCharacter的语法及示例。

功能描述

通用文字识别能力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

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：不超过3 MB。
- 图像分辨率：大于15×15像素，小于4096×4096像素。
- URL地址中不能包含中文字符。

计费说明

关于通用文字识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口，如需免费体验调试请前往体验中心。

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	系统规定参数。取值：**RecognizeCharacter**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeCharacter/RecognizeCharacter5.jpg	图像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	图像中文字的最小高度，单位像素。

名称	类型	是否必选	示例值	描述
OutputProbability	Boolean	是	true	是否输出文字框的概率。取值： • true：输出文字框的概率。 • false：不输出文字框的概率。

返回数据

名称	类型	示例值	描述
RequestId	String	7A9BC7FE-2D42-57AF-93BC-09A229DD2F1D	请求ID。
Data	Object		返回的结果数据内容。
Results	Array of Result		返回识别信息。
TextRectangles	Object		文字框区域位置。
Top	Integer	150	文字区域左上角y坐标。
Width	Integer	77	文字区域宽度。
Height	Integer	409	文字区域高度。
Angle	Integer	-65	文字区域角度，角度范围 `[-180, 180]` 。 **说明** 以文字区域中心点为旋转点，向右旋转角度为正，向左旋转角度为负。
Left	Integer	511	文字区域左上角x坐标。
Text	String	祝你生日快乐	文字内容。
Probability	Float	0.99	文字内容的概率，取值范围为0~1。

示例

请求示例

```
http(s)://[Endpoint]/?Action=RecognizeCharacter
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeChar
acter/RecognizeCharacter5.jpg
&MinHeight=10
&OutputProbability=true
&公共请求参数
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeCharacterResponse>
    <RequestId>7A9BC7FE-2D42-57AF-93BC-09A229DD2F1D</RequestId>
    <Data>
        <Results>
            <TextRectangles>
                <Left>599</Left>
                <Top>160</Top>
                <Angle>-69</Angle>
                <Height>107</Height>
                <Width>26</Width>
            </TextRectangles>
            <Probability>0.99</Probability>
            <Text>HAPPY</Text>
        </Results>
        <Results>
            <TextRectangles>
                <Left>576</Left>
                <Top>150</Top>
                <Angle>-63</Angle>
                <Height>200</Height>
                <Width>37</Width>
            </TextRectangles>
            <Probability>0.99</Probability>
            <Text>birthday</Text>
        </Results>
        <Results>
            <TextRectangles>
                <Left>511</Left>
                <Top>150</Top>
                <Angle>-65</Angle>
                <Height>409</Height>
                <Width>77</Width>
            </TextRectangles>
            <Probability>0.99</Probability>
            <Text>祝你生日快乐</Text>
        </Results>
    </Data>
</RecognizeCharacterResponse>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "7A9BC7FE-2D42-57AF-93BC-09A229DD2F1D",
  "Data" : {
    "Results" : [ {
      "TextRectangles" : {
        "Left" : 599,
        "Top" : 160,
        "Angle" : -69,
        "Height" : 107,
        "Width" : 26
      },
      "Probability" : 0.99,
      "Text" : "HAPPY"
    }, {
      "TextRectangles" : {
        "Left" : 576,
        "Top" : 150,
        "Angle" : -63,
        "Height" : 200,
        "Width" : 37
      },
      "Probability" : 0.99,
      "Text" : "birthday"
    }, {
      "TextRectangles" : {
        "Left" : 511,
        "Top" : 150,
        "Angle" : -65,
        "Height" : 409,
        "Width" : 77
      },
      "Probability" : 0.99,
      "Text" : "祝你生日快乐"
    } ]
  }
}
```

错误码

关于通用文字识别的错误码，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. 表格识别

本文介绍表格识别RecognizeTable的语法及示例。

功能描述

表格识别能力可以自动识别表格位置、表格内容信息。适用于边框线条为黑色、横竖框线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

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：不超过3 MB。
- 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。

计费说明

关于表格识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口，如需免费体验调试请前往体验中心。

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	系统规定参数。取值：**RecognizeTable**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeTable/RecognizeTable4.jpg	图像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。

名称	类型	是否必选	示例值	描述
UseFinanceModel	Boolean	是	true	是否使用财务报表模型。 • true：财务报表模型 • false：非财务报表模型
AssureDirection	Boolean	是	false	图像方向是否确定为正向。 • true：正向 • false：反向
HasLine	Boolean	是	false	是否无线条。 • true：无线条或者只有横线没有竖线 • false：有线条
SkipDetection	Boolean	是	false	是否跳过检测。 • true：跳过检查 • false：不跳过检查

返回数据

名称	类型	示例值	描述
RequestId	String	CBC36BE6-2A18-5256-82BD-8B5477E5D058	请求ID。
Data	Object		返回的结果数据内容。
FileContent	String	UEsDBBQAAAAIAAAAIQBukMk4WAIAA****	HTML字符串。仅在请求参数OutputFormat为 `html` 或 `xlsx` 时显示。
Tables	Array of Table		JSON格式的数据。仅在请求参数OutputFormat为 `json` 时显示。
Head	Array of String	存活盘点表	表头信息。
Tail	Array of String	职工券	表尾信息。
TableRows	Array of TableRow		JSON格式的表格数据。

名称	类型	示例值	描述
TableColumns	Array of TableColumn		JSON格式的表格数据。
EndRow	Integer	1	单元格所占的行数（rowspan）为ey-sy。
EndColumn	Integer	4	单元格所占的列数（colspan）为ex-sx。
Width	Integer	0	图片上单元格的宽度。
Height	Integer	0	图片上单元格的高度。
Texts	Array of String	序号	文本，每行（row）文字为一个block。
StartRow	Integer	0	单元格的起始行ID。
StartColumn	Integer	1	单元格的起始列ID。

示例

请求示例

```
http(s)://[Endpoint]/?Action=RecognizeTable
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeTabl
e/RecognizeTable4.jpg
&OutputFormat=json
&UseFinanceModel=true
&AssureDirection=false
&HasLine=false
&SkipDetection=false
&公共请求参数
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeTableResponse>
    <RequestId>CBC36BE6-2A18-5256-82BD-8B5477E5D058</RequestId>
    <Data>
        <Tables>
            <Head>存活盘点表</Head>
            <TableRows>
                <TableColumns>
                    <EndRow>1</EndRow>
```

```
            <EndColumn>1</EndColumn>
            <Texts>序号</Texts>
            <Height>0</Height>
            <Width>0</Width>
            <StartRow>0</StartRow>
            <StartColumn>0</StartColumn>
        </TableColumns>
        <TableColumns>
            <EndRow>1</EndRow>
            <EndColumn>2</EndColumn>
            <Texts>存货名称</Texts>
            <Height>0</Height>
            <Width>0</Width>
            <StartRow>0</StartRow>
            <StartColumn>1</StartColumn>
        </TableColumns>
        <TableColumns>
            <EndRow>1</EndRow>
            <EndColumn>3</EndColumn>
            <Texts>计量单位</Texts>
            <Height>0</Height>
            <Width>0</Width>
            <StartRow>0</StartRow>
            <StartColumn>2</StartColumn>
        </TableColumns>
        <TableColumns>
            <EndRow>1</EndRow>
            <EndColumn>4</EndColumn>
            <Texts>数量</Texts>
            <Height>0</Height>
            <Width>0</Width>
            <StartRow>0</StartRow>
            <StartColumn>3</StartColumn>
        </TableColumns>
        <TableColumns>
            <EndRow>1</EndRow>
            <EndColumn>5</EndColumn>
            <Texts>单价</Texts>
            <Height>0</Height>
            <Width>0</Width>
            <StartRow>0</StartRow>
            <StartColumn>4</StartColumn>
        </TableColumns>
        <TableColumns>
            <EndRow>1</EndRow>
            <EndColumn>6</EndColumn>
            <Texts>账面金额</Texts>
            <Height>0</Height>
            <Width>0</Width>
            <StartRow>0</StartRow>
            <StartColumn>5</StartColumn>
        </TableColumns>
        <TableColumns>
            <EndRow>1</EndRow>
```

```
                <EndColumn>7</EndColumn>
                <Texts>备注</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>0</StartRow>
                <StartColumn>6</StartColumn>
            </TableColumns>
        </TableRows>
        <TableRows>
            <TableColumns>
                <EndRow>2</EndRow>
                <EndColumn>1</EndColumn>
                <Texts>1</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>1</StartRow>
                <StartColumn>0</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>2</EndRow>
                <EndColumn>2</EndColumn>
                <Texts>发电机</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>1</StartRow>
                <StartColumn>1</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>2</EndRow>
                <EndColumn>3</EndColumn>
                <Texts>台</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>1</StartRow>
                <StartColumn>2</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>2</EndRow>
                <EndColumn>4</EndColumn>
                <Texts>2</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>1</StartRow>
                <StartColumn>3</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>2</EndRow>
                <EndColumn>5</EndColumn>
                <Texts>499,222,12</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>1</StartRow>
                <StartColumn>4</StartColumn>
            </TableColumns>
```

```
            <TableColumns>
                <EndRow>2</EndRow>
                <EndColumn>6</EndColumn>
                <Texts>99844424</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>1</StartRow>
                <StartColumn>5</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>2</EndRow>
                <EndColumn>7</EndColumn>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>1</StartRow>
                <StartColumn>6</StartColumn>
            </TableColumns>
        </TableRows>
        <TableRows>
            <TableColumns>
                <EndRow>3</EndRow>
                <EndColumn>1</EndColumn>
                <Texts>2</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>2</StartRow>
                <StartColumn>0</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>3</EndRow>
                <EndColumn>2</EndColumn>
                <Texts>变压器</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>2</StartRow>
                <StartColumn>1</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>3</EndRow>
                <EndColumn>3</EndColumn>
                <Texts>台</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>2</StartRow>
                <StartColumn>2</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>3</EndRow>
                <EndColumn>4</EndColumn>
                <Texts>1</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>2</StartRow>
                <StartColumn>3</StartColumn>
            </TableColumns>
```

```
            </TableColumns>
            <TableColumns>
                <EndRow>3</EndRow>
                <EndColumn>5</EndColumn>
                <Texts>205,121,18</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>2</StartRow>
                <StartColumn>4</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>3</EndRow>
                <EndColumn>6</EndColumn>
                <Texts>205,121,18</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>2</StartRow>
                <StartColumn>5</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>3</EndRow>
                <EndColumn>7</EndColumn>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>2</StartRow>
                <StartColumn>6</StartColumn>
            </TableColumns>
        </TableRows>
        <TableRows>
            <TableColumns>
                <EndRow>4</EndRow>
                <EndColumn>1</EndColumn>
                <Texts>3</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>3</StartRow>
                <StartColumn>0</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>4</EndRow>
                <EndColumn>2</EndColumn>
                <Texts>豆浆生产线</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>3</StartRow>
                <StartColumn>1</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>4</EndRow>
                <EndColumn>3</EndColumn>
                <Texts>台</Texts>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>3</StartRow>
                <StartColumn>2</StartColumn>
```

```
                </TableColumns>
                <TableColumns>
                    <EndRow>4</EndRow>
                    <EndColumn>4</EndColumn>
                    <Texts>1</Texts>
                    <Height>0</Height>
                    <Width>0</Width>
                    <StartRow>3</StartRow>
                    <StartColumn>3</StartColumn>
                </TableColumns>
                <TableColumns>
                    <EndRow>4</EndRow>
                    <EndColumn>5</EndColumn>
                    <Texts>123,987,23</Texts>
                    <Height>0</Height>
                    <Width>0</Width>
                    <StartRow>3</StartRow>
                    <StartColumn>4</StartColumn>
                </TableColumns>
                <TableColumns>
                    <EndRow>4</EndRow>
                    <EndColumn>6</EndColumn>
                    <Texts>123,987,23</Texts>
                    <Height>0</Height>
                    <Width>0</Width>
                    <StartRow>3</StartRow>
                    <StartColumn>5</StartColumn>
                </TableColumns>
                <TableColumns>
                    <EndRow>4</EndRow>
                    <EndColumn>7</EndColumn>
                    <Height>0</Height>
                    <Width>0</Width>
                    <StartRow>3</StartRow>
                    <StartColumn>6</StartColumn>
                </TableColumns>
            </TableRows>
            <TableRows>
                <TableColumns>
                    <EndRow>5</EndRow>
                    <EndColumn>1</EndColumn>
                    <Texts>4</Texts>
                    <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>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>4</StartRow>
                <StartColumn>2</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>5</EndRow>
                <EndColumn>4</EndColumn>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>4</StartRow>
                <StartColumn>3</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>5</EndRow>
                <EndColumn>5</EndColumn>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>4</StartRow>
                <StartColumn>4</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>5</EndRow>
                <EndColumn>6</EndColumn>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>4</StartRow>
                <StartColumn>5</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>5</EndRow>
                <EndColumn>7</EndColumn>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>4</StartRow>
                <StartColumn>6</StartColumn>
            </TableColumns>
        </TableRows>
        <TableRows>
            <TableColumns>
                <EndRow>6</EndRow>
                <EndColumn>1</EndColumn>
                <Texts>5</Texts>
                <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>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>5</StartRow>
                <StartColumn>2</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>6</EndRow>
                <EndColumn>4</EndColumn>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>5</StartRow>
                <StartColumn>3</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>6</EndRow>
                <EndColumn>5</EndColumn>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>5</StartRow>
                <StartColumn>4</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>6</EndRow>
                <EndColumn>6</EndColumn>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>5</StartRow>
                <StartColumn>5</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>6</EndRow>
                <EndColumn>7</EndColumn>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>5</StartRow>
                <StartColumn>6</StartColumn>
            </TableColumns>
        </TableRows>
        <TableRows>
            <TableColumns>
                <EndRow>7</EndRow>
                <EndColumn>1</EndColumn>
                <Texts>6</Texts>
                <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>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>6</StartRow>
                <StartColumn>2</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>7</EndRow>
                <EndColumn>4</EndColumn>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>6</StartRow>
                <StartColumn>3</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>7</EndRow>
                <EndColumn>5</EndColumn>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>6</StartRow>
                <StartColumn>4</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>7</EndRow>
                <EndColumn>6</EndColumn>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>6</StartRow>
                <StartColumn>5</StartColumn>
            </TableColumns>
            <TableColumns>
                <EndRow>7</EndRow>
                <EndColumn>7</EndColumn>
                <Height>0</Height>
                <Width>0</Width>
                <StartRow>6</StartRow>
                <StartColumn>6</StartColumn>
            </TableColumns>
        </TableRows>
        <TableRows>
            <TableColumns>
                <EndRow>8</EndRow>
```

```
            <EndRow>8</EndRow>
            <EndColumn>1</EndColumn>
            <Texts>7</Texts>
            <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>
            <Height>0</Height>
            <Width>0</Width>
            <StartRow>7</StartRow>
            <StartColumn>2</StartColumn>
        </TableColumns>
        <TableColumns>
            <EndRow>8</EndRow>
            <EndColumn>4</EndColumn>
            <Height>0</Height>
            <Width>0</Width>
            <StartRow>7</StartRow>
            <StartColumn>3</StartColumn>
        </TableColumns>
        <TableColumns>
            <EndRow>8</EndRow>
            <EndColumn>5</EndColumn>
            <Height>0</Height>
            <Width>0</Width>
            <StartRow>7</StartRow>
            <StartColumn>4</StartColumn>
        </TableColumns>
        <TableColumns>
            <EndRow>8</EndRow>
            <EndColumn>6</EndColumn>
            <Height>0</Height>
            <Width>0</Width>
            <StartRow>7</StartRow>
            <StartColumn>5</StartColumn>
        </TableColumns>
        <TableColumns>
            <EndRow>8</EndRow>
            <EndColumn>7</EndColumn>
            <Height>0</Height>
            <Width>0</Width>
            <StartRow>7</StartRow>
            <StartColumn>6</StartColumn>
```

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

JSON 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "CBC36BE6-2A18-5256-82BD-8B5477E5D058",
  "Data" : {
    "Tables" : {
      "Head" : "存活盘点表",
      "TableRows" : [ {
        "TableColumns" : [ {
          "EndRow" : 1,
          "EndColumn" : 1,
          "Texts" : "序号",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 0,
          "StartColumn" : 0
        }, {
          "EndRow" : 1,
          "EndColumn" : 2,
          "Texts" : "存货名称",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 0,
          "StartColumn" : 1
        }, {
          "EndRow" : 1,
          "EndColumn" : 3,
          "Texts" : "计量单位",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 0,
          "StartColumn" : 2
        }, {
          "EndRow" : 1,
          "EndColumn" : 4,
          "Texts" : "数量",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 0,
          "StartColumn" : 3
        }, {
          "EndRow" : 1,
          "EndColumn" : 5,
          "Texts" : "单价",
          "Height" : 0,
```

```
          "Width" : 0,
          "StartRow" : 0,
          "StartColumn" : 4
        }, {
          "EndRow" : 1,
          "EndColumn" : 6,
          "Texts" : "账面金额",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 0,
          "StartColumn" : 5
        }, {
          "EndRow" : 1,
          "EndColumn" : 7,
          "Texts" : "备注",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 0,
          "StartColumn" : 6
        } ]
      }, {
        "TableColumns" : [ {
          "EndRow" : 2,
          "EndColumn" : 1,
          "Texts" : 1,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 1,
          "StartColumn" : 0
        }, {
          "EndRow" : 2,
          "EndColumn" : 2,
          "Texts" : "发电机",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 1,
          "StartColumn" : 1
        }, {
          "EndRow" : 2,
          "EndColumn" : 3,
          "Texts" : "台",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 1,
          "StartColumn" : 2
        }, {
          "EndRow" : 2,
          "EndColumn" : 4,
          "Texts" : 2,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 1,
          "StartColumn" : 3
        }, {
          "EndRow" : 2,
```

```
          "EndRow" : 2,
          "EndColumn" : 5,
          "Texts" : "499,222,12",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 1,
          "StartColumn" : 4
        }, {
          "EndRow" : 2,
          "EndColumn" : 6,
          "Texts" : 99844424,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 1,
          "StartColumn" : 5
        }, {
          "EndRow" : 2,
          "EndColumn" : 7,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 1,
          "StartColumn" : 6
        } ]
      }, {
        "TableColumns" : [ {
          "EndRow" : 3,
          "EndColumn" : 1,
          "Texts" : 2,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 2,
          "StartColumn" : 0
        }, {
          "EndRow" : 3,
          "EndColumn" : 2,
          "Texts" : "变压器",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 2,
          "StartColumn" : 1
        }, {
          "EndRow" : 3,
          "EndColumn" : 3,
          "Texts" : "台",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 2,
          "StartColumn" : 2
        }, {
          "EndRow" : 3,
          "EndColumn" : 4,
          "Texts" : 1,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 2,
```

```
          "StartColumn" : 3
        }, {
          "EndRow" : 3,
          "EndColumn" : 5,
          "Texts" : "205,121,18",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 2,
          "StartColumn" : 4
        }, {
          "EndRow" : 3,
          "EndColumn" : 6,
          "Texts" : "205,121,18",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 2,
          "StartColumn" : 5
        }, {
          "EndRow" : 3,
          "EndColumn" : 7,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 2,
          "StartColumn" : 6
        } ]
      }, {
        "TableColumns" : [ {
          "EndRow" : 4,
          "EndColumn" : 1,
          "Texts" : 3,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 3,
          "StartColumn" : 0
        }, {
          "EndRow" : 4,
          "EndColumn" : 2,
          "Texts" : "豆浆生产线",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 3,
          "StartColumn" : 1
        }, {
          "EndRow" : 4,
          "EndColumn" : 3,
          "Texts" : "台",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 3,
          "StartColumn" : 2
        }, {
          "EndRow" : 4,
          "EndColumn" : 4,
          "Texts" : 1,
```

```
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 3,
          "StartColumn" : 3
        }, {
          "EndRow" : 4,
          "EndColumn" : 5,
          "Texts" : "123,987,23",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 3,
          "StartColumn" : 4
        }, {
          "EndRow" : 4,
          "EndColumn" : 6,
          "Texts" : "123,987,23",
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 3,
          "StartColumn" : 5
        }, {
          "EndRow" : 4,
          "EndColumn" : 7,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 3,
          "StartColumn" : 6
        } ]
      }, {
        "TableColumns" : [ {
          "EndRow" : 5,
          "EndColumn" : 1,
          "Texts" : 4,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 4,
          "StartColumn" : 0
        }, {
          "EndRow" : 5,
          "EndColumn" : 2,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 4,
          "StartColumn" : 1
        }, {
          "EndRow" : 5,
          "EndColumn" : 3,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 4,
          "StartColumn" : 2
        }, {
          "EndRow" : 5,
          "EndColumn" : 4,
```

```
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 4,
          "StartColumn" : 3
        }, {
          "EndRow" : 5,
          "EndColumn" : 5,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 4,
          "StartColumn" : 4
        }, {
          "EndRow" : 5,
          "EndColumn" : 6,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 4,
          "StartColumn" : 5
        }, {
          "EndRow" : 5,
          "EndColumn" : 7,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 4,
          "StartColumn" : 6
        } ]
      }, {
        "TableColumns" : [ {
          "EndRow" : 6,
          "EndColumn" : 1,
          "Texts" : 5,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 5,
          "StartColumn" : 0
        }, {
          "EndRow" : 6,
          "EndColumn" : 2,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 5,
          "StartColumn" : 1
        }, {
          "EndRow" : 6,
          "EndColumn" : 3,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 5,
          "StartColumn" : 2
        }, {
          "EndRow" : 6,
          "EndColumn" : 4,
          "Height" : 0,
          "Width" : 0,
```

```
          "StartRow" : 5,
          "StartColumn" : 3
        }, {
          "EndRow" : 6,
          "EndColumn" : 5,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 5,
          "StartColumn" : 4
        }, {
          "EndRow" : 6,
          "EndColumn" : 6,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 5,
          "StartColumn" : 5
        }, {
          "EndRow" : 6,
          "EndColumn" : 7,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 5,
          "StartColumn" : 6
        } ]
      }, {
        "TableColumns" : [ {
          "EndRow" : 7,
          "EndColumn" : 1,
          "Texts" : 6,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 6,
          "StartColumn" : 0
        }, {
          "EndRow" : 7,
          "EndColumn" : 2,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 6,
          "StartColumn" : 1
        }, {
          "EndRow" : 7,
          "EndColumn" : 3,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 6,
          "StartColumn" : 2
        }, {
          "EndRow" : 7,
          "EndColumn" : 4,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 6,
          "StartColumn" : 3
        }, {
```

```
        }, {
          "EndRow" : 7,
          "EndColumn" : 5,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 6,
          "StartColumn" : 4
        }, {
          "EndRow" : 7,
          "EndColumn" : 6,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 6,
          "StartColumn" : 5
        }, {
          "EndRow" : 7,
          "EndColumn" : 7,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 6,
          "StartColumn" : 6
        } ]
      }, {
        "TableColumns" : [ {
          "EndRow" : 8,
          "EndColumn" : 1,
          "Texts" : 7,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 7,
          "StartColumn" : 0
        }, {
          "EndRow" : 8,
          "EndColumn" : 2,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 7,
          "StartColumn" : 1
        }, {
          "EndRow" : 8,
          "EndColumn" : 3,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 7,
          "StartColumn" : 2
        }, {
          "EndRow" : 8,
          "EndColumn" : 4,
          "Height" : 0,
          "Width" : 0,
          "StartRow" : 7,
          "StartColumn" : 3
        }, {
          "EndRow" : 8,
          "EndColumn" : 5,
```

```
            "Height" : 0,
            "Width" : 0,
            "StartRow" : 7,
            "StartColumn" : 4
          }, {
            "EndRow" : 8,
            "EndColumn" : 6,
            "Height" : 0,
            "Width" : 0,
            "StartRow" : 7,
            "StartColumn" : 5
          }, {
            "EndRow" : 8,
            "EndColumn" : 7,
            "Height" : 0,
            "Width" : 0,
            "StartRow" : 7,
            "StartColumn" : 6
          } ]
        } ]
      }
    }
  }
```

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。

5.3. 文档结构化还原识别

本文介绍文档结构化还原识别TrimDocument的语法及示例。

功能描述

文档结构化还原识别能力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：当前仅支持PDF类型。
- 文档大小：不超过10 MB。

- 文档篇幅：PDF不超过2页。
- 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	系统规定参数。取值：**TrimDocument**。
FileURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/TrimDocument/TrimDocument1.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。
FileType	String	是	pdf	输入文档类型，目前仅支持pdf。
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
RequestId	String	E700C1C1-57E4-413A-9987-56A5A0443CE4	请求ID。
Data	Object		返回的结果数据内容。

名称	类型	示例值	描述
Content	String	<h2 > 2017 年 3 月 40 多家陶企上榜失信被执行人名单 </h2> 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。	解析文档得到的具体结构化内容。

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/viapi-3.0domepic/ocr/TrimDocument/
TrimDocument1.pdf
&OutputType=html
&<公共请求参数>
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<TrimDocumentResponse>
    <RequestId>E700C1C1-57E4-413A-9987-56A5A0443CE4</RequestId>
    <Data>
        <Content>&lt;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 1
2 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;</Content>
    </Data>
</TrimDocumentResponse>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "E700C1C1-57E4-413A-9987-56A5A0443CE4",
  "Data" : {
    "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>"
  }
}
```

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。

5.4. 证件翻拍识别

本文介绍证件翻拍识别DetectCardScreenshot的语法及示例。

功能描述

证件翻拍识别能力可以识别中华人民共和国居民身份证照片是否为屏幕翻拍图。

应用场景

通过网络认证身份证时，检查身份证图片是否为屏幕翻拍图，而非身份证实物获得，避免拍摄的身份证照片被不法分子非法使用。

特色优势

- 鲁棒性强：适用于屏幕翻拍、高清照片翻拍等识别。
- 通用性广：适用于居民身份证、护照、驾驶证、行驶证。

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

- 图像格式：JPEG、JPG、PNG、BMP。

- 图像大小：不超过3 MB。
- 图像分辨率：大于100×100像素，小于3000×3000像素。
- 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	DetectCardScreenshot	系统规定参数。取值：**DetectCardScreenshot**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/DetectCardScreenshot/DetectCardScreenshot3.jpg	图像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。 **说明** 本示例为测试数据，非真实数据。

返回数据

名称	类型	示例值	描述
RequestId	String	1BF3BA27-10BE-403D-AEAA-FAC26B3D2AA3	请求ID。
Data	Object		返回的结果数据内容。
IsBlur	Boolean	false	证件图片是否清晰。 • true：不清晰。 • false：清晰。
IsCard	Boolean	true	图片是否为一张证件照。 • true：证件。 • false：非证件。
SpoofResult	Object		输出结果。

名称	类型	示例值	描述
IsSpoof	Boolean	false	是否翻拍的预测结果。 • true：屏幕翻拍。 • false：相机直拍。
ResultMap	Object		输出分数结果。
ScreenThreshold	Float	90	决策阈值。
ScreenScore	Float	99.99965944391533	置信度。取值范围0~100。

示例

请求示例

```
http(s)://[Endpoint]/?Action=DetectCardScreenshot
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/DetectCardScreenshot/DetectCardScreenshot3.jpg
&公共请求参数
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RequestId>1BF3BA27-10BE-403D-AEAA-FAC26B3D2AA3</RequestId>
<Data>
    <SpoofResult>
        <ResultMap>
            <ScreenScore>99.99965944391533</ScreenScore>
            <ScreenThreshold>90</ScreenThreshold>
        </ResultMap>
        <IsSpoof>false</IsSpoof>
    </SpoofResult>
    <IsBlur>false</IsBlur>
    <IsCard>true</IsCard>
</Data>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "1BF3BA27-10BE-403D-AEAA-FAC26B3D2AA3",
  "Data" : {
    "SpoofResult" : {
      "ResultMap" : {
        "ScreenScore" : 99.99965944391533,
        "ScreenThreshold" : 90
      },
      "IsSpoof" : false
    },
    "IsBlur" : false,
    "IsCard" : true
  }
}
```

错误码

关于证件翻拍识别的错误码，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。

5.5. 门头照识别

本文介绍门头照识别RecognizePoiName的语法及示例。

功能描述

门头照识别能力可以识别门头招牌的图像，并且过滤出门头信息，例如门头照、Logo、店铺地址、联系方式等。

应用场景

- 外卖取货送达认证：外卖骑手在到店取货、送达外卖时拍摄并上传门头照片，通过门头照识别验证真实性。
- 巡店拜访：零售业经销商代表在拜访终端门店过程中，拍摄门店门头照片，结合位置信息，核验拜访的真实性。
- 店铺信息录入：通过拍摄门头照片，识别店铺信息，快速创建客户信息库。

特点优势：

- 更高的识别准确率：基于文字识别技术的门头照文字识别，和图片检索技术的多模型门头招牌图片识别，可提高识别的准确率。
- 实际场景验证：在全国范围的实际使用。算法模型经历了长时间的数据积累和迭代，每天支持海量的实际业务场景的识别。

- 涵盖大部分线下餐饮商户以及部分非餐饮类门店。

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

- 图像格式：JPEG、JPG、PNG、BMP。
- 图像大小：不超过10 MB。
- 图像分辨率：大于15×15像素，小于4096×4096像素。
- 图像要求无遮挡、无明显反光、曝光、阴影，在拍摄门头照时，请尽量以正对角度拍摄。
- 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	RecognizePoiName	系统规定参数。取值：**RecognizePoiName**。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/xuhan/October/RecognizePoiName9.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。

返回数据

名称	类型	示例值	描述
RequestId	String	76D4FAB5-4BFE-5A4F-B2EC-9DB2155F1AF7	请求ID。
Data	Object		返回的结果数据内容。
Summary	Object		主题招牌名。
Brand	String	煌上煌酱鸭	品牌名。

名称	类型	示例值	描述
Score	Float	0.9765625	置信度。
Signboards	Array of Signboards		招牌区域。
Texts	Array of Texts		文字信息。
Type	String	text	类别。枚举取值如下： • text：属于文字文本。 • logo：属于图标或其他。
Score	Float	0.9994	置信度。
Tag	String	name	属性标签。枚举取值如下： • name：招牌名称。 • address：店铺地址。 • phone：联系方式。 • other：其他信息。
Label	String	煌烤鸭	识别出招牌文字的具体内容。
Points	Array of Integer	[524,350,658,370,650,422,517,401]	文字区域按顺时针方向四个点坐标 `[x1,y1,x2,y2,x3,y3,x4,y4]` 。

示例

请求示例

```
http(s)://[Endpoint]/?Action=RecognizePoiName
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/xuhan/October/RecognizePoiName9.jpeg
&公共请求参数
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizePoiNameResponse>
    <RequestId>76D4FAB5-4BFE-5A4F-B2EC-9DB2155F1AF7</RequestId>
    <Data>
        <Signboards>
            <Texts>
```

```
            <Score>0.9083</Score>
            <Type>text</Type>
            <Points>413</Points>
            <Points>452</Points>
            <Points>491</Points>
            <Points>452</Points>
            <Points>491</Points>
            <Points>471</Points>
            <Points>413</Points>
            <Points>471</Points>
            <Label>煌烤鸭</Label>
            <Tag>name</Tag>
        </Texts>
        <Texts>
            <Score>0.7716</Score>
            <Type>text</Type>
            <Points>911</Points>
            <Points>460</Points>
            <Points>946</Points>
            <Points>460</Points>
            <Points>946</Points>
            <Points>472</Points>
            <Points>911</Points>
            <Points>472</Points>
            <Label>容大杜号</Label>
            <Tag>other</Tag>
        </Texts>
        <Texts>
            <Score>0.4564</Score>
            <Type>text</Type>
            <Points>227</Points>
            <Points>448</Points>
            <Points>237</Points>
            <Points>448</Points>
            <Points>237</Points>
            <Points>485</Points>
            <Points>227</Points>
            <Points>485</Points>
            <Label>肚品</Label>
            <Tag>other</Tag>
        </Texts>
        <Texts>
            <Score>0.469</Score>
            <Type>text</Type>
            <Points>320</Points>
            <Points>451</Points>
            <Points>353</Points>
            <Points>451</Points>
            <Points>353</Points>
            <Points>478</Points>
            <Points>320</Points>
            <Points>478</Points>
            <Label>家</Label>
            <Tag>name</Tag>
```

```
            </Texts>
            <Texts>
                <Score>0.9765625</Score>
                <Type>text</Type>
                <Points>132</Points>
                <Points>63</Points>
                <Points>810</Points>
                <Points>63</Points>
                <Points>810</Points>
                <Points>200</Points>
                <Points>132</Points>
                <Points>200</Points>
                <Label>煌上煌酱鸭</Label>
                <Tag>name</Tag>
            </Texts>
        </Signboards>
        <Summary>
            <Brand>煌上煌酱鸭</Brand>
            <Score>0.9765625</Score>
        </Summary>
    </Data>
</RecognizePoiNameResponse>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "76D4FAB5-4BFE-5A4F-B2EC-9DB2155F1AF7",
  "Data" : {
    "Signboards" : {
      "Texts" : [ {
        "Score" : 0.9083,
        "Type" : "text",
        "Points" : [ 413, 452, 491, 452, 491, 471, 413, 471 ],
        "Label" : "煌烤鸭",
        "Tag" : "name"
      }, {
        "Score" : 0.7716,
        "Type" : "text",
        "Points" : [ 911, 460, 946, 460, 946, 472, 911, 472 ],
        "Label" : "容大杜号",
        "Tag" : "other"
      }, {
        "Score" : 0.4564,
        "Type" : "text",
        "Points" : [ 227, 448, 237, 448, 237, 485, 227, 485 ],
        "Label" : "肚品",
        "Tag" : "other"
      }, {
        "Score" : 0.469,
        "Type" : "text",
        "Points" : [ 320, 451, 353, 451, 353, 478, 320, 478 ],
        "Label" : "家",
        "Tag" : "name"
      }, {
        "Score" : 0.9765625,
        "Type" : "text",
        "Points" : [ 132, 63, 810, 63, 810, 200, 132, 200 ],
        "Label" : "煌上煌酱鸭",
        "Tag" : "name"
      } ]
    },
    "Summary" : {
      "Brand" : "煌上煌酱鸭",
      "Score" : 0.9765625
    }
  }
}
```

错误码

关于门头照识别的错误码，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。

5.6. PDF识别

本文介绍PDF识别RecognizePdf的语法及示例。

功能描述

PDF识别能力可以对PDF上的文字进行结构化识别。

应用场景

- 内容审核：可以结合内容审核能力对识别结果进行审核，检测文档中是否存在违规信息。
- 企业报销：可以对PDF格式的增值税发票结构化识别处理后实现报销场景下的自动化。

特色优势

- 精准识别：智能算法升级，精准识别文件内容，保留原始排版。
- 多语种识别：可识别中、英、中英混合等多语种内容。

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

- 文件格式：PDF。
- 文件大小：不超过10 MB。
- 文档篇幅：PDF不超过5页。
- URL地址中不能包含中文字符。

计费说明

关于PDF识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口，如需免费体验调试请前往体验中心。

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	RecognizePdf	系统规定参数。取值：**RecognizePdf**。

名称	类型	是否必选	示例值	描述
FileURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/ocr/xxxx.pdf	文件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
RequestId	String	CD9A9659-ABEE-4A7D-837F-9FDF40879A97	请求ID。
Data	Object		返回的结果数据内容。
Height	Long	788	文档在图中旋转后的高度。
Width	Long	1220	文档在图中旋转后的宽度。
OrgHeight	Long	610	原图高度。
OrgWidth	Long	394	原图宽度。
PageIndex	Long	1	PDF页数。
Angle	Long	0	PDF文件被旋转的角度。
WordsInfo	Array of wordsInfo		文字信息。
Angle	Long	0	被识别字段被旋转的角度。
Word	String	发票代码：012002000211	文字信息。
Height	Long	16	被识别字段的高度。
Width	Long	205	被识别字段的宽度。

名称	类型	示例值	描述
X	Long	863	被识别字段左上角X坐标。
Y	Long	46	被识别字段左上角Y坐标。
Positions	Array of positions		文字信息位置。识别矩形框四个角的坐标顺时针排列，分别为左上XY坐标、右上XY坐标、右下XY坐标、左下XY坐标。
X	Long	863	矩形框X坐标。
Y	Long	43	矩形框Y坐标。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizePdf
&FileURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/ocr/xxxx.pdf
&公共请求参数
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizePdfResponse>
    <RequestId>CD9A9659-ABEE-4A7D-837F-9FDF40879A97</RequestId>
    <Data>
        <WordsInfo>
            <Word>天津增值税</Word>
            <Angle>-88</Angle>
            <X>514</X>
            <Positions>
                <X>397</X>
                <Y>45</Y>
            </Positions>
            <Positions>
                <X>662</X>
                <Y>52</Y>
            </Positions>
            <Positions>
                <X>661</X>
                <Y>82</Y>
            </Positions>
            <Positions>
                <X>396</X>
```

```
                <Y>75</Y>
            </Positions>
            <Y>-69</Y>
            <Height>265</Height>
            <Width>29</Width>
        </WordsInfo>
        <WordsInfo>
            <Word>普通发票</Word>
            <Angle>0</Angle>
            <X>678</X>
            <Positions>
                <X>678</X>
                <Y>48</Y>
            </Positions>
            <Positions>
                <X>824</X>
                <Y>47</Y>
            </Positions>
            <Positions>
                <X>824</X>
                <Y>76</Y>
            </Positions>
            <Positions>
                <X>678</X>
                <Y>77</Y>
            </Positions>
            <Y>48</Y>
            <Height>29</Height>
            <Width>146</Width>
        </WordsInfo>
        <WordsInfo>
            <Word>发票代码：012002000211</Word>
            <Angle>0</Angle>
            <X>863</X>
            <Positions>
                <X>863</X>
                <Y>46</Y>
            </Positions>
            <Positions>
                <X>1068</X>
                <Y>46</Y>
            </Positions>
            <Positions>
                <X>1068</X>
                <Y>62</Y>
            </Positions>
            <Positions>
                <X>863</X>
                <Y>62</Y>
            </Positions>
            <Y>46</Y>
            <Height>16</Height>
            <Width>205</Width>
        </WordsInfo>
```

```
        <OrgWidth>610</OrgWidth>
        <Angle>0</Angle>
        <OrgHeight>394</OrgHeight>
        <Height>788</Height>
        <PageIndex>1</PageIndex>
        <Width>1220</Width>
    </Data>
</RecognizePdfResponse>
```

JSON 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "CD9A9659-ABEE-4A7D-837F-9FDF40879A97",
  "Data" : {
    "WordsInfo" : [ {
      "Word" : "天津增值税",
      "Angle" : -88,
      "X" : 514,
      "Positions" : [ {
        "X" : 397,
        "Y" : 45
      }, {
        "X" : 662,
        "Y" : 52
      }, {
        "X" : 661,
        "Y" : 82
      }, {
        "X" : 396,
        "Y" : 75
      } ],
      "Y" : -69,
      "Height" : 265,
      "Width" : 29
    }, {
      "Word" : "普通发票",
      "Angle" : 0,
      "X" : 678,
      "Positions" : [ {
        "X" : 678,
        "Y" : 48
      }, {
        "X" : 824,
        "Y" : 47
      }, {
        "X" : 824,
        "Y" : 76
      }, {
        "X" : 678,
        "Y" : 77
      } ],
      "Y" : 48,
```

```
      "Height" : 29,
      "Width" : 146
    }, {
      "Word" : "发票代码：012002000211",
      "Angle" : 0,
      "X" : 863,
      "Positions" : [ {
        "X" : 863,
        "Y" : 46
      }, {
        "X" : 1068,
        "Y" : 46
      }, {
        "X" : 1068,
        "Y" : 62
      }, {
        "X" : 863,
        "Y" : 62
      } ],
      "Y" : 46,
      "Height" : 16,
      "Width" : 205
    } ],
    "OrgWidth" : 610,
    "Angle" : 0,
    "OrgHeight" : 394,
    "Height" : 788,
    "PageIndex" : 1,
    "Width" : 1220
  }
}
```


错误码

关于PDF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。

5.7. 视频文字识别

本文介绍RecognizeVideoCharacter视频文字识别的语法及示例。

功能描述

视频文字识别能力可以实现对输入视频的结构化处理，返回对应的文字内容、文字区域坐标及时间等信息。

特色优势

- 支持多种视频格式：支持AVI、MP4、H264编码格式。
- 快捷接入：支持控制台的AI存储零代码快速接入。

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

- 视频格式：AVI、MP4、H264。
- 视频大小：小于20 MB。
- 视频时长：小于60秒。
- 视频帧率：不大于35帧。
- 视频分辨率：大于20×20像素，小于4096×4096像素。
- URL地址中不能包含中文字符。

计费说明

关于视频文字识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口。

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	RecognizeVideoCharacter	系统规定参数。取值：**RecognizeVideoCharacter**。
VideoURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/videoenhan/RecognizeVideoCharacter/xxxx.mp4	视频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
RequestId	String	D3F5BA69-79C4-46A4-B02B-58C4EEBC4C33	请求ID。

名称	类型	示例值	描述
Data	Object		返回的结果数据内容。
Width	Long	1920	视频宽度分辨率，单位像素。
Height	Long	1080	视频高度分辨率，单位像素。
Frames	Array of Frame		视频帧的集合，空信息的帧不展示。
Timestamp	Long	6124533574	帧时间戳，单位秒。
Elements	Array of Element		
Score	Float	0.99	文字区域概率，概率值的范围为[0.0,1.0]。
Text	String	在桃花盛开的地方	文字内容。
TextRectangles	Array of TextRectangle		文字区域位置信息。
Angle	Long	-90	文字区域角度，角度范围[-180, 180]。
Left	Long	213	文字区域左上角X坐标。
Top	Long	98	文字区域左上角Y坐标。
Width	Long	46	文字区域宽度，单位像素。
Height	Long	213	文字区域高度，单位像素。
InputFile	String	oss://my-bucket/a/b/c.mp4	输入视频文件OSS地址。

示例

请求示例

```
http(s)://[Endpoint]/?Action=RecognizeVideoCharacter
&VideoURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/videoenhan/Recognizeo
&公共请求参数
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeVideoCharacterResponse>
    <RequestId>D3F5BA69-79C4-46A4-B02B-58C4EEBC4C33</RequestId>
    <Data>
        <Width>1920</Width>
        <Height>1080</Height>
        <Frames>
            <Timestamp>6124533574</Timestamp>
            <Elements>
                <Score>0.99</Score>
                <Text>在桃花盛开的地方</Text>
                <TextRectangles>
                    <Angle>-90</Angle>
                    <Left>213</Left>
                    <Top>98</Top>
                    <Width>46</Width>
                    <Height>213</Height>
                </TextRectangles>
            </Elements>
        </Frames>
        <InputFile>oss://my-bucket/a/b/c.mp4</InputFile>
    </Data>
</RecognizeVideoCharacterResponse>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "D3F5BA69-79C4-46A4-B02B-58C4EEBC4C33",
  "Data" : {
    "Width" : 1920,
    "Height" : 1080,
    "Frames" : [ {
      "Timestamp" : 6124533574,
      "Elements" : [ {
        "Score" : 0.99,
        "Text" : "在桃花盛开的地方",
        "TextRectangles" : [ {
          "Angle" : -90,
          "Left" : 213,
          "Top" : 98,
          "Width" : 46,
          "Height" : 213
        } ]
      } ]
    } ],
    "InputFile" : "oss://my-bucket/a/b/c.mp4"
  }
}
```

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

关于视频文字识别的错误码，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.车辆交通类识别

6.1. 驾驶证识别

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

功能描述

驾驶证识别能力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

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：不超过4 MB。
- 图像分辨率：大于15×15像素，小于4096×4096像素。
- URL地址中不能包含中文字符。

计费说明

关于驾驶证识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口，如需免费体验调试请前往体验中心。

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	系统规定参数。取值：**RecognizeDriverLicense**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeDriverLicense/jsz2.jpg	图像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	上传照片为驾驶证首页或者副页。 • face：首页 • back：副页

返回数据

名称	类型	示例值	描述
RequestId	String	374D8C7E-9ECC-4750-A87F-50571702F175	请求ID。
Data	Object		返回的结果数据内容。
BackResult	Object		驾驶证副页结果。
ArchiveNumber	String	130601473955	档案编号。
Name	String	张三	姓名。
CardNumber	String	210288898898898888	证号。
Record	String	实习期至2019年05月06日。	记录。
FaceResult	Object		驾驶证首页结果。
VehicleType	String	C1	驾驶证准驾车型。
IssueDate	String	20130208	初次发证日期。格式：YYYYMMDD，例如19800101，即1980年01月01日。
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。

名称	类型	示例值	描述
Address	String	江苏省徐州市铜山区棠张镇田河村1队129号	地址。
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。
LicenseNumber	String	210288898898898888	证号。
Name	String	张三	姓名。
IssueUnit	String	江苏省徐州市公安局交通巡逻警察支队	发证单位。

示例

请求示例

```
http(s)://[Endpoint]/?Action=RecognizeDriverLicense
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeDriverLicense/jsz2.jpg
&Side=face
&公共请求参数
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeDriverLicenseResponse>
    <RequestId>374D8C7E-9ECC-4750-A87F-50571702F175</RequestId>
    <Data>
        <BackResult>
            <ArchiveNumber>130601473955</ArchiveNumber>
            <Name>张三</Name>
            <CardNumber>210288898898898888</CardNumber>
            <Record>实习期至2019年05月06日。</Record>
        </BackResult>
        <FaceResult>
            <VehicleType>C1</VehicleType>
            <IssueDate>20130208</IssueDate>
            <EndDate>20190201</EndDate>
            <Gender>男</Gender>
            <Address>江苏省徐州市铜山区棠张镇田河村1队129号</Address>
            <StartDate>20130208</StartDate>
            <LicenseNumber>210288898898898888</LicenseNumber>
            <Name>张三</Name>
            <IssueUnit>江苏省徐州市公安局交通巡逻警察支队</IssueUnit>
        </FaceResult>
    </Data>
</RecognizeDriverLicenseResponse>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "374D8C7E-9ECC-4750-A87F-50571702F175",
  "Data" : {
    "BackResult" : {
      "ArchiveNumber" : "130601473955",
      "Name" : "张三",
      "CardNumber" : "210288898898898888",
      "Record" : "实习期至2019年05月06日。"
    },
    "FaceResult" : {
      "VehicleType" : "C1",
      "IssueDate" : "20130208",
      "EndDate" : "20190201",
      "Gender" : "男",
      "Address" : "江苏省徐州市铜山区棠张镇田河村1队129号",
      "StartDate" : "20130208",
      "LicenseNumber" : "210288898898898888",
      "Name" : "张三",
      "IssueUnit" : "江苏省徐州市公安局交通巡逻警察支队"
    }
  }
}
```

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。

6.2. 行驶证识别

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

功能描述

行驶证识别能力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

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：不超过3 MB。
- 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。

计费说明

关于行驶证识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口，如需免费体验调试请前往体验中心。

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	系统规定参数。取值：**RecognizeDrivingLicense**。

名称	类型	是否必选	示例值	描述
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeDrivingLicense/xsz2.jpg	图像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	上传照片为行驶证首页或副页。 • face：首页 • back：副页

返回数据

名称	类型	示例值	描述
RequestId	String	1DD989C1-4E08-4E04-9D5D-314901E91226	请求ID。
Data	Object		返回的结果数据内容。
BackResult	Object		行驶证副页结果。
OverallDimension	String	4945x1845x1480	外廓尺寸。
InspectionRecord	String	检验有效期至2014年09月云A(01)	检验记录。
UnladenMass	String	2000	整备质量。
FileNumber	String	苏F123E9	档案编号。
TractionMass	String	100	准牵引总质量。
GrossMass	String	2205	总质量。
PlateNumber	String	苏F123E9	号牌号码。
ApprovedPassengerCapacity	String	5	核定载人数。

名称	类型	示例值	描述
EnergyType	String	-	能源标志。
ApprovedLoad	String	300	核定载质量。
FaceResult	Object		行驶证首页结果。
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。
VehicleType	String	小型普通客车	车辆类型。
Owner	String	张三	所有人名称。
EngineNumber	String	111111	发动机号码。
PlateNumber	String	苏F123E9	车牌号码。
Address	String	中牟县三刘寨村	地址。
UseCharacter	String	非营运	使用性质。
Vin	String	SSVUDDTT2J2022555	车辆识别代号。
RegisterDate	String	20180312	注册日期，格式：YYYYMMDD，例如19800101，即1980年01月01日。

示例

请求示例

```
http(s)://[Endpoint]/?Action=RecognizeDrivingLicense
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeDrivingLicense/xsz2.jpg
&Side=face
&公共请求参数
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeDrivingLicenseResponse>
    <RequestId>1DD989C1-4E08-4E04-9D5D-314901E91226</RequestId>
    <Data>
        <BackResult>
            <OverallDimension>4945x1845x1480</OverallDimension>
            <InspectionRecord>检验有效期至2014年09月云A(01)</InspectionRecord>
            <UnladenMass>2000</UnladenMass>
            <FileNumber>苏F123E9</FileNumber>
            <TractionMass>100</TractionMass>
            <GrossMass>2205</GrossMass>
            <PlateNumber>苏F123E9</PlateNumber>
            <ApprovedPassengerCapacity>5</ApprovedPassengerCapacity>
            <EnergyType>-</EnergyType>
            <ApprovedLoad>300</ApprovedLoad>
        </BackResult>
        <FaceResult>
            <IssueDate>20180313</IssueDate>
            <Model>大众汽车牌SVW6666DFD</Model>
            <VehicleType>小型普通客车</VehicleType>
            <Owner>张三</Owner>
            <EngineNumber>111111</EngineNumber>
            <PlateNumber>苏F123E9</PlateNumber>
            <Address>中牟县三刘寨村</Address>
            <UseCharacter>非营运</UseCharacter>
            <Vin>SSVUDDTT2J2022555</Vin>
            <RegisterDate>20180312</RegisterDate>
        </FaceResult>
    </Data>
</RecognizeDrivingLicenseResponse>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "1DD989C1-4E08-4E04-9D5D-314901E91226",
  "Data" : {
    "BackResult" : {
      "OverallDimension" : "4945x1845x1480",
      "InspectionRecord" : "检验有效期至2014年09月云A(01)",
      "UnladenMass" : "2000",
      "FileNumber" : "苏F123E9",
      "TractionMass" : "100",
      "GrossMass" : "2205",
      "PlateNumber" : "苏F123E9",
      "ApprovedPassengerCapacity" : "5",
      "EnergyType" : "-",
      "ApprovedLoad" : "300"
    },
    "FaceResult" : {
      "IssueDate" : "20180313",
      "Model" : "大众汽车牌SVW6666DFD",
      "VehicleType" : "小型普通客车",
      "Owner" : "张三",
      "EngineNumber" : "111111",
      "PlateNumber" : "苏F123E9",
      "Address" : "中牟县三刘寨村",
      "UseCharacter" : "非营运",
      "Vin" : "SSVUDDTT2J2022555",
      "RegisterDate" : "20180312"
    }
  }
}
```

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。

6.3. 车牌识别

本文介绍车牌识别RecognizeLicensePlate的语法及示例。

功能描述

车牌识别能力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

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：不超过4 MB。
- 图像分辨率：大于15×15像素，小于4096×4096像素。
- URL地址中不能包含中文字符。

计费说明

关于车牌识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口，如需免费体验调试请前往体验中心。

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	系统规定参数。取值：**RecognizeLicensePlate**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeLicensePlate/cpsb1.jpg	图像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
RequestId	String	3F10DAC3-CF4A-487C-BF33-3B8EB9AA12F2	请求ID。
Data	Object		返回的结果数据内容。
Plates	Array of Plate		车牌面板详细信息。

名称	类型	示例值	描述
PlateTypeConfidence	Float	1	车牌类型的置信度，取值范围0~1。
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。
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。
Roi	Object		车牌位置。
W	Integer	141	车牌号宽度。
H	Integer	53	车牌号高度。
Y	Integer	256	车牌号左上角y坐标。
X	Integer	294	车牌号左上角x坐标。
Positions	Array of positions		车牌识别框位置信息。识别矩形框四个角的坐标顺时针排列，分别为左上XY坐标、右上XY坐标、右下XY坐标、左下XY坐标。
X	Long	466	矩形框X坐标。
Y	Long	293	矩形框Y坐标。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeLicensePlate
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeLicensePlate/cpsb1.jpg
&<公共请求参数>
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeLicensePlateResponse>
    <RequestId>3F10DAC3-CF4A-487C-BF33-3B8EB9AA12F2</RequestId>
    <Data>
        <Plates>
            <PlateTypeConfidence>1</PlateTypeConfidence>
            <PlateType>小型汽车</PlateType>
            <Confidence>0.9974534</Confidence>
            <PlateNumber>粤BP57E7</PlateNumber>
            <Roi>
                <W>141</W>
                <H>53</H>
                <Y>256</Y>
                <X>294</X>
            </Roi>
            <Positions>
                <X>466</X>
                <Y>293</Y>
            </Positions>
            <Positions>
                <X>667</X>
                <Y>283</Y>
            </Positions>
            <Positions>
                <X>669</X>
                <Y>330</Y>
            </Positions>
            <Positions>
                <X>469</X>
                <Y>341</Y>
            </Positions>
        </Plates>
    </Data>
</RecognizeLicensePlateResponse>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "3F10DAC3-CF4A-487C-BF33-3B8EB9AA12F2",
  "Data" : {
    "Plates" : {
      "PlateTypeConfidence" : 1,
      "PlateType" : "小型汽车",
      "Confidence" : 0.9974534,
      "PlateNumber" : "粤BP57E7",
      "Roi" : {
        "W" : 141,
        "H" : 53,
        "Y" : 256,
        "X" : 294
      },
      "Positions" : [ {
        "X" : 466,
        "Y" : 293
      }, {
        "X" : 667,
        "Y" : 283
      }, {
        "X" : 669,
        "Y" : 330
      }, {
        "X" : 469,
        "Y" : 341
      } ]
    }
  }
}
```

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。

6.4. VIN码识别

本文介绍VIN码识别RecognizeVINCode的语法及示例。

功能描述

VIN码识别能力可以识别车辆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

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：不超过3 MB。
- 图像分辨率：不限制图像分辨率，但图像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。

计费说明

关于VIN码识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口，如需免费体验调试请前往体验中心。

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	系统规定参数。取值：**RecognizeVINCode**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeVINCode/vin1.jpg	图像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
RequestId	String	911FC8CF-CC27-477E-BE3B-7ED77DF4DFE0	请求ID。
Data	Object		返回的结果数据内容。
VinCode	String	LVBB2FAF777999888	识别的车辆VIN码。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeVINCode
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeVINC
ode/vin1.jpg
&<公共请求参数>
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeVINCodeResponse>
    <RequestId>911FC8CF-CC27-477E-BE3B-7ED77DF4DFE0</RequestId>
    <Data>
        <VinCode>LVBB2FAF777999888</VinCode>
    </Data>
</RecognizeVINCodeResponse>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "911FC8CF-CC27-477E-BE3B-7ED77DF4DFE0",
  "Data" : {
    "VinCode" : "LVBB2FAF777999888"
  }
}
```

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。

7.行业票证类识别

7.1. 出租车发票识别

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

功能描述

出租车发票识别能力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。

输入限制

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：不超过3 MB。
- 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	系统规定参数。取值：**RecognizeTaxiInvoice**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeTaxiInvoice/RecognizeTaxiInvoice2.jpg	图像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
RequestId	String	B2BBBD26-1D3E-4CFA-A80B-6A9266B8D125	请求ID。

名称	类型	示例值	描述
Data	Object		返回的数据内容。
Invoices	Array of Invoice		发票列表详细信息。
Items	Array of Item		每张发票的文字列表。
ItemRoi	Object		发票每个字段对应位置信息。
Size	Object		位置框长宽。
W	Float	887.9998779296875	位置框宽度。
H	Float	81.999984741210938	位置框高度。
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	Object		位置框中心点。
Y	Float	1360	位置框中心点Y坐标。
X	Float	1593	位置框中心点X坐标。
Text	String	86655664	发票每个字段信息。
RotateType	Integer	0	发票旋转角度。 • 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。
InvoiceRoi	Object		发票位置。

名称	类型	示例值	描述
W	Float	1773	发票宽度。
H	Float	3625	发票高度。
Y	Float	302	发票左上角Y坐标。
X	Float	513	发票左上角X坐标。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeTaxiInvoice
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeTaxi
Invoice/RecognizeTaxiInvoice2.jpg
&<公共请求参数>
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<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>
```

```
                <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

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "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" : 1503
```

```
            "X" : 1593,
            "Y" : 1360
          },
          "Angle" : 0,
          "Size" : {
            "W" : 887.9998779296875,
            "H" : 81.99998474121094
          }
        },
        "Text" : "013560325483"
      }, {
        "ItemRoi" : {
          "Center" : {
            "X" : 1744,
            "Y" : 1507.5
          },
          "Angle" : 0,
          "Size" : {
            "W" : 585.9998779296875,
            "H" : 80.99998474121094
          }
        },
        "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" : 81.99998474121094,
            "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" : 91.99998474121094
          }
        },
        "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" : 99.99998474121094
          }
        },
        "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。

7.2. 增值税发票识别

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

功能描述

增值税发票识别能力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

- 图像格式：JPG。
- 图像大小：不超过3 MB。
- 图像分辨率：大于15×15像素，小于4096×4096像素。
- URL地址中不能包含中文字符。

计费说明

关于增值税发票识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口，如需免费体验调试请前往体验中心。

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	系统规定参数。取值：RecognizeVATInvoice。
FileURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeVATInvoice/RecognizeVATInvoice3.jpg	图像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。
FileType	String	是	jpg	输入的图像格式。

返回数据

名称	类型	示例值	描述
RequestId	String	56A10D65-ECE0-59DE-9775-F6494D2AF13B	请求ID。

名称	类型	示例值	描述
Data	Object		返回的结果数据内容。
Box	Object		发票关键字位置框，格式为[左上角点X轴坐标，左上角点Y轴坐标，右下角点X轴坐标，右下角点Y轴坐标]。
PayerRegisterNoes	Array of Float	358,262,567,290	购买方纳税人识别号位置框信息。
PayeeAddresses	Array of Float	355,909,734,939	购买方地址及电话位置框信息。
PayeeBankNames	Array of Float	354,947,938,977	购买方开户行及账号位置框信息。
Checkers	Array of Float	589,1003,662,1033	复核人位置框信息。
TaxAmounts	Array of Float	1606,721,1658,748	合计税额位置框信息。
SumAmounts	Array of Float	32,774,629,805	价税合计位置框信息。
Clerks	Array of Float	986,1003,1060,1033	开票人位置框信息。
InvoiceNoes	Array of Float	1377,78,1478,105	发票号码位置框信息。
InvoiceDates	Array of Float	1376,115,1596,145	开票日期位置框信息。
InvoiceCodes	Array of Float	1378,41,1520,68	发票代码位置框信息。
InvoiceFakeCodes	Array of Float	1376,153,1640,181	校验码位置框信息。
PayerNames	Array of Float	354,222,700,255	购买方名称位置框信息。
PayerBankNames	Array of Float	0,0,0,0	销售方开户行及账号位置框信息。
Payees	Array of Float	189,1003,264,1033	收款人位置框信息。
PayeeNames	Array of Float	356,833,633,865	购买方名称位置框信息。

名称	类型	示例值	描述
InvoiceAmounts	Array of Float	1364, 776,1438,804	价税合计位置框信息。
WithoutTaxAmounts	Array of Float	1265,721,1339,749	合计金额位置框信息。
PayerAddresses	Array of Float	0,0,0,0	销售方地址及电话位置框信息。
PayeeRegisterNoes	Array of Float	356,873,571,902	销售方纳税人识别号位置框信息。
Content	Object		合计金额识别发票各项结果。
PayerAddress	String	浙江省杭州市西湖区杭大路9号聚龙大厦西区15-18楼0571-87901580	购买方地址、电话。
PayeeRegisterNo	String	91420200000123403	销售方纳税人识别号。
PayeeBankName	String	中国银行浙江省分行358458326718	销售方开户行及账号。
InvoiceNo	String	03753869	发票号码。
PayerRegisterNo	String	91420200000123403	购买方纳税人识别号。
Checker	String	张三	复核人。
TaxAmount	String	9.52	合计税额。
InvoiceDate	String	20190415	开票日期。
WithoutTaxAmount	String	190.48	合计金额。
InvoiceAmount	String	200.00	价税合计（小写）。
AntiFakeCode	String	02702870934284730434	校验码。
PayerName	String	三号技术有限责任公司	购买方名称。

名称	类型	示例值	描述
Payee	String	张三	收款人。
SumAmount	String	87	价税合计（大写）。
PayerBankName	String	6221**********1234	购买方开户行及账号。
Clerk	String	张三	开票人。
PayeeName	String	上海机场（集团）有限公司	销售方名称。
PayeeAddress	String	上海虹桥机场迎宾二路161号22342185	销售方地址、电话。
InvoiceCode	String	031001600311	发票代码。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeVATInvoice
&FileType=jpg
&FileURL=https://vision.aliyun.com/experience/detail?&tagName=ocr&children=RecognizeVATInvoice
&<公共请求参数>
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeVATInvoiceResponse>
    <RequestId>56A10D65-ECE0-59DE-9775-F6494D2AF13B</RequestId>
    <Data>
        <Content>
            <Checker/>
            <PayeeAddress>浙江省杭州市西湖区杭大路9号聚龙大厦西区15-18楼0571-87901580</PayeeAddress>
            <PayeeName>三号技术有限责任公司</PayeeName>
            <PayerAddress/>
            <AntiFakeCode>60966753003521159330</AntiFakeCode>
            <PayerBankName/>
            <InvoiceNo>03753869</InvoiceNo>
            <PayerRegisterNo>91420200000123403</PayerRegisterNo>
            <WithoutTaxAmount>82.08</WithoutTaxAmount>
            <InvoiceAmount>87</InvoiceAmount>
            <PayerName>三号技术有限责任公司</PayerName>
```

```
            <TaxAmount>4.92</TaxAmount>
            <PayeeBankName>中国银行浙江省分行358458326718</PayeeBankName>
            <Clerk/>
            <PayeeRegisterNo>91420200000123403</PayeeRegisterNo>
            <InvoiceCode>033002000411</InvoiceCode>
            <InvoiceDate>2020年12月15日</InvoiceDate>
            <SumAmount>87</SumAmount>
            <Payee/>
        </Content>
        <Box>
            <PayerRegisterNoes>118</PayerRegisterNoes>
            <PayerRegisterNoes>128</PayerRegisterNoes>
            <PayerRegisterNoes>278</PayerRegisterNoes>
            <PayerRegisterNoes>138</PayerRegisterNoes>
            <PayerAddresses>0</PayerAddresses>
            <PayerAddresses>0</PayerAddresses>
            <PayerAddresses>0</PayerAddresses>
            <PayerAddresses>0</PayerAddresses>
            <InvoiceAmounts>207</InvoiceAmounts>
            <InvoiceAmounts>347</InvoiceAmounts>
            <InvoiceAmounts>278</InvoiceAmounts>
            <InvoiceAmounts>355</InvoiceAmounts>
            <InvoiceDates>585</InvoiceDates>
            <InvoiceDates>60</InvoiceDates>
            <InvoiceDates>675</InvoiceDates>
            <InvoiceDates>69</InvoiceDates>
            <PayeeAddresses>118</PayeeAddresses>
            <PayeeAddresses>407</PayeeAddresses>
            <PayeeAddresses>407</PayeeAddresses>
            <PayeeAddresses>415</PayeeAddresses>
            <Clerks>0</Clerks>
            <Clerks>0</Clerks>
            <Clerks>0</Clerks>
            <Clerks>0</Clerks>
            <WithoutTaxAmounts>546</WithoutTaxAmounts>
            <WithoutTaxAmounts>327</WithoutTaxAmounts>
            <WithoutTaxAmounts>584</WithoutTaxAmounts>
            <WithoutTaxAmounts>337</WithoutTaxAmounts>
            <Checkers>0</Checkers>
            <Checkers>0</Checkers>
            <Checkers>0</Checkers>
            <Checkers>0</Checkers>
            <TaxAmounts>696</TaxAmounts>
            <TaxAmounts>328</TaxAmounts>
            <TaxAmounts>724</TaxAmounts>
            <TaxAmounts>335</TaxAmounts>
            <InvoiceCodes>583</InvoiceCodes>
            <InvoiceCodes>16</InvoiceCodes>
            <InvoiceCodes>643</InvoiceCodes>
            <InvoiceCodes>25</InvoiceCodes>
            <SumAmounts>207</SumAmounts>
            <SumAmounts>347</SumAmounts>
            <SumAmounts>278</SumAmounts>
            <SumAmounts>355</SumAmounts>
```

```
            <PayerNames>119</PayerNames>
            <PayerNames>109</PayerNames>
            <PayerNames>237</PayerNames>
            <PayerNames>118</PayerNames>
            <InvoiceFakeCodes>583</InvoiceFakeCodes>
            <InvoiceFakeCodes>82</InvoiceFakeCodes>
            <InvoiceFakeCodes>700</InvoiceFakeCodes>
            <InvoiceFakeCodes>90</InvoiceFakeCodes>
            <PayeeBankNames>119</PayeeBankNames>
            <PayeeBankNames>423</PayeeBankNames>
            <PayeeBankNames>282</PayeeBankNames>
            <PayeeBankNames>431</PayeeBankNames>
            <PayeeNames>117</PayeeNames>
            <PayeeNames>371</PayeeNames>
            <PayeeNames>226</PayeeNames>
            <PayeeNames>381</PayeeNames>
            <Payees>0</Payees>
            <Payees>0</Payees>
            <Payees>0</Payees>
            <Payees>0</Payees>
            <InvoiceNoes>582</InvoiceNoes>
            <InvoiceNoes>38</InvoiceNoes>
            <InvoiceNoes>623</InvoiceNoes>
            <InvoiceNoes>46</InvoiceNoes>
            <PayerBankNames>0</PayerBankNames>
            <PayerBankNames>0</PayerBankNames>
            <PayerBankNames>0</PayerBankNames>
            <PayerBankNames>0</PayerBankNames>
            <PayeeRegisterNoes>119</PayeeRegisterNoes>
            <PayeeRegisterNoes>389</PayeeRegisterNoes>
            <PayeeRegisterNoes>247</PayeeRegisterNoes>
            <PayeeRegisterNoes>400</PayeeRegisterNoes>
        </Box>
    </Data>
</RecognizeVATInvoiceResponse>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "56A10D65-ECE0-59DE-9775-F6494D2AF13B",
  "Data" : {
    "Content" : {
      "Checker" : "",
      "PayeeAddress" : "浙江省杭州市西湖区杭大路9号聚龙大厦西区15-18楼0571-87901580",
      "PayeeName" : "三号技术有限责任公司",
      "PayerAddress" : "",
      "AntiFakeCode" : "60966753003521159330",
      "PayerBankName" : "",
      "InvoiceNo" : 3753869,
      "PayerRegisterNo" : "91420200000123403",
      "WithoutTaxAmount" : 82.08,
      "InvoiceAmount" : 87,
      "PayerName" : "三号技术有限责任公司",
      "TaxAmount" : 4.92,
      "PayeeBankName" : "中国银行浙江省分行358458326718",
      "Clerk" : "",
      "PayeeRegisterNo" : "91420200000123403",
      "InvoiceCode" : 33002000411,
      "InvoiceDate" : "2020年12月15日",
      "SumAmount" : 87,
      "Payee" : ""
    },
    "Box" : {
      "PayerRegisterNoes" : [ 118, 128, 278, 138 ],
      "PayerAddresses" : [ 0, 0, 0, 0 ],
      "InvoiceAmounts" : [ 207, 347, 278, 355 ],
      "InvoiceDates" : [ 585, 60, 675, 69 ],
      "PayeeAddresses" : [ 118, 407, 407, 415 ],
      "Clerks" : [ 0, 0, 0, 0 ],
      "WithoutTaxAmounts" : [ 546, 327, 584, 337 ],
      "Checkers" : [ 0, 0, 0, 0 ],
      "TaxAmounts" : [ 696, 328, 724, 335 ],
      "InvoiceCodes" : [ 583, 16, 643, 25 ],
      "SumAmounts" : [ 207, 347, 278, 355 ],
      "PayerNames" : [ 119, 109, 237, 118 ],
      "InvoiceFakeCodes" : [ 583, 82, 700, 90 ],
      "PayeeBankNames" : [ 119, 423, 282, 431 ],
      "PayeeNames" : [ 117, 371, 226, 381 ],
      "Payees" : [ 0, 0, 0, 0 ],
      "InvoiceNoes" : [ 582, 38, 623, 46 ],
      "PayerBankNames" : [ 0, 0, 0, 0 ],
      "PayeeRegisterNoes" : [ 119, 389, 247, 400 ]
    }
  }
}
```

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。

7.3. 火车票识别

本文介绍火车票识别RecognizeTrainTicket的语法及示例。

功能描述

火车票识别能力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

- 图像格式：JPEG、JPG、PNG、BMP、GIF。
- 图像大小：不超过3 MB。
- 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	系统规定参数。取值：**RecognizeTrainTicket**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeTrainTicket/RecognizeTrainTicket3.jpg	图像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
RequestId	String	BE4B73EA-30A0-4573-A548-3A101B34641A	请求ID。
Data	Object		返回的结果数据内容。
Price	Float	104.5	票价。
Destination	String	南京南站	目的站点。
DepartureStation	String	苏州站	始发站点。
Date	String	2017年08月05日22:09开	乘车日期时间。
Number	String	G7350	车次号。
Seat	String	04车13A号	座位车厢及座次号。
Name	String	帅帅	乘车人姓名。
Level	String	二等座	座位席别。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeTrainTicket
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeTrainTicket/RecognizeTrainTicket3.jpg
&<公共请求参数>
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeTrainTicketResponse>
    <RequestId>BE4B73EA-30A0-4573-A548-3A101B34641A</RequestId>
    <Data>
        <Price>104.5</Price>
        <Destination>南京南站</Destination>
        <DepartureStation>苏州站</DepartureStation>
        <Date>2017年08月05日22：09开</Date>
        <Number>G7350</Number>
        <Seat>04车13A号</Seat>
        <Name>帅帅</Name>
        <Level>二等座</Level>
    </Data>
</RecognizeTrainTicketResponse>
```

`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "BE4B73EA-30A0-4573-A548-3A101B34641A",
  "Data" : {
    "Price" : 104.5,
    "Destination" : "南京南站",
    "DepartureStation" : "苏州站",
    "Date" : "2017年08月05日22：09开",
    "Number" : "G7350",
    "Seat" : "04车13A号",
    "Name" : "帅帅",
    "Level" : "二等座"
  }
}
```

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小时后会被系统自动清理删除。

7.4. 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

外卖单识别能力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。

输入限制

- 图像格式：JPEG、JPG、PNG、BMP。
- 图像大小：不超过10 MB。
- 图像分辨率：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	系统规定参数。取值：**RecognizeTakeoutOrder**。
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeTakeoutOrder/RecognizeTakeoutOrder1.jpg	图像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
RequestId	String	CFB676F0-45D9-520F-BB27-232B90409F87	请求ID。
Data	Object		返回的结果数据内容。
Elements	Array of Element		识别出的外卖单具体内容。
Value	String	4.00	具体字段对应的内容。

名称	类型	示例值	描述
Boxes	Array of Integer	[2177, 2082, 2321, 2152]	字段位置框信息，格式为[左上角点X轴坐标，左上角点Y轴坐标，右上角点X轴坐标，右上角点Y轴坐标]。
Score	Float	0.68	识别出的字段信息的置信度。
Name	String	delivery_fee	外卖单上关键字段。包括： • 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

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeTakeoutOrder
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeTake
outOrder/RecognizeTakeoutOrder1.jpg
&<公共请求参数>
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeTakeoutOrderResponse>
    <RequestId>CFB676F0-45D9-520F-BB27-232B90409F87</RequestId>
    <Data>
        <Elements>
            <Score>0.9995644688606262</Score>
            <Value>4.00</Value>
            <Name>delivery_fee</Name>
            <Boxes>508</Boxes>
            <Boxes>206</Boxes>
            <Boxes>531</Boxes>
            <Boxes>215</Boxes>
        </Elements>
```

```
        <Elements>
            <Score>0.9991578459739685</Score>
            <Value>0.00</Value>
            <Name>packaging_fee</Name>
            <Boxes>509</Boxes>
            <Boxes>195</Boxes>
            <Boxes>531</Boxes>
            <Boxes>203</Boxes>
        </Elements>
        <Elements>
            <Score>0.9390342831611633</Score>
            <Value>起峰便利店（普迅中路店）</Value>
            <Name>shop_name</Name>
            <Boxes>386</Boxes>
            <Boxes>52</Boxes>
            <Boxes>486</Boxes>
            <Boxes>61</Boxes>
        </Elements>
        <Elements>
            <Score>0.9989892244338989</Score>
            <Value>13588503737</Value>
            <Name>telephone_number</Name>
            <Boxes>386</Boxes>
            <Boxes>62</Boxes>
            <Boxes>437</Boxes>
            <Boxes>69</Boxes>
        </Elements>
        <Elements>
            <Score>0.9986419677734375</Score>
            <Value>-2.00</Value>
            <Name>total_benefits</Name>
            <Boxes>504</Boxes>
            <Boxes>252</Boxes>
            <Boxes>531</Boxes>
            <Boxes>260</Boxes>
        </Elements>
        <Elements>
            <Score>0.9988936185836792</Score>
            <Value>4.00</Value>
            <Name>total_other</Name>
            <Boxes>509</Boxes>
            <Boxes>241</Boxes>
            <Boxes>532</Boxes>
            <Boxes>248</Boxes>
        </Elements>
    </Data>
</RecognizeTakeoutOrderResponse>
```


`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "CFB676F0-45D9-520F-BB27-232B90409F87",
  "Data" : {
    "Elements" : [ {
      "Score" : 0.9995644688606262,
      "Value" : 4,
      "Name" : "delivery_fee",
      "Boxes" : [ 508, 206, 531, 215 ]
    }, {
      "Score" : 0.9991578459739685,
      "Value" : 0,
      "Name" : "packaging_fee",
      "Boxes" : [ 509, 195, 531, 203 ]
    }, {
      "Score" : 0.9390342831611633,
      "Value" : "起峰便利店(普迅中路店) ",
      "Name" : "shop_name",
      "Boxes" : [ 386, 52, 486, 61 ]
    }, {
      "Score" : 0.9989892244338989,
      "Value" : 13588503737,
      "Name" : "telephone_number",
      "Boxes" : [ 386, 62, 437, 69 ]
    }, {
      "Score" : 0.9986419677734375,
      "Value" : -2,
      "Name" : "total_benefits",
      "Boxes" : [ 504, 252, 531, 260 ]
    }, {
      "Score" : 0.9988936185836792,
      "Value" : 4,
      "Name" : "total_other",
      "Boxes" : [ 509, 241, 532, 248 ]
    } ]
  }
}
```

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。

7.5. 定额发票识别

本文介绍定额发票识别RecognizeQuotaInvoice的语法及示例。

功能描述

定额发票识别能力可以对定额发票上的发票号码、发票代码、发票金额进行结构化识别。

应用场景

- 财税报销：识别录入各类定额发票票面关键信息，应用于税务核算及内部报销场景，有效减少人工核算工作量，简化报销流程，实现财务报销的自动化。
- 账单记录：识别提取发票金额信息，应用于理财记账场景，快速录入账单信息，有效降低用户输入成本，提升用户使用体验。

特色优势

- 实现自动化：有效减少人工核算工作量，简化报销流程，实现财务报销的自动化。
- 节约成本：快速录入账单信息，有效降低用户输入成本，提升用户使用体验。

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

- 图像格式：JPEG、JPG、PNG、BMP。
- 图像大小：不超过4 MB。
- 图像分辨率：大于15×15像素，小于4096×4096像素。
- URL地址中不能包含中文字符。

计费说明

关于定额发票识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口，如需免费体验调试请前往体验中心。

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	RecognizeQuotaInvoice	系统规定参数。取值：**RecognizeQuotaInvoice**。

名称	类型	是否必选	示例值	描述
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeQuotaInvoice/RecognizeQuotaInvoice1.jpg	图像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
RequestId	String	BC4C12D0-7FD3-419A-B997-A91212DF6D82	请求ID。
Data	Object		返回的结果数据内容。
Angle	Long	1	票据被旋转的角度。范围为0~360，0表示向上，90表示向右，180表示向下，270度表示向左。
Height	Long	624	票据旋转后的高度。
Width	Long	865	票据旋转后的宽度。
OrgHeight	Long	610	原图高度。
OrgWidth	Long	855	原图宽度。
Content	Object		识别内容。
SumAmount	String	壹拾元整	大写金额。
InvoiceCode	String	144031800103	发票代码。
InvoiceNo	String	40637706	发票号码。

名称	类型	示例值	描述
InvoiceAmount	String	10	小写金额。
InvoiceDetails	String	税务局代码：国税，行政区划代码：深圳市，年份：2018，发票行业代码：None，发票类别代码：None，金额版：万元版，批次号：03	发票代码解析。
KeyValueInfos	Array of keyValueInfos		位置信息。
Key	String	大写金额	被识别字段名称。
Value	String	壹拾元整	被识别字段的值。
ValuePositions	Array of valuePositions		被识别字段位置信息。四个角的坐标顺时针排列。
X	Long	544	矩形框X坐标。
Y	Long	387	矩形框Y坐标。
ValueScore	Float	100	被识别字段**Value**值置信度，范围0~100。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeQuotaInvoice
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeQuot
aInvoice/RecognizeQuotaInvoice1.jpg
&公共请求参数
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeQuotaInvoiceResponse>
    <RequestId>BC4C12D0-7FD3-419A-B997-A91212DF6D82</RequestId>
    <Data>
        <OrgWidth>855</OrgWidth>
        <Angle>1</Angle>
```

```
        <KeyValueInfos>
            <ValuePositions>
                <X>544</X>
                <Y>387</Y>
            </ValuePositions>
            <ValuePositions>
                <X>544</X>
                <Y>443</Y>
            </ValuePositions>
            <ValuePositions>
                <X>308</X>
                <Y>443</Y>
            </ValuePositions>
            <ValuePositions>
                <X>308</X>
                <Y>387</Y>
            </ValuePositions>
            <Value>壹拾元整</Value>
            <ValueScore>100</ValueScore>
            <Key>大写金额</Key>
        </KeyValueInfos>
        <KeyValueInfos>
            <ValuePositions>
                <X>310</X>
                <Y>237</Y>
            </ValuePositions>
            <ValuePositions>
                <X>575</X>
                <Y>234</Y>
            </ValuePositions>
            <ValuePositions>
                <X>576</X>
                <Y>259</Y>
            </ValuePositions>
            <ValuePositions>
                <X>311</X>
                <Y>262</Y>
            </ValuePositions>
            <Value>144031800103</Value>
            <ValueScore>100</ValueScore>
            <Key>发票代码</Key>
        </KeyValueInfos>
        <KeyValueInfos>
            <ValuePositions>
                <X>480</X>
                <Y>298</Y>
            </ValuePositions>
            <ValuePositions>
                <X>480</X>
                <Y>321</Y>
            </ValuePositions>
            <ValuePositions>
                <X>309</X>
                <Y>321</Y>
```

```
            </ValuePositions>
            <ValuePositions>
                <X>309</X>
                <Y>298</Y>
            </ValuePositions>
            <Value>40637706</Value>
            <ValueScore>100</ValueScore>
            <Key>发票号码</Key>
        </KeyValueInfos>
        <KeyValueInfos>
            <Value/>
            <ValueScore>100</ValueScore>
            <Key>小写金额</Key>
        </KeyValueInfos>
        <KeyValueInfos>
            <ValuePositions>
                <X>310</X>
                <Y>237</Y>
            </ValuePositions>
            <ValuePositions>
                <X>575</X>
                <Y>234</Y>
            </ValuePositions>
            <ValuePositions>
                <X>576</X>
                <Y>259</Y>
            </ValuePositions>
            <ValuePositions>
                <X>311</X>
                <Y>262</Y>
            </ValuePositions>
            <Value>税务局代码：国税，行政区划代码：深圳市，年份：2018，发票行业代码：None，发票类别代码：None，金额版：万元版，批次号：03</Value>
            <ValueScore>100</ValueScore>
            <Key>发票代码解析</Key>
        </KeyValueInfos>
        <Content>
            <InvoiceDetails>{"批次号":"03","年份":"2018","税务局代码":"国税","发票行业代码":"None","金额版":"万元版","行政区划代码":"深圳市","发票类别代码":"None"}</InvoiceDetails>
            <InvoiceCode>144031800103</InvoiceCode>
            <InvoiceNo>40637706</InvoiceNo>
            <InvoiceAmount/>
            <SumAmount>壹拾元整</SumAmount>
        </Content>
        <OrgHeight>610</OrgHeight>
        <Height>624</Height>
        <Width>865</Width>
    </Data>
</RecognizeQuotaInvoiceResponse>
```


`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
```

Content-Type:application/json


```
{
  "RequestId" : "BC4C12D0-7FD3-419A-B997-A91212DF6D82",
  "Data" : {
    "OrgWidth" : 855,
    "Angle" : 1,
    "KeyValueInfos" : [ {
      "ValuePositions" : [ {
        "X" : 544,
        "Y" : 387
      }, {
        "X" : 544,
        "Y" : 443
      }, {
        "X" : 308,
        "Y" : 443
      }, {
        "X" : 308,
        "Y" : 387
      } ],
      "Value" : "壹拾元整",
      "ValueScore" : 100,
      "Key" : "大写金额"
    }, {
      "ValuePositions" : [ {
        "X" : 310,
        "Y" : 237
      }, {
        "X" : 575,
        "Y" : 234
      }, {
        "X" : 576,
        "Y" : 259
      }, {
        "X" : 311,
        "Y" : 262
      } ],
      "Value" : 144031800103,
      "ValueScore" : 100,
      "Key" : "发票代码"
    }, {
      "ValuePositions" : [ {
        "X" : 480,
        "Y" : 298
      }, {
        "X" : 480,
        "Y" : 321
      }, {
        "X" : 309,
        "Y" : 321
      }, {
        "X" : 309,
        "Y" : 298
      } ],
      "Value" : 40637706,
```

```
      "ValueScore" : 100,
      "Key" : "发票号码"
    }, {
      "Value" : "",
      "ValueScore" : 100,
      "Key" : "小写金额"
    }, {
      "ValuePositions" : [ {
        "X" : 310,
        "Y" : 237
      }, {
        "X" : 575,
        "Y" : 234
      }, {
        "X" : 576,
        "Y" : 259
      }, {
        "X" : 311,
        "Y" : 262
      } ],
      "Value" : "税务局代码：国税，行政区划代码：深圳市，年份：2018，发票行业代码：None，发票类别代码：None，金额版：万元版，批次号：03",
      "ValueScore" : 100,
      "Key" : "发票代码解析"
    } ],
    "Content" : {
      "InvoiceDetails" : "{\"批次号\":\"03\",\"年份\":\"2018\",\"税务局代码\":\"国税\",\"发票行业代码\":\"None\",\"金额版\":\"万元版\",\"行政区划代码\":\"深圳市\",\"发票类别代码\":\"None\"}",
      "InvoiceCode" : 144031800103,
      "InvoiceNo" : 40637706,
      "InvoiceAmount" : "",
      "SumAmount" : "壹拾元整"
    },
    "OrgHeight" : 610,
    "Height" : 624,
    "Width" : 865
  }
}
```


错误码

关于定额发票识别的错误码，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。

7.6. 增值税发票卷票识别

本文介绍增值税发票卷票识别RecognizeTicketInvoice的语法及示例。

功能描述

增值税发票卷票识别能力支持对卷票上包括价税合计、发票代码、发票号码、合计税额、合计金额、密码区、开票日期、税率、购买方识别号、销售方识别号等结构化识别。

应用场景

- 发票验真：智能识别发票代码、号码、开具金额、开票日期四个关键字段，以便快速接入税务机关发票查验平台进行真伪查验，有效降低人力成本，控制业务风险。
- 账单记录：对发票金额、开票日期等信息进行自动识别和录入，应用于理财记账场景，帮助用户快速录入账单信息，降低用户输入成本，提升使用体验。

特色优势

- 全字段识别：支持对值税发票卷票关键字段的结构化识别，能够满足财税报销等场景中对票据字段的识别需求。
- 节约成本：识别发票信息后，根据需求录入发票信息，有效降低用户输入成本，提升用户使用体验。

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

- 图像格式：JPEG、JPG、PNG、BMP。
- 图像大小：不超过4 MB。
- 图像分辨率：大于15×15像素，小于4096×4096像素。
- URL地址中不能包含中文字符。

计费说明

关于增值税发票卷票识别的计费方式及报价，请参见计费介绍。

> **说明** 下方调试接口为付费接口，如需免费体验调试请前往体验中心。

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	RecognizeTicketInvoice	系统规定参数。取值：**RecognizeTicketInvoice**。

名称	类型	是否必选	示例值	描述
ImageURL	String	是	http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeTicketInvoice/RecognizeTicketInvoice1.png	图像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
RequestId	String	063C0178-7EA3-4754-96FB-C0C9AE6B9AAE	请求ID。
Data	Object		返回的结果数据内容。
Count	Long	1	票据数量。
Height	Long	594	票据旋转后的高度。
Width	Long	594	票据旋转后的宽度。
OrgHeight	Long	1417	原图高度。
OrgWidth	Long	1417	原图宽度。
Results	Array of results		识别结果。
Index	Long	1	票据在图像中的索引。
Content	Object		识别内容。
InvoiceCode	String	044031860107	发票代码。
InvoiceNumber	String	09267581	发票号码。

名称	类型	示例值	描述
InvoiceDate	String	2018-09-20	开票日期。
AntiFakeCode	String	81931914902643039780	校验码。
PayeeName	String	深圳市xxxx有限公司	销售方名称。
PayeeRegisterNo	String	914403002794492693	销售方税号。
PayerName	String	深圳市xxxx有限公司	购买方名称。
PayerRegisterNo	String	91440300MA5EXWHW6F	购买方税号。
SumAmount	String	￥220.00	总价。
Type	String	卷票	票据类型。支持识别以下几种： • 增值税发票 • 出租车票 • 定额发票 • 机动车销售发票 • 卷票
KeyValueInfos	Array of keyValueInfos		位置信息。
Key	String	发票代码	被识别字段名称。
Value	String	044031860107	被识别字段的值。
ValuePositions	Array of valuePositions		被识别字段位置信息。四个角的坐标顺时针排列。
X	Long	586	矩形框X坐标。
Y	Long	16	矩形框Y坐标。
ValueScore	Float	100	被识别字段**Value**值置信度，范围0~100。

名称	类型	示例值	描述
SliceRectangle	Array of sliceRectangle		票据识别矩形框四个角的坐标顺时针排列。
X	Long	586	矩形框X坐标。
Y	Long	16	矩形框Y坐标。

示例

请求示例

```
http(s)://ocr.cn-shanghai.aliyuncs.com/?Action=RecognizeTicketInvoice
&ImageURL=http://viapi-test.oss-cn-shanghai.aliyuncs.com/viapi-3.0domepic/ocr/RecognizeTick
etInvoice/RecognizeTicketInvoice1.png
&公共请求参数
```

正常返回示例

`XML` 格式

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RecognizeTicketInvoiceResponse>
    <RequestId>063C0178-7EA3-4754-96FB-C0C9AE6B9AAE</RequestId>
    <Data>
        <OrgWidth>594</OrgWidth>
        <Results>
            <Type>卷票</Type>
            <SliceRectangle>
                <X>2</X>
                <Y>19</Y>
            </SliceRectangle>
            <SliceRectangle>
                <X>586</X>
                <Y>16</Y>
            </SliceRectangle>
            <SliceRectangle>
                <X>580</X>
                <Y>1408</Y>
            </SliceRectangle>
            <SliceRectangle>
                <X>0</X>
                <Y>1410</Y>
            </SliceRectangle>
            <KeyValueInfos>
                <Value>44031860107</Value>
                <ValueScore>100</ValueScore>
                <Key>发票代码</Key>
            </KeyValueInfos>
            <KeyValueInfos>
```

```
            <Value>9267581</Value>
            <ValueScore>100</ValueScore>
            <Key>发票号码</Key>
        </KeyValueInfos>
        <KeyValueInfos>
            <Value>2018-09-20</Value>
            <ValueScore>100</ValueScore>
            <Key>开票日期</Key>
        </KeyValueInfos>
        <KeyValueInfos>
            <Value>81931914902643030000</Value>
            <ValueScore>100</ValueScore>
            <Key>校验码</Key>
        </KeyValueInfos>
        <KeyValueInfos>
            <Value>深圳市xxxx有限公司</Value>
            <ValueScore>100</ValueScore>
            <Key>销售方名称</Key>
        </KeyValueInfos>
        <KeyValueInfos>
            <Value>914403002794492700</Value>
            <ValueScore>100</ValueScore>
            <Key>销售方税号</Key>
        </KeyValueInfos>
        <KeyValueInfos>
            <Value>深圳市xxxx有限公司</Value>
            <ValueScore>100</ValueScore>
            <Key>购买方名称</Key>
        </KeyValueInfos>
        <KeyValueInfos>
            <Value>91440300MA5EXWHW6F</Value>
            <ValueScore>99</ValueScore>
            <Key>购买方税号</Key>
        </KeyValueInfos>
        <KeyValueInfos>
            <Value>￥220.00</Value>
            <ValueScore>99</ValueScore>
            <Key>总价</Key>
        </KeyValueInfos>
        <Content>
            <PayeeName>深圳市xxxx有限公司</PayeeName>
            <AntiFakeCode>81931914902643030000</AntiFakeCode>
            <InvoiceCode>44031860107</InvoiceCode>
            <PayeeRegisterNo>914403002794492700</PayeeRegisterNo>
            <PayerRegisterNo>91440300MA5EXWHW6F</PayerRegisterNo>
            <InvoiceNumber>9267581</InvoiceNumber>
            <InvoiceDate>2018-09-20</InvoiceDate>
            <PayerName>深圳市xxxx有限公司</PayerName>
            <SumAmount>￥220.00</SumAmount>
        </Content>
        <Index>1</Index>
    </Results>
    <OrgHeight>1417</OrgHeight>
    <Height>1417</Height>
```

```
        <Count>1</Count>
        <Width>594</Width>
    </Data>
</RecognizeTicketInvoiceResponse>
```

JSON 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "063C0178-7EA3-4754-96FB-C0C9AE6B9AAE",
  "Data" : {
    "OrgWidth" : 594,
    "Results" : {
      "Type" : "卷票",
      "SliceRectangle" : [ {
        "X" : 2,
        "Y" : 19
      }, {
        "X" : 586,
        "Y" : 16
      }, {
        "X" : 580,
        "Y" : 1408
      }, {
        "X" : 0,
        "Y" : 1410
      } ],
      "KeyValueInfos" : [ {
        "Value" : 44031860107,
        "ValueScore" : 100,
        "Key" : "发票代码"
      }, {
        "Value" : 9267581,
        "ValueScore" : 100,
        "Key" : "发票号码"
      }, {
        "Value" : "2018-09-20",
        "ValueScore" : 100,
        "Key" : "开票日期"
      }, {
        "Value" : 81931914902643030000,
        "ValueScore" : 100,
        "Key" : "校验码"
      }, {
        "Value" : "深圳市xxxx有限公司",
        "ValueScore" : 100,
        "Key" : "销售方名称"
      }, {
        "Value" : 914403002794492700,
        "ValueScore" : 100,
        "Key" : "销售方税号"
      }, {
        "Value" : "深圳市xxxx有限公司",
```

```
        "ValueScore" : 100,
        "Key" : "购买方名称"
      }, {
        "Value" : "91440300MA5EXWHW6F",
        "ValueScore" : 99,
        "Key" : "购买方税号"
      }, {
        "Value" : "￥220.00",
        "ValueScore" : 99,
        "Key" : "总价"
      } ],
      "Content" : {
        "PayeeName" : "深圳市xxxx有限公司",
        "AntiFakeCode" : 81931914902643030000,
        "InvoiceCode" : 44031860107,
        "PayeeRegisterNo" : 914403002794492700,
        "PayerRegisterNo" : "91440300MA5EXWHW6F",
        "InvoiceNumber" : 9267581,
        "InvoiceDate" : "2018-09-20",
        "PayerName" : "深圳市xxxx有限公司",
        "SumAmount" : "￥220.00"
      },
      "Index" : 1
    },
    "OrgHeight" : 1417,
    "Height" : 1417,
    "Count" : 1,
    "Width" : 594
  }
}
```


错误码

关于增值税发票卷票识别的错误码，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。

8.查询异步任务结果

本文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。

> **说明** 异步任务的文件有过期时间，有效期是30分钟。如需长期使用，需及时将文件下载到本地服务器或存储在OSS中。有关OSS更多操作信息，请参见上传文件。

当前文字识别类目中文档结构化还原识别（TrimDocument）和视频文字识别（RecognizeVideoCharacter）为异步接口，需要调用GetAsyncJobResult来获取真实的请求结果。

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
RequestId	String	A1F44EC4-118D-4A03-B213-F908F36F7DAA	请求ID。
Data	Object		返回的数据内容。
Status	String	PROCESS_SUCCESS	异步任务状态。包括： • 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

名称	类型	示例值	描述
ErrorMessage	String	paramsIllegal	异步任务错误码信息。
Result	String	{\"Content\":\" <h2 > 2017 年 3 月 40 多家陶企上榜失信被执行人名单 </h2> 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。 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。 <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	异步任务返回的真实请求结果。

名称	类型	示例值	描述
		</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>\"}	
ErrorCode	String	InvalidParameter	异步任务错误码。
JobId	String	49E2CC28-ED1D-4CC5-854D-7D0AE2B20976	异步任务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

```
HTTP/1.1 200 OK
Content-Type:application/xml
<RequestId>A1F44EC4-118D-4A03-B213-F908F36F7DAA</RequestId>
<Data>
    <Status>PROCESS_SUCCESS</Status>
    <ErrorMessage>paramsIllegal</ErrorMessage>
    <Result>{\"Content\":\"&lt;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>
    <ErrorCode>InvalidParameter</ErrorCode>
    <JobId>49E2CC28-ED1D-4CC5-854D-7D0AE2B20976</JobId>
</Data>
```


`JSON` 格式

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "RequestId" : "A1F44EC4-118D-4A03-B213-F908F36F7DAA",
  "Data" : {
    "Status" : "PROCESS_SUCCESS",
    "ErrorMessage" : "paramsIllegal",
    "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>\\\"}",
    "ErrorCode" : "InvalidParameter",
    "JobId" : "49E2CC28-ED1D-4CC5-854D-7D0AE2B20976"
  }
}
```

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

关于查询异步任务结果的错误码，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。

9.公共信息

9.1. 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。

9.2. 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。
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。

名称	类型	是否必填	说明
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 `java.util.UUID.randomUUID()` 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。

9.3. 公共响应参数

阿里云视觉智能开放平台是RPC 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>
```

10.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。

问题诊断中心

使用问题诊断中心可帮助开发者更高效地诊断API使用过程中的问题。

文字识别错误码

状态码	错误码	描述
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： • 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。 • 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。

状态码	错误码	描述
400	InvalidImage.Region	图片所在地域错误。
400	InvalidResult	OCR识别失败、OCR识别异常。
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AccessKeyId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。具体请参见能力开通。
404	InvalidAccessKeyId.NotFound	AccessKeyID不存在。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。

状态码	错误码	描述
500	InternalError	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。当前系统仅支持 `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（23109592）中反馈问题。