

ALIBABA CLOUD

阿里云

视觉智能开放平台

目标检测

文档版本：20200925

 阿里云

法律声明

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

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

通用约定

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

目录

1.目标检测介绍	05
2.车辆检测	06
2.1. 机动车检测	06
2.2. 车辆部件识别	09
2.3. 车险图片分类	23
2.4. 仪表盘识别	28
2.5. 车辆损伤识别	33
3.车辆维修	50
3.1. 生成车辆维修方案	50
3.2. 查询车辆维修方案及费用	52
4.通用检测	59
4.1. 主体检测	59
4.2. 白底图检测	61
4.3. 透明图检测	63
4.4. 物体检测	65
5.公共信息	78
5.1. 请求结构	78
5.2. 公共请求参数	78
5.3. 公共响应参数	79
6.常见错误码	82

1. 目标检测介绍

目标检测技术基于阿里云计算机视觉与深度学习技术，对通用或指定的目标进行检测、识别、打标。目标检测技术广泛应用于安防、交通等行业内图像场景的目标检测与跟踪。

服务开通

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

能力介绍

目前阿里云视觉智能开放平台上线的目标检测能力包括：

类别	能力	说明
通用检测	主体检测	检测图像中的内容主体，返回该主体的区域位置/坐标信息。
	白底图检测	检测图片背景是否为白底。
	透明图检测	检测图片背景是否透明。
	物体检测	检测输入图像中的物体。
车辆检测	车辆损伤识别	针对常见小汽车车型，识别车辆外观受损部件及损伤类型，可识别数十种车辆部件，五大类外观损伤（刮擦、凹陷、开裂、褶皱、穿洞）。
	车辆部件识别	检测图片中车辆部件的位置以及名称。
	车辆仪表盘识别	识别车辆仪表盘上故障灯等信息。
	机动车检测	检测图像中的机动车主体，返回该机动车主体的区域位置/坐标信息。
	车险图片分类	对输入的车险图片进行分类。

应用场景

目标检测应用场景如下：

- 电商详情页装修

检测图像中商品主体的坐标信息，将其放置在商品详情页中适当的位置上，以完成详情页智能装修布局与智能设计。

- 路况分析

检测图像中道路、卡口的车辆位置和数量信息，分析道路交通状况，为交通疏导提供参考信息。

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

2. 车辆检测

2.1. 机动车检测

本文档介绍机动车检测DetectVehicle的语法及示例。

功能描述

DetectVehicle可以检测图像中的机动车主体，返回该机动车主体的区域位置，坐标信息。

前提条件

请确保您已开通[目标检测服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 请求格式：JPEG、JPG、PNG、BMP。
- 图像大小：图像大小不超过3M。
- 图像分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- 图片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	DetectVehicle	要执行的操作。取值：DetectVehicle。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/7%E6%B1%BD%E8%BD%A6%E5%9B%BE%E7%89%87cljc.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。

名称	类型	示例值	描述
DetectObjectInfoList	Array of DetectObjectInfo		机动车检测框的集合。
Boxes	List	0	检测框坐标，格式为 [left, top, right, bottom]。例如 [877, 95, 1021, 177]。
Id	Integer	1	检测框的ID。
Score	Float	0.897	检测框的分数，范围为0~1。
Type	String	vehicle	检测框对应的物体，值为 vehicle 代表机动车。
Height	Integer	436	图片的高度。
Width	Integer	612	图片的宽度。
RequestId	String	77BB1EB9-1707-4423-8391-CDAE6D29F9D5	请求ID。

示例

请求示例

```
http(s)://[Endpoint]/?Action=DetectVehicle
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/7%E6%B1%BD%E8%BD%A6%E5%9B%BE%E7%89%87cljc.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>77BB1EB9-1707-4423-8391-CDAE6D29F9D5</RequestId>
<Data>
  <DetectObjectInfoList>
    <Score>0.897</Score>
    <Type>vehicle</Type>
    <Id>1</Id>
    <Boxes>26</Boxes>
    <Boxes>17</Boxes>
    <Boxes>604</Boxes>
    <Boxes>335</Boxes>
  </DetectObjectInfoList>
  <Height>436</Height>
  <Width>612</Width>
</Data>
```

JSON 格式

```
{
  "RequestId": "77BB1EB9-1707-4423-8391-CDAE6D29F9D5",
  "Data": {
    "DetectObjectInfoList": [
      {
        "Score": 0.897,
        "Type": "vehicle",
        "Id": 1,
        "Boxes": [
          26,
          17,
          604,
          335
        ]
      }
    ],
    "Height": 436,
    "Width": 612
  }
}
```

错误码

关于机动车检测的错误码，详情请参见[常见错误码](#)。

SDK参考

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

安全声明

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

2.2. 车辆部件识别

本文档为您介绍车辆部件识别RecognizeVehicleParts的语法及示例。

功能描述

RecognizeVehicleParts用于检测图片中车辆部件的位置以及名称。

前提条件

请确保您已开通[目标检测服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG。
- 图片分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为3秒。
- 图片大小：图片大小不超过10M。
- 图片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	RecognizeVehicleParts	要执行的操作。取值：RecognizeVehicleParts。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test/ant_ai/part_detection/07000008002016000514_00005-D6319603-9DD1-45ae-950E-FCA36EB6DFA7.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Elements	Array of Element		各个子元素的识别结果。
Boxes	List	336,1,635,57	部件位置框信息，【左上角点x坐标，左上角点y坐标，右上角点x坐标，右上角点y坐标】。
Score	Float	0.9929494857788086	部件类型对应的概率值。
Type	String	hood	部件类型。具体请参见type说明表格。
OriginShapes	List	480, 640	输入图片的原始尺寸【高度，宽度】。
RequestId	String	35771730-0FC5-4E7D-8C2F-2FADDFECA39	请求ID。

Type说明

名字	英文名字	部件编码	标注编码	是否是34部件	是否计划支持
前保险杠	front_bumper	PR01	11	是	是
后保险杠	rear_bumper	PR02	13	是	是
左前大灯	left_light	PR03	71	是	是
右前大灯	right_light	PR04	72	是	是
中网	grille	PR05	b11	是	是
中网	grille	PR05	B11	是	是
前机盖	hood	PR06	21	是	是

名字	英文名字	部件编码	标注编码	是否是34部件	是否计划支持
左前门	left_front_door	PR07	51	是	是
右前门	right_front_door	PR08	52	是	是
左后门	left_rear_door	PR09	53	是	是
右后门	right_rear_door	PR010	54	是	是
左前翼子板	left_front_wing	PR11	41	是	是
右前翼子板	right_front_wing	PR12	42	是	是
左后翼子板	left_rear_wing	PR13	43	是	是
右后翼子板	right_rear_wing	PR14	44	是	是
后机盖	decklid	PR015	23	是	是
前挡风玻璃	front_windshield	PR16	61	是	是
后挡风玻璃	rear_windshield	PR17	63	是	是
左后视镜	left_mirror	PR18	93	是	是
右后视镜	right_mirror	PR19	94	是	是
左尾灯	left_tail_light	PR20	73	是	是

名字	英文名字	部件编码	标注编码	是否是34部件	是否计划支持
右尾灯	right_tail_lig ht	PR21	74	是	是
左雾灯	left_foglight	PR22	b13	是	是
左雾灯	left_foglight	PR22	B13	是	是
右雾灯	right_foglig ht	PR23	b14	是	是
右雾灯	right_foglig ht	PR23	B14	是	是
格栅	grates	PR24	b12	是	是
格栅	grates	PR24	B12	是	是
右前车窗	right_front_ window	PR25	82	是	是
左前车窗	left_front_w indow	PR26	81	是	是
右后车窗	right_rear_w indow	PR27	84	是	是
左后车窗	right_tail_lig ht	PR28	83	是	是
右底大边	right_doorsil l	PR29	92	是	是
左底大边	left_doorsill	PR30	91	是	是
右前门把手	right_front_ doorknob	PR31	502	是	是

名字	英文名字	部件编码	标注编码	是否是34部件	是否计划支持
左前门把手	left_front_doorknob	PR32	501	是	是
左后门把手	left_rear_doorknob	PR34	503	是	是
右后门把手	right_tail_light	PR33	504	是	是
左前轮胎	left_front_tire	PR3601	31	是	是
左后轮胎	left_rear_tire	PR3602	33	是	是
右前轮胎	right_front_tire	PR3603	32	是	是
右后轮胎	right_tail_light	PR3604	34	是	是
轮胎	right_tail_light	PR36	3	是	是
右前轮眉	right_front_WheelBrow	PR1203	352	否	是
左前轮眉	left_front_WheelBrow	PR1103	351	否	是
右后轮眉	right_rear_WheelBrow	PR1403	354	否	是
左后轮眉	left_rear_WheelBrow	PR1303	353	否	是
左前钢圈	left_front_rim	PR3501	316	否	是

名字	英文名字	部件编码	标注编码	是否是34部件	是否计划支持
左后钢圈	left_rear_rin g	PR3502	336	否	是
右前钢圈	right_tail_lig ht	PR3503	326	否	是
右后钢圈	right_rear_ri ng	PR3504	346	否	是
钢圈	ring	PR35	306	否	是
左前门饰条	left_front_d oor_Panel	PR0701	6511	否	是
右前门饰条	right_front_ door_Panel	PR0801	6512	否	是
左后门饰条	left_rear_do or_Panel	PR0901	6513	否	是
右后门饰条	right_rear_d oor_Panel	PR1001	6514	否	是
左前门亮条	left_front_d oor_Wisp	PR0701	6521	否	是
右前门亮条	right_front_ door_Wisp	PR0801	6522	否	是
左后门亮条	left_rear_do or_Wisp	PR0901	6523	否	是
右后门亮条	right_rear_d oor_Wisp	PR1001	6524	否	是
左前门饰板	left_front_d oor_Plaque	PR0703	166	否	是

名字	英文名字	部件编码	标注编码	是否是34部件	是否计划支持
右前门饰板	right_front_door_Plaque	PR0803	266	否	是
左后门饰板	left_rear_door_Plaque	PR0903	366	否	是
右后门饰板	right_rear_door_Plaque	PR1003	466	否	是
前保险杠饰条	front_bumper_Panel	PR0101	1651	否	是
后保险杠饰条	rear_bumper_Panel	PR0201	3651	否	是
前保险杠亮条	front_bumper_Wisp	PR0102	1652	否	是
后保险杠亮条	rear_bumper_Wisp	PR0202	3652	否	是
前保险杠护板	front_bumper_Backplate	PR0103	6621	否	是
后保险杠护板	rear_bumper_Backplate	PR0203	6623	否	是
前保险杠导流板	front_bumper_Deflector	PR0104	6611	否	是
后保险杠导流板	rear_bumper_Deflector	PR0204	6613	否	是
后保反光板	BumperLight	PR0205	761	否	是
车顶	Roof	待定	25	否	否
A柱	Apillar	待定	26	否	否

名字	英文名字	部件编码	标注编码	是否是34部件	是否计划支持
挡泥板	Fender	待定	36	否	否
反光镜灯	MirrorLight	待定	762	否	否
翼子板灯	WingLight	待定	763	否	否
牌照灯	PlateLight	待定	764	否	否
车窗三角玻璃	TriangleWindow	待定	85	否	否
天窗	RoofWindow	待定	86	否	否
字标	WordMark	待定	67	否	否
牵引钩盖板	DragCover	待定	681	否	否
喷水嘴盖板	LightWaterCover	待定	682	否	否
电眼	ElectricEye	待定	683	否	否
油箱盖	TankCover	待定	684	否	否
导流板（无方向）	Deflector	待定	661	否	否
门把手（无方向）	Deflector	待定	50	否	否
车标	logo	待定	b16,B16	否	否
饰板（无方向）	Plaque	待定	66	否	否

名字	英文名字	部件编码	标注编码	是否是34部件	是否计划支持
保险杠（无方向）	Roof	待定	1	否	否
车牌	plate	待定	b17,B17	否	否
雾灯（无方向）	foglight	待定	b15,B15	否	否
亮条（无方向）	Wisp	待定	652	否	否
翼子板（无方向）	wing	待定	4	否	否
饰条（无方向）	Panel	待定	651	否	否
底大边（无方向）	doorsill	待定	9	否	否
门（无方向）	door	待定	5	否	否
挡风玻璃	windshield	待定	6	否	否
大灯（无方向）	light	待定	70	否	否
挡泥板	Fender	待定	36	否	否
轮眉（无方向）	WheelBrow	待定	35	否	否
看不出什么部件	无	待定	00	无	无

示例

请求示例

```
http(s)://objectdet.cn-shanghai.aliyuncs.com/?Action=RecognizeVehicleParts
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test/ant_ai/part_detection/07000008002
016000514_00005-D6319603-9DD1-45ae-950E-FCA36EB6DFA7.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>35771730-0FC5-4E7D-8C2F-2FADDFECAA39</RequestId>
<Data>
  <OriginShapes>480</OriginShapes>
  <OriginShapes>640</OriginShapes>
  <Elements>
    <Score>0.9929494857788086</Score>
    <Type>hood</Type>
    <Boxes>336</Boxes>
    <Boxes>1</Boxes>
    <Boxes>635</Boxes>
    <Boxes>57</Boxes>
  </Elements>
  <Elements>
    <Score>0.9916594624519348</Score>
    <Type>right_front_door</Type>
    <Boxes>0</Boxes>
    <Boxes>54</Boxes>
    <Boxes>348</Boxes>
    <Boxes>288</Boxes>
  </Elements>
  <Elements>
    <Score>0.9896328449249268</Score>
    <Type>right_front_wing</Type>
    <Boxes>311</Boxes>
    <Boxes>50</Boxes>
    <Boxes>636</Boxes>
    <Boxes>281</Boxes>
  </Elements>
  <Elements>
    <Score>0.9857422113418579</Score>
    <Type>right_front_tire</Type>
    <Boxes>406</Boxes>
    <Boxes>153</Boxes>
    <Boxes>610</Boxes>
    <Boxes>334</Boxes>
  </Elements>
  <Elements>
    <Score>0.9717473387718201</Score>
```

```
<Type>right_mirror</Type>
<Boxes>134</Boxes>
<Boxes>22</Boxes>
<Boxes>240</Boxes>
<Boxes>103</Boxes>
</Elements>
<Elements>
  <Score>0.954562246799469</Score>
  <Type>doorknob</Type>
  <Boxes>3</Boxes>
  <Boxes>154</Boxes>
  <Boxes>59</Boxes>
  <Boxes>175</Boxes>
</Elements>
<Elements>
  <Score>0.9486961960792542</Score>
  <Type>right_front_window</Type>
  <Boxes>8</Boxes>
  <Boxes>1</Boxes>
  <Boxes>195</Boxes>
  <Boxes>95</Boxes>
</Elements>
<Elements>
  <Score>0.9402061700820923</Score>
  <Type>front_windshield</Type>
  <Boxes>253</Boxes>
  <Boxes>1</Boxes>
  <Boxes>395</Boxes>
  <Boxes>46</Boxes>
</Elements>
<Elements>
  <Score>0.8994627594947815</Score>
  <Type>right_doorsill</Type>
  <Boxes>4</Boxes>
  <Boxes>274</Boxes>
  <Boxes>376</Boxes>
  <Boxes>301</Boxes>
</Elements>
<Elements>
  <Score>0.8943594694137573</Score>
  <Type>right_rear_door</Type>
```

```
<type>right_rear_door</type>
<Boxes>2</Boxes>
<Boxes>183</Boxes>
<Boxes>69</Boxes>
<Boxes>293</Boxes>
</Elements>
</Data>
```

JSON 格式

```
{
  "RequestId": "35771730-0FC5-4E7D-8C2F-2FADDFECA39",
  "Data": {
    "OriginShapes": [
      480,
      640
    ],
    "Elements": [
      {
        "Score": "0.9929494857788086",
        "Type": "hood",
        "Boxes": [
          336,
          1,
          635,
          57
        ]
      },
      {
        "Score": "0.9916594624519348",
        "Type": "right_front_door",
        "Boxes": [
          0,
          54,
          348,
          288
        ]
      },
      {
        "Score": "0.9896328449249268",
        "Type": "right_front_wing",

```

```
"Boxes": [
  311,
  50,
  636,
  281
]
},
{
  "Score": "0.9857422113418579",
  "Type": "right_front_tire",
  "Boxes": [
    406,
    153,
    610,
    334
  ]
},
{
  "Score": "0.9717473387718201",
  "Type": "right_mirror",
  "Boxes": [
    134,
    22,
    240,
    103
  ]
},
{
  "Score": "0.954562246799469",
  "Type": "doorknob",
  "Boxes": [
    3,
    154,
    59,
    175
  ]
},
{
  "Score": "0.9486961960792542",
  "Type": "right_front_window",
  "Boxes": [
```

```
8,
1,
195,
95
]
},
{
  "Score": "0.9402061700820923",
  "Type": "front_windshield",
  "Boxes": [
    253,
    1,
    395,
    46
  ]
},
{
  "Score": "0.8994627594947815",
  "Type": "right_doorsill",
  "Boxes": [
    4,
    274,
    376,
    301
  ]
},
{
  "Score": "0.8943594694137573",
  "Type": "right_rear_door",
  "Boxes": [
    2,
    183,
    69,
    293
  ]
}
]
}
}
```

错误码

关于车辆部件识别的错误码，详情请参见[常见错误码](#)。

SDK参考

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

安全声明

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

2.3. 车险图片分类

本文档为您介绍车险图片分类ClassifyVehicleInsurance的语法及示例。

功能描述

ClassifyVehicleInsurance对输入的车险图片进行分类。

前提条件

请确保您已开通[目标检测服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG。
- 图片分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- 图片大小：图片大小不超过10M。
- 图片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	ClassifyVehicleInsurance	要执行的操作。取值：ClassifyVehicleInsurance。

名称	类型	是否必选	示例值	描述
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test/ant_ai/vehicle_insurance/Z114C005969E_00004-8DA3D707-8E1F-48f7-BF3C-B997220C0A1A.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Labels	Array of Label		输出分类结果。
Name	String	others	label名称。枚举类型。枚举值包括： <ul style="list-style-type: none"> • others：其他 • detail：细节图 • component：汽车部件图 • vin：汽车vin码 • people：人物 • motor：发动机 • semi-car：半车图 • panoramic：全车图 • license：行驶证 • CT-scan：CT-扫描 • truck：卡车 • disassembly：拆解件 • scene：现场图
Score	Float	0.0023	分类结果对应的概率值。
Threshold	Float	0	阈值。根据阈值控制输出label，取值范围0~1。
RequestId	String	878EB16C-87C6-4C8B-897D-F68036F8AC97	请求ID。

示例

请求示例

```
http(s)://objectdet.cn-shanghai.aliyuncs.com/?Action=ClassifyVehicleInsurance
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test/ant_ai/vehicle_insurance/Z114C005
969E_00004-8DA3D707-8E1F-48f7-BF3C-B997220C0A1A.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>878EB16C-87C6-4C8B-897D-F68036F8AC97</RequestId>
<Data>
  <Labels>
    <Score>0.0023</Score>
    <Name>others</Name>
  </Labels>
  <Labels>
    <Score>0</Score>
    <Name>detail</Name>
  </Labels>
  <Labels>
    <Score>0.0001</Score>
    <Name>component</Name>
  </Labels>
  <Labels>
    <Score>0</Score>
    <Name>vin</Name>
  </Labels>
  <Labels>
    <Score>0.0001</Score>
    <Name>people</Name>
  </Labels>
  <Labels>
    <Score>0</Score>
    <Name>motor</Name>
  </Labels>
  <Labels>
    <Score>0</Score>
    <Name>semi-car</Name>
  </Labels>
```

```
<Labels>
  <Score>0.2215</Score>
  <Name>panoramic</Name>
</Labels>
<Labels>
  <Score>0</Score>
  <Name>license</Name>
</Labels>
<Labels>
  <Score>0</Score>
  <Name>CT-scan</Name>
</Labels>
<Labels>
  <Score>0.0001</Score>
  <Name>truck</Name>
</Labels>
<Labels>
  <Score>0.0001</Score>
  <Name>disassembly</Name>
</Labels>
<Labels>
  <Score>0.7759</Score>
  <Name>scene</Name>
</Labels>
<Threshold>0</Threshold>
</Data>
```

JSON 格式

```
{
  "RequestId": "878EB16C-87C6-4C8B-897D-F68036F8AC97",
  "Data": {
    "Labels": [
      {
        "Score": 0.0023,
        "Name": "others"
      },
      {
        "Score": 0,
        "Name": "detail"
      },
    ],
  },
}
```

```
"
{
  "Score": 0.0001,
  "Name": "component"
},
{
  "Score": 0,
  "Name": "vin"
},
{
  "Score": 0.0001,
  "Name": "people"
},
{
  "Score": 0,
  "Name": "motor"
},
{
  "Score": 0,
  "Name": "semi-car"
},
{
  "Score": 0.2215,
  "Name": "panoramic"
},
{
  "Score": 0,
  "Name": "license"
},
{
  "Score": 0,
  "Name": "CT-scan"
},
{
  "Score": 0.0001,
  "Name": "truck"
},
{
  "Score": 0.0001,
  "Name": "disassembly"
},
}
```

```
{
  "Score": 0.7759,
  "Name": "scene"
},
"Threshold": 0
}
```

错误码

关于车险图片分类的错误码，详情请参见[常见错误码](#)。

SDK参考

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

安全声明

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

2.4. 仪表盘识别

本文档为您介绍仪表盘识别RecognizeVehicleDashboard的语法及示例。

功能描述

RecognizeVehicleDashboard用于识别仪表盘上故障灯等信息。当前可以识别的故障灯类别包括：1（TA01 门窗未关闭）、2（TA02 安全带未系）、3（TA03 霜冻提示）、4（TG01 定速巡航）、5（TG02 雾灯打开）、6（TG03 远光打开）、7（TG04 坡道辅助）、8（TG05 玻璃加热）、9（TG06 制动提示灯）、10（FZ01 燃油不足）、11（FZ02 胎压过低）、12（FZ03 清洗液不足）、13（FZ04 电瓶缺电）、14（FW01 需保养）、15（FW02 水温异常）、16（FW03 机油不足）、17（FW04 冷却液不足）、18（FW05 刹车片磨损）、19（FW06 安全气囊故障）、20（FW07 发动机故障）、21（QT00 其他）。

前提条件

请确保您已开通[目标检测服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG、PNG。
- 图片分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- 图片大小：图片的大小不超过10M。
- 图片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	RecognizeVehicleDashboard	要执行的操作。取值：RecognizeVehicleDashboard。
ImageUrl	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test/ant_ai/vehicle_dashboard/test_dashboard.png	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Elements	Array of Element		识别结果。
Boxes	List	"692.4900512695312", "423.0277099609375", "709.9772338867188", "436.11529541015625"	识别故障灯位置。格式为 [x1,y1,x2,y2]，x1和y1为矩形框左上角点坐标，x2和y2为右下角点坐标。
ClassName	String	FZ04 电瓶缺电	识别故障灯名称，unicode编码。
Label	String	13	识别故障灯类别。例如13表示电瓶缺电。
Score	Float	0.9990643858909607	label分数。
RequestId	String	3BDA7297-2515-4C9C-99EA-C5950B906BCF	请求ID。

示例

请求示例

```
http(s)://objectdet.cn-shanghai.aliyuncs.com/?Action=RecognizeVehicleDashboard
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test/ant_ai/vehicle_dashboard/test_dash
board.png
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>3BDA7297-2515-4C9C-99EA-C5950B906BCF</RequestId>
<Data>
  <Elements>
    <Score>0.9990643858909607</Score>
    <ClassName>FZ04 电瓶缺电</ClassName>
    <Label>13</Label>
    <Boxes>692.4900512695312</Boxes>
    <Boxes>423.0277099609375</Boxes>
    <Boxes>709.9772338867188</Boxes>
    <Boxes>436.11529541015625</Boxes>
  </Elements>
  <Elements>
    <Score>0.9948421120643616</Score>
    <ClassName>FW07 发动机故障</ClassName>
    <Label>20</Label>
    <Boxes>262.644287109375</Boxes>
    <Boxes>419.08294677734375</Boxes>
    <Boxes>282.2294921875</Boxes>
    <Boxes>432.89495849609375</Boxes>
  </Elements>
  <Elements>
    <Score>0.9942097663879395</Score>
    <ClassName>TG06 制动提示灯</ClassName>
    <Label>9</Label>
    <Boxes>280.72662353515625</Boxes>
    <Boxes>405.62225341796875</Boxes>
    <Boxes>301.68280029296875</Boxes>
    <Boxes>421.94085693359375</Boxes>
  </Elements>
  <Elements>
    <Score>0.9878326654434204</Score>
    <ClassName>TA02 安全带未系</ClassName>
```

```
<Label>2</Label>
<Boxes>690.0284423828125</Boxes>
<Boxes>374.8208923339844</Boxes>
<Boxes>705.72021484375</Boxes>
<Boxes>394.5099182128906</Boxes>
</Elements>
<Elements>
  <Score>0.957288920879364</Score>
  <ClassName>FZ01 燃油不足</ClassName>
  <Label>10</Label>
  <Boxes>716.0230712890625</Boxes>
  <Boxes>426.26654052734375</Boxes>
  <Boxes>730.5159912109375</Boxes>
  <Boxes>445.77960205078125</Boxes>
</Elements>
<Elements>
  <Score>0.9438304901123047</Score>
  <ClassName>TG04 坡道辅助</ClassName>
  <Label>7</Label>
  <Boxes>688.2109375</Boxes>
  <Boxes>353.2281188964844</Boxes>
  <Boxes>708.463134765625</Boxes>
  <Boxes>368.4597473144531</Boxes>
</Elements>
</Data>
```

JSON 格式

```
{
  "RequestId": "3BDA7297-2515-4C9C-99EA-C5950B906BCF",
  "Data": {
    "Elements": [
      {
        "Score": "0.9990643858909607",
        "ClassName": "FZ04 电瓶缺电",
        "Label": 13,
        "Boxes": [
          "692.4900512695312",
          "423.0277099609375",
          "709.9772338867188",
          "436.11529541015625"
        ]
      }
    ]
  }
}
```

```
]
},
{
  "Score": "0.9948421120643616",
  "ClassName": "FW07 发动机故障",
  "Label": 20,
  "Boxes": [
    "262.644287109375",
    "419.08294677734375",
    "282.2294921875",
    "432.89495849609375"
  ]
},
{
  "Score": "0.9942097663879395",
  "ClassName": "TG06 制动提示灯",
  "Label": 9,
  "Boxes": [
    "280.72662353515625",
    "405.62225341796875",
    "301.68280029296875",
    "421.94085693359375"
  ]
},
{
  "Score": "0.9878326654434204",
  "ClassName": "TA02 安全带未系",
  "Label": 2,
  "Boxes": [
    "690.0284423828125",
    "374.8208923339844",
    "705.72021484375",
    "394.5099182128906"
  ]
},
{
  "Score": "0.957288920879364",
  "ClassName": "FZ01 燃油不足",
  "Label": 10,
  "Boxes": [
    "716.0230712890625",
```



```
"426.26654052734375",
"730.5159912109375",
"445.77960205078125"
]
},
{
  "Score": "0.9438304901123047",
  "ClassName": "TG04 坡道辅助",
  "Label": 7,
  "Boxes": [
    "688.2109375",
    "353.2281188964844",
    "708.463134765625",
    "368.4597473144531"
  ]
}
]
}
}
```

错误码

关于仪表盘识别的错误码，详情请参见[常见错误码](#)。

SDK参考

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

安全声明

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

2.5. 车辆损伤识别

本文档介绍车辆损伤识别RecognizeVehicleDamage的语法及示例。

功能描述

RecognizeVehicleDamage检测图片中车辆损伤的位置以及类型。

前提条件

请确保您已开通[目标检测服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG。

- 图片分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- 图片大小：图片大小不超过10M。
- 图片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	RecognizeVehicleDamage	要执行的操作。取值：RecognizeVehicleDamage。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test/ant_ai/damage_rfcn/ZZAzQ6e50swAAAAAAAAAAAAA BjARcnAQ.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Elements	Array of Element		各个子元素的识别结果。
Boxes	List	173,178,277,259	部件位置框信息，形式为【左上角点x坐标，右下角点y坐标，右上角点x坐标，左下角点y坐标】。
Score	Float	0.683465	损伤类型对应的概率值。
Scores	List	0.683465	对应所有15种损伤类型的损伤概率。

名称	类型	示例值	描述
Type	String	1	<p>损伤类型ID。具体类型如下所示。</p> <ul style="list-style-type: none"> • 1: 轻微刮擦 • 2: 重度刮擦 • 3: 轻度变形 • 4: 中度变形 • 5: 重度变形 • 6: crack破损孔洞 • 7: 翼子板和大灯缝隙 • 8: 翼子板保险杠缝隙 • 9: 大灯轻微刮擦 • 10: 大灯重度刮擦 • 11: 大灯破损 • 12: 后视镜轻微刮擦 • 13: 后视镜玻璃破损 • 14: 后视镜脱落 • 15: 挡风玻璃破损
RequestId	String	91FCB447-807D-4839-A5F0-E0A79D6B33C6	请求ID。

示例

请求示例

```
http(s)://objectdet.cn-shanghai.aliyuncs.com/?Action=RecognizeVehicleDamage
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test/ant_ai/damage_rfcn/ZZAzQ6e50sw
AAAAAAAAAAAAABjARcnAQ.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>91FCB447-807D-4839-A5F0-E0A79D6B33C6</RequestId>
<Data>
  <Elements>
    <Score>0.683465</Score>
    <Type>1</Type>
    <Scores>0.683465</Scores>
    <Scores>0</Scores>
    <Scores>0</Scores>
    <Scores>0</Scores>
```



```
0,  
0  
],  
"Boxes": [  
  788,  
  158,  
  884,  
  189  
]  
},  
{  
  "Score": 0.549272,  
  "Type": "1",  
  "Scores": [  
    0.549272,  
    0,  
    0,  
    0,  
    0,  
    0,  
    0,  
    0,  
    0,  
    0,  
    0,  
    0,  
    0,  
    0,  
    0,  
    0  
  ],  
  "Boxes": [  
    667,  
    213,  
    846,  
    242  
  ]  
},  
{  
  "Score": 0.52864,  
  "Type": "1",  
  "Scores": [  

```



```
0,  
0  
],  
"Boxes": [  
692,  
256,  
761,  
285  
]  
},  
{  
"Score": 0.323798,  
"Type": "1",  
"Scores": [  
0.323798,  
0,  
0,  
0,  
0,  
0,  
0,  
0,  
0,  
0,  
0,  
0,  
0,  
0,  
0,  
0,  
0,  
0,  
0,  
0  
],  
"Boxes": [  
684,  
221,  
827,  
296  
]  
},  
{  
"Score": 0.634771,  
"Type": "2",  
"Scores": [  

```



```
0
],
"Boxes": [
  562,
  364,
  621,
  409
]
},
{
  "Score": 0.603638,
  "Type": "2",
  "Scores": [
    0,
    0.603638,
    0,
    0,
    0,
    0,
    0,
    0,
    0,
    0,
    0,
    0,
    0,
    0,
    0,
    0,
    0,
    0,
    0,
    0
  ],
  "Boxes": [
    379,
    302,
    462,
    352
  ]
},
{
  "Score": 0.578584,
  "Type": "2",
  "Scores": [
```


3. 车辆维修

3.1. 生成车辆维修方案

本文档介绍生成车辆维修方案GenerateVehicleRepairPlan的语法及示例。

功能描述

GenerateVehicleRepairPlan可以识别车辆受损照片，并根据受损情况生成车辆维修方案，然后您通过调用[查询车辆维修方案](#)接口获取维修方案及费用结果。

前提条件

请确保您已开通[目标检测服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG、PNG。
- 图片分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- 图片大小：图片大小不超过2M，最多可以输入5张图片。
- 图片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	GenerateVehicleRepairPlan	要执行的操作。取值：GenerateVehicleRepairPlan。
DamageImageList.N.CreateTimeStamp	String	是	1589784388081	受损照片拍摄时间。格式为13位时间戳，例如1589784388081。
DamageImageList.N.ImageUrl	String	是	https://img.alicdn.com/tfs/TB1mKnME1H2gK0jSZJnXXaT1FXa-1000-750.jpg	受损照片的CDN地址，任意CDN图片均可。

返回数据

名称	类型	示例值	描述
Code	String	无	请求发生错误时返回的错误码。

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
TaskId	String	f393e279-ed37-43fb-86d9-fa3c8ca8f066	任务ID。
ErrorMessage	String	无	错误码对应的错误描述信息。
HttpCode	Integer	400	HTTP状态码。
RequestId	String	42dc1b1b-1957-4d6d-acef-4b98884252f4	请求ID。
Success	Boolean	true	本次请求是否成功。

示例

请求示例

```
http(s)://[Endpoint]/?Action=GenerateVehicleRepairPlan
&DamageImageList.1.CreateTimeStamp=1589784388081
&DamageImageList.1.ImageUrl=https://img.alicdn.com/tfs/TB1mKnME1H2gK0jSZjnXXaT1FXa-1000-750.jpg
pg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>42dc1b1b-1957-4d6d-acef-4b98884252f4</RequestId>
<Data>
  <TaskId>f393e279-ed37-43fb-86d9-fa3c8ca8f066</TaskId>
</Data>
<ErrorMessage/>
<Code/>
<Success>true</Success>
```

JSON 格式

```
{
  "RequestId": "42dc1b1b-1957-4d6d-acef-4b98884252f4",
  "Data": {
    "TaskId": "f393e279-ed37-43fb-86d9-fa3c8ca8f066"
  },
  "ErrorMessage": "",
  "Code": "",
  "Success": true
}
```

错误码

关于车辆维修方案的错误码，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. 查询车辆维修方案及费用

本文档为您介绍查询车辆维修方案及费用GetVehicleRepairPlan的语法及示例。

功能描述

您使用[GenerateVehicleRepairPlan](#)接口生成车辆维修方案后，可以通过GetVehicleRepairPlan接口查询车辆具体的维修方案及维修费用。

前提条件

- 请确保您已开通[目标检测服务](#)，若未开通服务请[立即开通](#)。
- 已经通过[GenerateVehicleRepairPlan](#)接口生成了车辆维修方案。

输入限制

- 图片格式：JPEG、JPG、PNG。
- 图片分辨率：不限制图片分辨率，但图片分辨率太高可能会导致API识别超时，超时时间为5秒。
- 图片大小：大小不超过2M。
- 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	GetVehicleRepairPlan	要执行的操作。取值：GetVehicleRepairPlan。
TaskId	String	是	f393e279-ed37-43fb-86d9-fa3c8ca8f066	GenerateVehicleRepairPlan接口返回的车辆维修方案任务ID。
VinCodeImage	String	是	https://img.alicdn.com/tfs/TB1wz2GE.H1gK0jSZSyXXXtlpXa-414-259.jpg	受损车辆VIN码截图。
CarNumberImage	String	否	https://img.alicdn.com/tfs/TB1LV9UC1L2gK0jSZPhXXahvXXa-602-440.jpg	受损车辆车牌号截图。

返回数据

名称	类型	示例值	描述
Code	String	无	请求发生错误时返回的错误码。
Data	Struct		返回的结果数据内容。
FrameNo	String	LFV2A21K0G4053021	识别出的车辆VIN码。
RepairParts	Array of RepairItems		维修方案数据。
GarageType	String	4S	维修厂类型。
OeMatch	Boolean	false	OE码是否能匹配配件。
OutStandardPartsId	String	A1601001	合作机构标准配件ID。
OutStandardPartsName	String	后翼子板（右）	合作机构标准配件名称。
PartNameMatch	Boolean	true	部件名称与部件ID是否能正确匹配。

名称	类型	示例值	描述
PartsStdCode	String	PR14	统一配件码。
PartsStdName	String	后翼子板（右）	配件名称。
RelationType	String	non	关联分析结果类型。
RepairFee	String	62400	维修费用。
RepairType	String	R1	维修方案代码。
RepairTypeNa me	String	喷漆	维修方案名称。
ErrorMessage	String	无	错误码对应的错误描述信息。
HttpCode	Integer	400	HTTP状态码。
RequestId	String	8acad38e-b502-409e-ac9c-b24f167ead00	请求ID。
Success	Boolean	true	本次请求是否成功。

示例

请求示例

```
http(s)://[Endpoint]/?Action=GetVehicleRepairPlan
&TaskId=f393e279-ed37-43fb-86d9-fa3c8ca8f066
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>8acad38e-b502-409e-ac9c-b24f167ead00</RequestId>
<Data>
  <FrameNo>LFV2A21K0G4053021</FrameNo>
  <RepairParts>
    <PartNameMatch>true</PartNameMatch>
    <PartsStdCode>PR14</PartsStdCode>
```

```
<OutStandardPartsName>后翼子板（右）</OutStandardPartsName>
<RepairType>R1</RepairType>
<RelationType>non</RelationType>
<RepairFee>62400</RepairFee>
<GarageType>4S</GarageType>
<PartsStdName>后翼子板（右）</PartsStdName>
<OutStandardPartsId>A1601001</OutStandardPartsId>
<RepairTypeName>喷漆</RepairTypeName>
<OeMatch>>false</OeMatch>
</RepairParts>
<RepairParts>
  <PartNameMatch>>true</PartNameMatch>
  <PartsStdCode>PR10</PartsStdCode>
  <OutStandardPartsName>后门壳（右）</OutStandardPartsName>
  <RepairType>R1</RepairType>
  <RelationType>non</RelationType>
  <RepairFee>62400</RepairFee>
  <GarageType>4S</GarageType>
  <PartsStdName>后门壳（右）</PartsStdName>
  <OutStandardPartsId>A0901003</OutStandardPartsId>
  <RepairTypeName>喷漆</RepairTypeName>
  <OeMatch>>false</OeMatch>
</RepairParts>
<RepairParts>
  <PartNameMatch>>true</PartNameMatch>
  <PartsStdCode>PR14</PartsStdCode>
  <OutStandardPartsName>后翼子板（右）</OutStandardPartsName>
  <RepairType>R1</RepairType>
  <RelationType>non</RelationType>
  <RepairFee>41600</RepairFee>
  <GarageType>GENERAL</GarageType>
  <PartsStdName>后翼子板（右）</PartsStdName>
  <OutStandardPartsId>A1601001</OutStandardPartsId>
  <RepairTypeName>喷漆</RepairTypeName>
  <OeMatch>>false</OeMatch>
</RepairParts>
<RepairParts>
  <PartNameMatch>>true</PartNameMatch>
  <PartsStdCode>PR10</PartsStdCode>
  <OutStandardPartsName>后门壳（右）</OutStandardPartsName>
  <RepairType>R1</RepairType>
```

```
<RelationType>non</RelationType>
<RepairFee>41600</RepairFee>
<GarageType>GENERAL</GarageType>
<PartsStdName>后门壳（右）</PartsStdName>
<OutStandardPartsId>A0901003</OutStandardPartsId>
<RepairTypeName>喷漆</RepairTypeName>
<OeMatch>>false</OeMatch>
</RepairParts>
</Data>
<ErrorMessage/>
<Code/>
<Success>>true</Success>
```

JSON 格式

```
{
  "RequestId": "8acad38e-b502-409e-ac9c-b24f167ead00",
  "Data": {
    "FrameNo": "LFV2A21K0G4053021",
    "RepairParts": [
      {
        "PartNameMatch": true,
        "PartsStdCode": "PR14",
        "OutStandardPartsName": "后翼子板（右）",
        "RepairType": "R1",
        "RelationType": "non",
        "RepairFee": "62400",
        "GarageType": "4S",
        "PartsStdName": "后翼子板（右）",
        "OutStandardPartsId": "A1601001",
        "RepairTypeName": "喷漆",
        "OeMatch": false
      },
      {
        "PartNameMatch": true,
        "PartsStdCode": "PR10",
        "OutStandardPartsName": "后门壳（右）",
        "RepairType": "R1",
        "RelationType": "non",
        "RepairFee": "62400",
        "GarageType": "4S",
```



```
"PartsStdName": "后门壳 (右) ",
"OutStandardPartsId": "A0901003",
"RepairTypeName": "喷漆",
"OeMatch": false
},
{
  "PartNameMatch": true,
  "PartsStdCode": "PR14",
  "OutStandardPartsName": "后翼子板 (右) ",
  "RepairType": "R1",
  "RelationType": "non",
  "RepairFee": "41600",
  "GarageType": "GENERAL",
  "PartsStdName": "后翼子板 (右) ",
  "OutStandardPartsId": "A1601001",
  "RepairTypeName": "喷漆",
  "OeMatch": false
},
{
  "PartNameMatch": true,
  "PartsStdCode": "PR10",
  "OutStandardPartsName": "后门壳 (右) ",
  "RepairType": "R1",
  "RelationType": "non",
  "RepairFee": "41600",
  "GarageType": "GENERAL",
  "PartsStdName": "后门壳 (右) ",
  "OutStandardPartsId": "A0901003",
  "RepairTypeName": "喷漆",
  "OeMatch": false
}
]
},
"ErrorMessage": "",
"Code": "",
"Success": true
}
```

错误码

关于查询车辆维修方案的错误码，详情请参见[常见错误码](#)。

SDK参考

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

安全声明

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

4.通用检测

4.1. 主体检测

本文档介绍主体检测DetectMainBody的语法及示例。

功能描述

DetectMainBody对输入图片进行检测，输出主体定位信息。

前提条件

请确保您已开通[目标检测服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：PNG（不支持8位、16位和64位的PNG）、JPG、JPEG、BMP。
- 图像大小：图片大小不超过3M。
- 图像分辨率：图片分辨率小于1280×1280像素。
- 图片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	DetectMainBody	要执行的操作。取值：DetectMainBody。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/5%E4%B8%BB%E4%BD%93%E6%A3%80%E6%B5%8B.jpg	图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。

名称	类型	示例值	描述
Location	Struct		图像定位信息。
Height	Integer	320	元素高度。
Width	Integer	583	元素宽度。
X	Integer	28	元素x坐标。
Y	Integer	20	元素y坐标。
RequestId	String	2E59C333-5480-4231-A8AB-BEE1001EA7FE	请求ID。

示例

请求示例

```
http(s)://objectdet.cn-shanghai.aliyuncs.com/?Action=DetectMainBody
&ImageURL=https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/tiankong/5%E4%B8%BB%E4%BD%93%E6%A3%80%E6%B5%8B.jpg
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RequestId>2E59C333-5480-4231-A8AB-BEE1001EA7FE</RequestId>
<Data>
  <Location>
    <X>28</X>
    <Y>20</Y>
    <Height>320</Height>
    <Width>583</Width>
  </Location>
</Data>
```

JSON 格式

```
{
  "RequestId": "2E59C333-5480-4231-A8AB-BEE1001EA7FE",
  "Data": {
    "Location": {
      "X": 28,
      "Y": 20,
      "Height": 320,
      "Width": 583
    }
  }
}
```

错误码

关于主体检测的错误码，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. 白底图检测

本文介绍白底图检测DetectWhiteBaseImage的语法及示例。

功能描述

DetectWhiteBaseImage用于检测图片背景是否为白底图。

前提条件

请确保您已开通[目标检测服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图像大小：图片大小不超过3M。
- 图像分辨率：分辨率要求大于32×32像素，且小于1280×1280像素。
- 图像格式：JPG、JPEG、PNG。
- 图片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	DetectWhiteBaseImage	要执行的操作。取值：DetectWhiteBaseImage。
ImageURL	String	是	https://viapi-test.oss-cn-shanghai.aliyuncs.com/test-team/whiteBase.jpg	输入图片URL地址。当前仅支持上海地域的OSS链接，如何生成URL请参见 生成URL 。

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Elements	Array of Element		返回判断结果。
WhiteBase	Integer	0	是否白底图。 <ul style="list-style-type: none"> 0: 非白底 1: 白底
RequestId	String	UUID	请求ID。

示例

请求示例

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

正常返回示例

XML 格式

```
<RequestId>UUID</RequestId>
<Data>
  <Elements>
    <WhiteBase>0</WhiteBase>
  </Elements>
</Data>
```

JSON 格式

```
{
  "RequestId": "UUID",
  "Data": {
    "Elements": [
      {
        "WhiteBase": 0
      }
    ]
  }
}
```

错误码

关于白底图检测的错误码，详情请参见[常见错误码](#)。

SDK参考

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

安全声明

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

4.3. 透明图检测

本文介绍透明图检测DetectTransparentImage的语法及示例。

功能描述

DetectTransparentImage用于检测图片背景是否为透明图。

前提条件

请确保您已开通[目标检测服务](#)，若未开通服务请[立即开通](#)。

调试

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

请求参数

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

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

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Elements	Array of Element		返回判断结果。
TransparentImage	Integer	0	是否透明图。 <ul style="list-style-type: none"> 0: 非透明图 1: 透明图
RequestId	String	UUID	请求ID。

示例

请求示例

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

正常返回示例

XML 格式


```
<RequestId>UUID</RequestId>
<Data>
  <Elements>
    <TransparentImage>0</TransparentImage>
  </Elements>
</Data>
```

JSON 格式

```
{
  "RequestId": "UUID",
  "Data": {
    "Elements": [
      {
        "TransparentImage": 0
      }
    ]
  }
}
```

错误码

关于透明图检测的错误码，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。

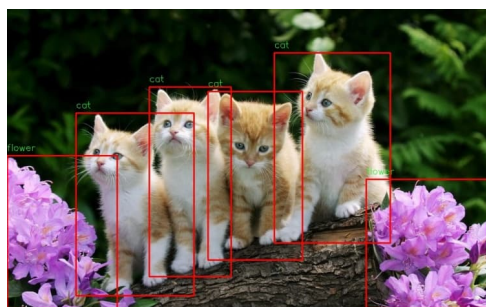
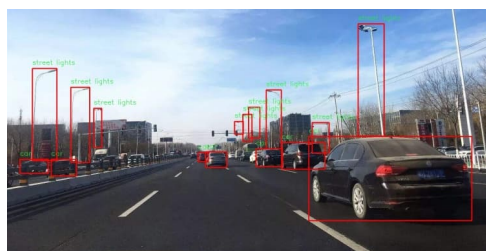
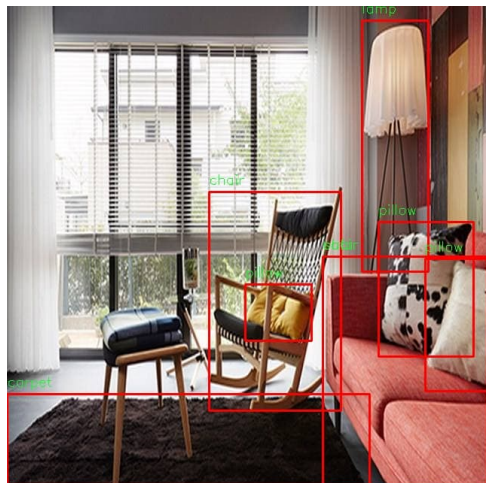
4.4. 物体检测

本文介绍物体检测DetectObject的语法及示例。

功能描述

DetectObject可以检测输入图像中的物体。

示例图如下。



前提条件

请确保您已开通[目标检测服务](#)，若未开通服务请[立即开通](#)。

输入限制

- 图片格式：JPEG、JPG、PNG、BMP。
- 图片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	DetectObject	要执行的操作。取值：DetectObject。

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

返回数据

名称	类型	示例值	描述
Data	Struct		返回的结果数据内容。
Elements	Array of Element		检测框数据集合。
Boxes	List	[468, 238, 531, 299]	检测框坐标，格式为 [left, top, right, bottom]。
Score	Float	0.266	检测到物体的置信度，取值范围为0~1。
Type	String	chair	检测框对应的物体名称。具体类型如下所示。
Height	Integer	300	图片的高度。
Width	Integer	533	图片的宽度。
RequestId	String	6EF97B44-2763-4EAD-8737-FB9F5EE25FE2	请求ID。

Type类型包括：

human (人体)、sneakers (胶底运动鞋)、chair (椅子)、hat (帽子)、lamp (灯)、cabinet/shelf (橱柜/架子)、car (汽车)、glasses (眼镜)、picture/frame (照片/图画)、street lights (街灯)、helmet (头盔)、pillow (枕头)、glove (手套)、potted plant (盆栽植物)、flower (花)、monitor (显示屏)、plants pot/vase (花盆)、boots (靴子)、umbrella (伞)、boat (小船)、flag (旗帜)、speaker (扬声器/话筒)、trash bin/can (垃圾桶)、backpack (双肩背包)、sofa (沙发)、belt (腰带)、carpet (地毯)、coffee table (咖啡桌/茶几)、tie (领带)、bed (床)、traffic light (红绿灯)、necklace (项链)、mirror (镜子)、bicycle (自行车)、watch (手表)、horse (马)、traffic sign (交通标志)、stuffed animal (填充玩具动物)、motorbike/motorcycle (摩托车)、wild bird (鸟)、laptop (笔记本电脑)、cow (奶牛)、clock (时钟)、bus (公共汽车)、nightstand (床头柜)、sheep (绵羊)、traffic cone (锥形交通路标)、keyboard (键盘)、hockey stick (曲棍球球棍)、fan (电扇)、dog (狗)、blackboard/whiteboard (白板/黑板)、mouse (鼠标)、telephone (电话)、airplane (飞机)、skis (滑雪板)、soccer (英式足球)、combine with glove (棒球手套)、train (火车)、tent (帐篷)、sailboat (帆船)、kite (风筝)、computer box (计算机主机机箱)、elephant (大象)、stroller (折叠式婴儿车)、baseball bat (棒球棒)、skateboard (溜冰板)、surfboard (冲浪板)、cat (猫)、zebra (斑马)、sports car (跑车)、giraffe (长颈鹿)、radiator (散热器)、tennis racket (网球拍)、skating and skiing shoes (溜冰鞋)、baseball (棒球)、american football (美式橄榄球)、basketball (篮球)、printer (打印机)、fire hydrant (消防栓)、projector (投影仪)、fire extinguisher (灭火器)、tennis ball (网球)、frisbee (飞盘)、fire truck (消防车)、helicopter (直升飞机)、carriage (四轮马车)、bear (熊)、globe (地球仪)、volleyball (排球)。

示例

请求示例

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

正常返回示例

XML 格式

```
<RequestId>6EF97B44-2763-4EAD-8737-FB9F5EE25FE2</RequestId>
<Data>
  <Height>300</Height>
  <Elements>
    <Score>0.266</Score>
    <Type>chair</Type>
    <Boxes>468</Boxes>
    <Boxes>238</Boxes>
    <Boxes>531</Boxes>
    <Boxes>299</Boxes>
  </Elements>
  <Elements>
    <Score>0.213</Score>
    <Type>chair</Type>
```

```
<Boxes>452</Boxes>
<Boxes>168</Boxes>
<Boxes>531</Boxes>
<Boxes>233</Boxes>
</Elements>
<Elements>
  <Score>0.308</Score>
  <Type>picture/frame</Type>
  <Boxes>487</Boxes>
  <Boxes>44</Boxes>
  <Boxes>501</Boxes>
  <Boxes>96</Boxes>
</Elements>
<Elements>
  <Score>0.257</Score>
  <Type>picture/frame</Type>
  <Boxes>477</Boxes>
  <Boxes>67</Boxes>
  <Boxes>484</Boxes>
  <Boxes>98</Boxes>
</Elements>
<Elements>
  <Score>0.201</Score>
  <Type>picture/frame</Type>
  <Boxes>503</Boxes>
  <Boxes>26</Boxes>
  <Boxes>524</Boxes>
  <Boxes>85</Boxes>
</Elements>
<Elements>
  <Score>0.515</Score>
  <Type>pillow</Type>
  <Boxes>161</Boxes>
  <Boxes>141</Boxes>
  <Boxes>181</Boxes>
  <Boxes>158</Boxes>
</Elements>
<Elements>
  <Score>0.293</Score>
  <Type>pillow</Type>
  <Boxes>473</Boxes>
```

```
<Boxes>475</Boxes>
<Boxes>143</Boxes>
<Boxes>496</Boxes>
<Boxes>157</Boxes>
</Elements>
<Elements>
  <Score>0.214</Score>
  <Type>pillow</Type>
  <Boxes>241</Boxes>
  <Boxes>145</Boxes>
  <Boxes>253</Boxes>
  <Boxes>166</Boxes>
</Elements>
<Elements>
  <Score>0.374</Score>
  <Type>potted plant</Type>
  <Boxes>449</Boxes>
  <Boxes>97</Boxes>
  <Boxes>477</Boxes>
  <Boxes>140</Boxes>
</Elements>
<Elements>
  <Score>0.412</Score>
  <Type>sofa</Type>
  <Boxes>145</Boxes>
  <Boxes>135</Boxes>
  <Boxes>304</Boxes>
  <Boxes>202</Boxes>
</Elements>
<Elements>
  <Score>0.269</Score>
  <Type>sofa</Type>
  <Boxes>443</Boxes>
  <Boxes>136</Boxes>
  <Boxes>531</Boxes>
  <Boxes>190</Boxes>
</Elements>
<Elements>
  <Score>0.262</Score>
  <Type>sofa</Type>
  <Boxes>301</Boxes>
```

```
<Boxes>136</Boxes>
<Boxes>352</Boxes>
<Boxes>168</Boxes>
</Elements>
<Elements>
  <Score>0.242</Score>
  <Type>sofa</Type>
  <Boxes>452</Boxes>
  <Boxes>167</Boxes>
  <Boxes>531</Boxes>
  <Boxes>230</Boxes>
</Elements>
<Elements>
  <Score>0.219</Score>
  <Type>carpet</Type>
  <Boxes>0</Boxes>
  <Boxes>153</Boxes>
  <Boxes>272</Boxes>
  <Boxes>293</Boxes>
</Elements>
<Elements>
  <Score>0.458</Score>
  <Type>coffee table</Type>
  <Boxes>171</Boxes>
  <Boxes>172</Boxes>
  <Boxes>214</Boxes>
  <Boxes>214</Boxes>
</Elements>
<Elements>
  <Score>0.358</Score>
  <Type>coffee table</Type>
  <Boxes>337</Boxes>
  <Boxes>185</Boxes>
  <Boxes>420</Boxes>
  <Boxes>274</Boxes>
</Elements>
<Elements>
  <Score>0.285</Score>
  <Type>coffee table</Type>
  <Boxes>386</Boxes>
  <Boxes>159</Boxes>
```

```
<Boxes>428</Boxes>
<Boxes>192</Boxes>
</Elements>
<Elements>
  <Score>0.214</Score>
  <Type>mirror</Type>
  <Boxes>503</Boxes>
  <Boxes>26</Boxes>
  <Boxes>524</Boxes>
  <Boxes>85</Boxes>
</Elements>
<Width>533</Width>
</Data>
```

JSON 格式

```
{
  "RequestId": "6EF97B44-2763-4EAD-8737-FB9F5EE25FE2",
  "Data": {
    "Height": 300,
    "Elements": [
      {
        "Score": 0.266,
        "Type": "chair",
        "Boxes": [
          468,
          238,
          531,
          299
        ]
      },
      {
        "Score": 0.213,
        "Type": "chair",
        "Boxes": [
          452,
          168,
          531,
          233
        ]
      }
    ]
  }
}
```



```
”  
{  
  "Score": 0.308,  
  "Type": "picture/frame",  
  "Boxes": [  
    487,  
    44,  
    501,  
    96  
  ]  
},  
{  
  "Score": 0.257,  
  "Type": "picture/frame",  
  "Boxes": [  
    477,  
    67,  
    484,  
    98  
  ]  
},  
{  
  "Score": 0.201,  
  "Type": "picture/frame",  
  "Boxes": [  
    503,  
    26,  
    524,  
    85  
  ]  
},  
{  
  "Score": 0.515,  
  "Type": "pillow",  
  "Boxes": [  
    161,  
    141,  
    181,  
    158  
  ]  
},
```

```
{
  "Score": 0.293,
  "Type": "pillow",
  "Boxes": [
    473,
    143,
    496,
    157
  ]
},
{
  "Score": 0.214,
  "Type": "pillow",
  "Boxes": [
    241,
    145,
    253,
    166
  ]
},
{
  "Score": 0.374,
  "Type": "potted plant",
  "Boxes": [
    449,
    97,
    477,
    140
  ]
},
{
  "Score": 0.412,
  "Type": "sofa",
  "Boxes": [
    145,
    135,
    304,
    202
  ]
},
{
```

```
"Score": 0.269,  
"Type": "sofa",  
"Boxes": [  
  443,  
  136,  
  531,  
  190  
]  
},  
{  
  "Score": 0.262,  
  "Type": "sofa",  
  "Boxes": [  
    301,  
    136,  
    352,  
    168  
  ]  
},  
{  
  "Score": 0.242,  
  "Type": "sofa",  
  "Boxes": [  
    452,  
    167,  
    531,  
    230  
  ]  
},  
{  
  "Score": 0.219,  
  "Type": "carpet",  
  "Boxes": [  
    0,  
    153,  
    272,  
    293  
  ]  
},  
{  
  "Score": 0.158
```

```
    "Score": 0.456,  
    "Type": "coffee table",  
    "Boxes": [  
      171,  
      172,  
      214,  
      214  
    ]  
  },  
  {  
    "Score": 0.358,  
    "Type": "coffee table",  
    "Boxes": [  
      337,  
      185,  
      420,  
      274  
    ]  
  },  
  {  
    "Score": 0.285,  
    "Type": "coffee table",  
    "Boxes": [  
      386,  
      159,  
      428,  
      192  
    ]  
  },  
  {  
    "Score": 0.214,  
    "Type": "mirror",  
    "Boxes": [  
      503,  
      26,  
      524,  
      85  
    ]  
  }  
],  
"Width": 533
```

```
}  
}
```

错误码

关于物体检测的错误码，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. 公共信息

5.1. 请求结构

目标检测服务支持基于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://objectdet.cn-shanghai.aliyuncs.com/?Action=DetectMainBody<公共请求参数>
```

- `https`：指定了请求通信协议。
- `objectdet.cn-shanghai.aliyuncs.com`：指定了目标检测服务的接入地址（Endpoint）。
- `cn-shanghai`：区域（RegionId），请参见[访问域名查看支持的域名](#)。
- `Action=DetectMainBody`：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](#)。

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

请求参数

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

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

5.3. 公共响应参数

阿里云视觉智能开放平台是基于HTTP协议的RESTful API类型接口，所有的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://objectdet.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": "objectdet.cn-shanghai.aliyunecs.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>objectdet.cn-shanghai.aliyuncs.com</HostId>
  <Code>UnsupportedOperation</Code>
  <Message>The specified action is not supported.</Message>
</Error>
```

6. 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。

机动车检测错误码

状态码	错误码	描述
400	InvalidImage.Download	图片下载失败。
400	InvalidImage.Timeout	图片下载超时。
400	InvalidImage.URL	参数错误：无效的图像URL。
500	InternalError.Decode	图像解码失败。
500	InternalError.Process	推理过程异常。

主体检测错误码

状态码	错误码	描述
400	MissingParameter	入参缺失。
400	InvalidParameter	入参错误。
400	InvalidParameter.Unsuitable	数据为空或不适合被处理。
500	InternalError	调用算法抛错。
500	InternalError.Env	初始化环境失败。
500	InternalError.Model	模型加载错误。
500	InternalError.Convert	输入参数转换失败。
500	InternalError.Algo	算法服务出错。
500	InternalError.Transfer	下载或上传数据失败。

公共错误码

API的公共错误码请参见[API错误中心](#)。

处理建议

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

- URL类错误码

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

- 图像加载错误码

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

- 参数类错误码

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

- 系统服务类错误码

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