

车型识别开发文档

- 1、根据图片查询车辆品牌型号
- 2、错误码参照

1、根据图片查询车辆品牌型号 [\[顶部\]](#)

接口地址：http://apis.juhe.cn/carDetect/index

返回格式：json

请求方式：http post

请求示例：http://apis.juhe.cn/carDetect/index

接口备注：根据上传的车辆图片，查询车辆品牌型号年份等信息。若图片中有多个车辆，则识别目标最大的车辆

请求Header：

| 名称 | 值 |
|--------------|-----------------------------------|
| Content-Type | application/x-www-form-urlencoded |

请求参数说明：

| 名称 | 类型 | 必填 | 说明 |
|-------|--------|----|--|
| image | string | 是 | 车辆图片，base64编码，编码后大小不超过4M，最短边至少15px，最长边最大4096px,支持jpg/png/bmp格式。注意：图片需要base64编码、去掉编码头后再进行urlencode。 |
| key | string | 是 | 在个人中心->我的数据,接口名称上方查看 |

返回参数说明：

| 名称 | 类型 | 说明 |
|------------|--------|-------|
| error_code | int | 返回码 |
| reason | string | 返回说明 |
| result | string | 返回结果集 |

| 名称 | 类型 | 说明 |
|-------------|--------|---------------|
| name | string | 车型名称，示例：奥迪A4L |
| score | number | 置信度，如0.9984 |
| year | string | 年份 |
| baike_url | string | 百科词条名称，可能为空 |
| image_url | string | 百科图片链接，可能为空 |
| description | string | 百科内容描述，可能为空 |
| color | string | 颜色 |

JSON返回示例：

```
{
  "reason": "success",
  "result": {
    "data": [
      {
        "score": 0.66788327693939,
        "year": "2015-2017",
        "baike_info": {
          "baike_url": "",
          "image_url": "",
          "description": ""
        },
        "name": "奥迪_A4"
      },
      {
        "score": 0.17167422175407,
        "year": "2017",
        "baike_info": {
          "baike_url": "http://baike.baidu.com/item/%E5%A5%A5%E8%BF%AAA4L/7771482",
          "image_url": "http://imgsrc.baidu.com/baike/pic/item/8694a4c27d1ed21b1000b2b4a16eddc450da3f91.jpg",
          "description": "奥迪A4L是奥迪公司2008年11月16日正式发布的一款车型，全新奥迪A4L采用了全新的外观和底盘设计，在正面和后面该车与奥迪A5十分相似。参考价格27.28-57.81万。外观时尚动感，线条流畅；内饰做工精致，配置不错；音响效果不错；空间宽敞实用，舒适性不错；品牌经典。"
        },
        "name": "奥迪A4L"
      },
      {
        "score": 0.064318783581257,
        "year": "2003",
        "baike_info": {
          "baike_url": "",
          "image_url": "",
          "description": ""
        },
        "name": "奥迪_S4"
      }
    ],
    "color": "黄色"
  },
  "error_code": 0
}
```

```
}
```

2、错误码参照

服务级错误码参照(error_code) : [\[顶部\]](#)

| 错误码 | 说明 |
|--------|------------|
| 237101 | 识别失败 |
| 237102 | 非车类或识别不到信息 |

系统级错误码参照：

| 错误码 | 说明 | 旧版本(resultcode) |
|-------|--|-----------------|
| 10001 | 错误的请求KEY | 101 |
| 10002 | 该KEY无请求权限 | 102 |
| 10003 | KEY过期 | 103 |
| 10004 | 错误的OPENID | 104 |
| 10005 | 应用未审核超时，请提交认证 | 105 |
| 10007 | 未知的请求源 | 107 |
| 10008 | 被禁止的IP | 108 |
| 10009 | 被禁止的KEY | 109 |
| 10011 | 当前IP请求超过限制 | 111 |
| 10012 | 请求超过次数限制 | 112 |
| 10013 | 测试KEY超过请求限制 | 113 |
| 10014 | 系统内部异常(调用充值类业务时，请务必联系客服或通过订单查询接口检测订单，避免造成损失) | 114 |
| 10020 | 接口维护 | 120 |
| 10021 | 接口停用 | 121 |

错误码格式说明（示例：200201）：

| | | |
|---|-----|----|
| 2 | 002 | 01 |
|---|-----|----|

| | | |
|----------------|---------------|--------|
| 2 | 002 | 01 |
| 服务级错误（1为系统级错误） | 服务模块代码(即数据ID) | 具体错误代码 |

版本日期：2024-11-21 19:26