

# 接口说明

图像识别（OCR）可以将图片中的文字转换为计算机可编程的文字。该能力通过HTTP API的方式提供给开发者，相较于SDK，该方式具有轻量、跨平台、跨开发语言的特点。

使用时请求方通过HTTP协议的POST方法将图片信息一次性的发送到服务器，然后通过HTTP协议的GET方法去服务器获取识别结果。与一次性交互的方式（既将图片信息一次性发送到服务器，然后等服务器处理完成之后该请求才携带识别结果返回）相比，发送图片的请求会在服务器收到图片之后就返回，不会阻塞到服务器识别完图片，减少调用等待的时间，应用处理起来更灵活。

## 接口要求

项目	说明
请求地址	<a href="http://api.baller-tech.com/v1/service/v1/ocr">http://api.baller-tech.com/v1/service/v1/ocr</a>
请求方式	发送图像数据时使用POST；获取识别结果时使用GET
字符编码	UTF-8
图像格式	jpg; jpeg; bmp; png; gif; tif; tiff
图像大小	图像大小不超过4M

## 接口签名

为了防止通信过程中发送的消息被他人窃取和修改，每一个HTTP协议接口都需要进行签名验证，服务器发现请求的签名不一致时会拒绝处理。

将**app\_key**（由北京大牛儿科技发展有限公司统一分配）、**请求时间**（GMT格式）、**base64编码后的业务参数**按照固定的顺序组成的字符串MD5后的结果作为签名，放到请求报文的Header的B-Checksum参数中。

## 接口调用模式

根据识别结果获取的方式不同，分为两种调用模式：

1. 连续调用HTTP的GET方法获取识别结果，适用于直接在终端客户的设备上发起请求时。
2. 将识别结果推送到请求时指定的HTTP 地址上，适用于在对接方公司服务器上发起请求时（终端客户与对接方公司服务器通信，对接方服务器调用本请求）。

### 连续调用HTTP的GET方法获取识别结果

1. 通过HTTP协议POST方法，将图像数据一次性的发送到服务器。
2. 通过HTTP协议GET方法，去服务器获取识别结果以及是否获取结束的状态；
3. 如果HTTP协议GET方法的响应中是否获取结束的状态为未结束，需要继续调用HTTP协议GET方法请求识别结果；为了避免频繁的交互浪费CPU和网络资源，两次HTTP协议GET方法的请求之间可以间隔一段时间（具体值可以根据使用场景进行测试确定，建议150~200毫秒）。

### 将识别结果推送到请求时指定的HTTP 地址上

1. 通过HTTP协议POST方法，将图像数据一次性的发送到服务器，发送数据时携带结果推送的地址。
2. 服务器通过HTTP协议的POST方法，分多次将识别的结果推送到请求时指定的地址。

## 接口参数

### 1. POST方法请求参数

#### 1.1 HTTP请求Header中需设置参数

参数	类型	说明	举例
B-AppId	string	由北京大牛儿科技发展有限公司统一分配； 分配的值为64位的整型，此处需要转换为string	1176611429127553031
B-CurTime	string	GMT+0时区的符合RFC1123格式的日期和时间，星期和月份只能使用英文表示； 需和接口签名时的请求时间一致； 服务器会拒绝处理请求时间与当前时间相差300秒的请求	Fri, 10 Jan 2020 07:31:50 GMT
B-Param	string	经过BASE64编码后的业务参数，参见 <a href="#">业务参数</a>	
B-CheckSum	string	参见 <a href="#">接口签名</a> 。	
Content-Type	string	传输数据的类型，此处使用固定值	application/octet-stream

##### 1.1.1 业务参数介绍

参数	类型	说明	举例
request_id	string	本次图像识别事务的请求ID； 获取该请求识别结果时需携带相同的请求ID； 调用者需保证请求ID的唯一性，建议使用UUID	6497c282-9371-4c68-a9f1-522212b5ac1d
image_mode	string	传入图片的文本模式，可选值为：multi_row	multi_row
language	string	识别语种参见 <a href="#">支持的图像识别语种</a>	chs
callback_url	string	识别结果推送的回调地址； 通过调用HTTP的GET方法获取识别结果时不需设置	<a href="http://192.168.1.234:18888/ocr/callback">http://192.168.1.234:18888/ocr/callback</a>

#### 1.2 HTTP请求Body

待识别的图像数据（二进制）。

#### 1.3 响应报文

http响应数据为json格式，具体字段的含义如下

参数	类型	说明
code	int	请求处理的结果码 (0：成功；其他：失败)
message	string	对code字段的文本说明
request_id	string	请求时传入的request_id

```
{
  "code": 0,
  "message": "success",
  "request_id": "f7409982-dc05-4d19-80c9-6169dd70b247"
}
```

## 2. GET方法请求参数

### 2.1 HTTP请求Header中需设置参数

参数	类型	说明	举例
B-AppId	string	由北京大牛儿科技发展有限公司统一分配； 分配的值为64位的整型，此处需要转换为string	1176611429127553031
B-CurTime	string	GMT+0时区的符合RFC1123格式的日期和时间，星期和月份只能使用英文表示； 需和接口签名时的请求时间一致； 服务器会拒绝处理请求时间与当前时间相差300秒的请求	Fri, 10 Jan 2020 07:31:50 GMT
B-Param	string	经过BASE64编码后的业务参数，参见 <a href="#">业务参数</a>	
B-Checksum	string	参见 <a href="#">接口签名</a> 。	

#### 2.1.1 业务参数介绍

参数	类型	说明	举例
request_id	string	本次图像识别事务的请求ID； 需与POST时保持一致	6497c282-9371-4c68-a9f1- 522212b5ac1d

## 2.2 响应报文

图像识别时，会将图像按一定的规则（目前为按行）分为不同的子项，每次返回的识别结果是一个或多个子项的识别结果。响应数据为json格式，具体字段的含义如下

参数	类型	说明
code	int	请求处理的结果码 (0: 成功; 其他: 失败)
message	string	对code字段的文本说明
request_id	string	请求时传入的request_id
is_end	int	识别结果是否获取结束 (1: 结束; 0: 未结束)
data	json数组	本次获取到的子句的识别结果, <a href="#">详见</a>

### 2.2.1 data字段介绍

参数	类型	说明
order	int	当前响应报文中子句的顺序 (是一次GET响应报文的顺序, 不是整个识别事务的)
result	string	该子句的识别结果

```
{
  "code": 0,
  "message": "success",
  "is_end": 0,
  "request_id": "84d13f58-5f6e-11ea-9438-4023431f608e",
  "data": [
    {
      "order": 0,
      "result": "xxx"
    }
  ]
}
```

## 3. 推送识别结果的消息格式

采用服务器推送识别结果时, 推送的消息格式与GET请求的响应报文格式基本一致。不一样的地方是会在code参数同级添加一个order参数, 表示本次事务推送的次序, 从0开始依次递增。

```
{
  "code": 0,
  "message": "success",
  "is_end": 0,
  "request_id": "84d13f58-5f6e-11ea-9438-4023431f608e",
  "order": 0,
  "data": [
    {
      "order": 0,
      "result": "xxx"
    }
  ]
}
```

## 支持的语种

图像识别语种	对应的language字段
简体中文	chs
繁体中文	cht
藏文	tib
蒙文(传统)	mon
蒙文(西里尔)	mon_o
维文【暂不支持】	uig
彝文【暂不支持】	iii
壮文【暂不支持】	zha
韩文【暂不支持】	kor
哈萨克文【暂不支持】	kaz