

思图场景 OpenApi

行驶证 OCR 接口文档

版本修订历史纪录				
序号	修订内容	修订日期	修订人	版本
1	基本接口文档	2017-09-28	李利雪	

行驶证 OCR 接口

接口介绍

行驶证 OCR 接口，回传一张行驶证的照片，返回 OCR 结果。

接口信息

调用地址：<http://openapi.sitodata.com/v1/ocr>

测试调用地址：<http://test-openapi.sitodata.com/v1/ocr>

请求方式：POST

返回类型：JSON

编码格式：utf-8

请求参数：Header

名称	类型	是否必须	描述
X-Auth-Token	String	是	每个账号都会唯一指定一个 token，请妥善保管，token 是接口认证的唯一方法

请求参数：Body

名称	类型	必须	描述
vehicleLicense	String	是	图片转 base64 编码以后的字符串（json）

返回参数：

参数名	类型	描述
code	int	结果： 成功 = 1， 参数错误 = 100， 账户余额不足 = 101 测试次数不足 = 102 文件编码后大小超限 = 103 图片格式传入错误 = 104

		查询失败= 105 不支持的卡片类型 = 106 识别失败 = 107	
message	String	结果说明	
result	owner	String	所有人
	vType	String	车辆类型
	address	String	住址
	model	String	品牌型号
	registerDate	String	注册日期
	issueDate	String	发证日期
	vin	String	车辆识别代号
	plateNo	String	号牌号码
	useCharacter	String	使用性质
	engineNo	String	发动机号码

请求示例: Java

```
import sun.misc.BASE64Encoder;

import java.io.*;
import java.net.HttpURLConnection;
import java.net.URL;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class BankCardTest {

    private static final String strUrl = "http://test-openapi.situdata.com/vl/ocr";

    public static void main(String[] args) {
        String filePath = "行驶证照片路径";
        File file = new File(filePath);

        String[] filename = file.list();
        for(String vfile : filename) {
            String imageToBase64 = imageToBase64(filePath + vfile);
            String replaceBlank = replaceBlank(imageToBase64);
            System.out.println(replaceBlank);
            String params = "{\"vehicleLicense\":\"\" + replaceBlank + "\"}";

            try {
                String readByPost = readByPost(strUrl, params);
            }
        }
    }
}
```

```
        System.out.println(readByGet);
    } catch (IOException e) {
        e.printStackTrace();
    }
}

}

private static String readByPost(String inUrl,String params) throws IOException {
    StringBuffer sbf = new StringBuffer();
    String strRead = null;

    URL url = new URL(inUrl);
    HttpURLConnection connection = (HttpURLConnection) url.openConnection();
    connection.setRequestMethod("POST");
    connection.setDoInput(true);
    connection.setDoOutput(true);
    connection.setRequestProperty("Content-Type", "application/json");
    connection.setRequestProperty("X-Auth-Token", "您的 token 秘钥");

    connection.connect();

    OutputStreamWriter writer = new
OutputStreamWriter(connection.getOutputStream(),"UTF-8");
    writer.write(params);
    writer.flush();
    InputStream is = connection.getInputStream();
    BufferedReader reader = new BufferedReader(new InputStreamReader(is,
        "UTF-8"));
    while ((strRead = reader.readLine()) != null) {
        sbf.append(strRead);
        sbf.append("\r\n");
    }
    reader.close();
    connection.disconnect();
    return sbf.toString();
}

public static String imageToBase64(String path) {
    byte[] data = null;
    try {
        InputStream in = new FileInputStream(path);
        data = new byte[in.available()];
        in.read(data);
    }
```

```
        in.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
    BASE64Encoder encoder = new BASE64Encoder();
    return encoder.encode(data);
}
public static String replaceBlank(String str) {
    String dest = "";
    if (str!=null) {
        Pattern p = Pattern.compile("\\s*|\t|\r|\n");
        Matcher m = p.matcher(str);
        dest = m.replaceAll("");
    }
    return dest;
}
}
```

正常返回示例:

案例一:

```
{
  "result": {
    "owner": "陆波",
    "vType": "小型轿车",
    "address": "重庆市大足县回龙镇永兴村4组",
    "model": "名爵牌 CSA7130ACS",
    "registerDate": "2012-08-01",
    "issueDate": "2014-08-28",
    "vin": "LSJZ14C31CS046221",
    "plateNo": "渝 C65H29",
    "useCharacter": "非营运",
    "engineNo": "B2GC5220188"
  },
  "code": "1",
  "message": "Ok",
  "analyzeId": 73352
}
```

案例二

```
{
  "result": null,
  "code": "105",
  "message": "查询失败",
  "analyzeId": 0
}
```

