



文字识别实验手册

技术背景：百度 AI 文字识别技术（卡片识别方向）

实验案例：身份证识别

编著者：百度 AI 开放平台&极客学院

百度 AI 技术课程专区（极客学院）<http://zt.jikexueyuan.com/baiduai>

欢迎来到场景实验室！

在这里，我们将从真实工作场景及实际项目需求出发，以某一案例为基础带领大家完成此阶段学习。本实验手册可以帮助大家快速的进入产品使用阶段，跟着实验步骤一步步的完成，即可快速获得输出成果。

实验案例：身份证识别

掌握了文字识别的基础技术知识后，我们可以自己设计实验项目来巩固理论知识。今天我们将以【身份证识别】项目实验为案例，教会大家如何利用文字识别技术一步步的部署自己的应用。



一、实验概述

随着人们对个人信息的维护意识越来越强，更多的应用需要针对个人用户进行身份验证。在身份验证的流程中，【卡证识别】不可缺少的一环。本实验为使用百度 AI 开放平台的文字识别相关功能，实现工作场景中的实际需求 — 身份证识别。

二、实验收获

- ◆ 熟悉百度 AI 开放平台服务 — 授权&认证
- ◆ 快速掌握文字识别中多种 API 接口使用流程

- ◆ 快速掌握文字识别中各 SDK 对应语言使用流程
- ◆ 实现身份证识别效果

二、实验步骤

第一步、接入百度 AI 开放平台



详情可参考：<http://ai.baidu.com/docs#/Begin/top>

第二步、开始动手实验

温馨提示：动手实验前，请先确定【身份证识别】实验的基础工作已经准备完毕！

- ◆ 掌握 Android/iOS 开发中摄像头功能原理
- ◆ 熟悉摄像头应用开发流程
- ◆ 项目相关的 UI 工作已完成

1、选择方式调用 API 接口

1.1 本实验以文字识别相关功能为例。

需参考：<http://ai.baidu.com/tech/ocr>

1.2 调用 AI 服务相关的 API 接口有两种调用方式，两种不同的调用方式采用相同的接口 URL。区别在于请求方式和鉴权方法不一样，请求参数和返回结果一致。

1.3 接入文字识别-网络图片文字识别功能。

需参考：<http://ai.baidu.com/docs#/OCR-API/top>

2、核心代码实现

2.1 请求 URL 数据格式

向 API 服务地址使用 POST 发送请求，必须在 URL 中带上参数。access_token: 必须参数，参考新手指南中的【鉴权认证机制】。

请求示例：文字识别 API，使用 HTTPS POST 发送：

```
https://aip.baidubce.com/rest/2.0/ocr/v1/general?  
access_token=24.f9ba9c5241b67688bb4adbed8bc91dec.2592000.1485570332.282335-8574074
```

获取示例：Python 代码

```
bash  PHP  Java  Python  C++  C#  
  
import urllib, urllib2, sys  
import ssl  
  
# client_id 为官网获取的AK, client_secret 为官网获取的SK  
host = 'https://aip.baidubce.com/oauth/2.0/token?grant_type=client_credentials&client_id=【官  
网获取的AK】&client_secret=【官网获取的SK】'  
request = urllib2.Request(host)  
request.add_header('Content-Type', 'application/json; charset=UTF-8')  
response = urllib2.urlopen(request)  
content = response.read()  
if (content):  
    print(content)
```

说明：鉴权使用的 Access_token 必须通过 API Key 和 Secret Key 获取，且有效期为 30 天，需要每 30 天进行定期更换。

2.2 身份证识别-请求说明

用户向服务请求识别身份证，身份证识别包括正面和背面。

HTTP 方法：POST

请求 URL：<https://aip.baidubce.com/rest/2.0/ocr/v1/idcard>

URL 参数：

参数	值
access_token	通过 API Key 和 Secret Key 获取的 access_token,参考新手指南中的【鉴权认证机制】。

Header 如下：

参数	值
Content-Type	application/x-www-form-urlencoded

请求参数：

参数	是否必选	类型	可选范围	说明
detect_direction	0	boolean	true、false	是否检测图像朝向，默认不检测，即：false。朝向是指输入图像是正常方向、逆时针旋转 90/180/270 度。可选值包括： - true：检测朝向； - false：不检测朝向。

id_card_side	1	string	true、false	front : 身份证正面 ; back : 身份证背面
image	1	string	-	图像数据 , base64 编码 , 要求 base64 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
detect_risk	0	string	true、false	是否开启身份证风险类型(身份证复印件、临时身份证、身份证翻拍、修改过的身份证)功能 , 默认不开启 , 即 : false。 可选值:true-开启 ; false-不开启

2.3 身份证识别-返回说明

请求参数 :

参数	是否必选	类型	说明
direction	0	int32	图像方向 , 当 detect_direction=true 时存在。 - -1:未定义 , - 0:正向 , - 1: 逆时针 90 度 , - 2:逆时针 180 度 , - 3:逆时针 270 度
image_status	1	string	normal-识别正常 reversed_side-未摆正身份证 non_idcard-上传的图片中不包含身份证

			<p>blurred-身份证模糊</p> <p>over_exposure-身份证关键字段反光或过曝</p> <p>unknown-未知状态</p>
risk_type	0	string	<p>输入参数 detect_risk = true 时，则返回该字段识别身份证类型: normal-正常身份证；copy-复印件；temporary-临时身份证；screen-翻拍；unknow-其他未知情况</p>
edit_tool	0	string	<p>如果参数 detect_risk = true 时，则返回此字段。如果检测身份证被编辑过，该字段指定编辑软件名称，如:Adobe Photoshop CC 2014 (Macintosh),如果没有被编辑过则返回值无此参数</p>
log_id	1	uint64	<p>唯一的 log id，用于问题定位</p>
words_result	1	array()	<p>定位和识别结果数组</p>
words_result_num	1	uint32	<p>识别结果数，表示 words_result 的元素个数</p>

+location	1	array()	位置数组（坐标 0 点为左上角）
++left	1	uint32	表示定位位置的长方形左上顶点的水平坐标
++top	1	uint32	表示定位位置的长方形左上顶点的垂直坐标
++width	1	uint32	表示定位位置的长方形的宽度
++height	1	uint32	表示定位位置的长方形的高度
+words	0	string	识别结果字符串

返回示例：

```
{
  "log_id": 2648325511,
  "direction": 0,
  "image_status": "normal",
  "idcard_type": "normal",
  "edit_tool": "Adobe Photoshop CS3 Windows",
  "words_result": {
    "住址": {
      "location": {
        "left": 267,
        "top": 453,
        "width": 459,
        "height": 99
      },
      "words": "南京市江宁区弘景大道3889号"
    },
    "公民身份号码": {
      "location": {
        "left": 443,
        "top": 681,
        "width": 589,
        "height": 45
      },
      "words": "330881199904173914"
    },
    "出生": {
      "location": {
        "left": 270,
        "top": 355,
        "width": 357,
        "height": 45
      },
      "words": "19990417"
    },
    "姓名": {
      "location": {
        "left": 267,
        "top": 176,
        "width": 152,
        "height": 50
      },
      "words": "伍云龙"
    },
    "性别": {
      "location": {
        "left": 269,
        "top": 262,
        "width": 33,
        "height": 52
      },
      "words": "男"
    },
    "民族": {
      "location": {
        "left": 492,
        "top": 279,
        "width": 30,
        "height": 37
      },
      "words": "汉"
    }
  },
  "words_result_num": 6
}
```

通过授权与认证机制向服务器发送 POST 请求并接收返回数据，便可以根据项目的需求来选择性的使用 API 提供的相关功能。

第三步、代码调试及实验完成

程序调试是将编制的程序投入实际运行前，用手工或编译程序等方法进行测试，修正语法错误和逻辑错误的过程。

在代码调试的过程中，需遵循以下原则：

- ✓ 用头脑去分析思考与错误征兆有关的信息。
- ✓ 避开死胡同。
- ✓ 只把调试工具当做手段。利用调试工具，可以帮助思考，但不能代替思考，因为调试工具给的是一种无规律的调试方法。
- ✓ 避免用试探法，最多只能把它当做最后手段。
- ✓ 再出现错误的地方，可能还有别的错误。
- ✓ 修改错误的一个常见失误是只修改了这个错误的征兆或这个错误的表现，而没有修改错误本身。如果提出的修改不能解释与这个错误有关的全部线索，那就表明只修改了错误的一部分。
- ✓ 注意修正一个错误的同时可能会引入新的错误。

第四步、常见问题

实验期间如遇任何问题，可通过以下两种方式解决

方式一、去常见问题 <http://ai.baidu.com/docs#/FAQ/top> 查阅文字识别相关各类问题及解答。

方式二、去技术论坛（<http://developer.baidu.com/forum#ai>）中发表遇到的问题。

通过每次实验，相信大家可以快速入手百度 AI 开放平台中文字识别相关功能的研发工作。大家可以举一反三，将文字识别或平台中的其他技术（图像识别/人脸识别等）应用到多领域多端的产品中，赶快动手完成自己的实验吧！

百度 AI 技术课程专区（极客学院）<http://zt.jikexueyuan.com/baiduai>

欢迎大家加入我们，一起交流实验内容及更多百度 AI 知识。



百度 AI (图像技术) 小组

这里有来自百度的技术大牛，这里有热爱 AI 技术的小伙伴。加入我们，让复杂的世界更简单！

QQ 扫码或直接加群：[664176161](https://qm.qq.com/join/664176161)