**Android 图像识别之OCR识别身份证，银行卡等证件信息**

2017年11月14日 16:42:31 [world\_kun](https://me.csdn.net/world_kun) 阅读数：4219

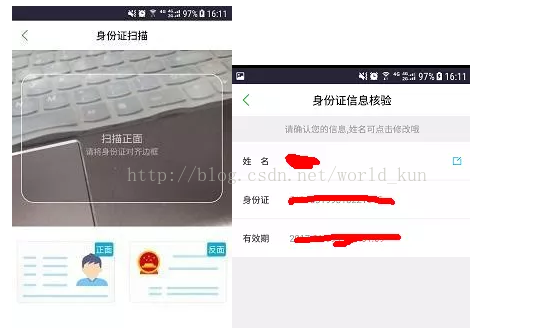
尊重作者劳动成果，转载请注明出处

原文链接：https://mp.weixin.qq.com/s/oRpW9AWLx60GpINHJ\_\_BNw

前言

        OCR （Optical Character Recognition，光学字符识别）是指电子设备（例如扫描仪或数码相机）检查纸上打印的字符，通过检测暗、亮的模式确定其形状，然后用字符识别方法将形状翻译成计算机文字的过程；

效果图

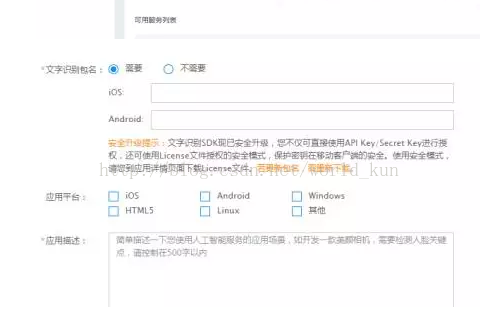


        关于OCR文字识别技术，它的官方文档搞得很复杂，而且最终的效果也不理想，但是还好，百度的AI平台已经在此基础上改进了，于是，今天就给大家带来百度AI平台的文字识别技术，可识别身份证，银行卡，驾驶证，车牌等多种证件。

1、我们需要进入百度云文字识别的开发平台，进行一系列的注册，登陆。  
https://cloud.baidu.com/product/ocr

2、创建应用





        在创建应用的时候，应用名称我们可以随便写一个，但是我们要注意的是：我们在写文字识别包名的时候一定要写我们项目的包名，不然在识别的时候会提示你获取Token出错，接下来我们就直接创建我们的应用。

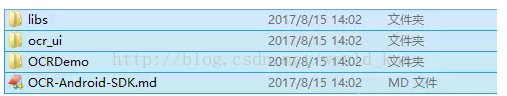
3、查看管理应用列表



        这个就是我们刚刚创建的应用，我们需要记住它的API Key和Secret Key，在我们的项目中可能会用到。

4、点击应用列表左边最下面的sdk下载，就会到下载界面下载我们需要的技术文档和sdk。在sdk里面也有一份技术文档

        下载下来sdk后，解压后，目录如下：



        关于这几个文件的作用，引用官方文档的一张图来说明



https://img-blog.csdn.net/20171122165344213?watermark/2/text/aHR0cDovL2Jsb2cuY3Nkbi5uZXQvd29ybGRfa3Vu/font/5a6L5L2T/fontsize/400/fill/I0JBQkFCMA==/dissolve/70/gravity/Center

* libs：包含核心jar包以及so文件，直接拷贝到你项目的libs中即可
* ocr-ui：是它提供的一个扫描的默认ui，我们可以用，也可以不用，
* OCRDemo ：官方提供的一个包含多种功能，可以直接运行的demo，但是必须要自己配置AK和SK才能运行，否则会提示获取Token失败。

必要的说明：看完了这里，你就可以少踩坑

        直接运行官方demo，会失败，他会提示获取AccessToken失败。那什么是AccessToken？怎么获取？别急，笔者这就为大家讲述

注意：我们在使用百度AI平台的SDK之前必须先获取AccessToken成功

百度AI开放平台使用OAuth2.0授权调用开放API，调用API时必须在URL中带上accesss\_token参数。AccessToken可用AK/SK或者授权文件的方式获得。

        OCR Android SDK提供了以下3种获取AccessToken的方法.

方案一、API Key / Secret Key获取Token

        也就是说调用下面这段代码获取AccessToken时，需要填写刚才你创建的应用的API Key / Secret Key，注意包名必须和你项目的包名一致，否则会失败。

1. OCR.getInstance().initAccessTokenWithAkSk(new OnResultListener<AccessToken>() {  
       @Override
2. public void onResult(AccessToken result) {  
           *// 调用成功，返回AccessToken对象*
3. String token = result.getAccessToken();
4. }  
       @Override
5. public void onError(OCRError error) {  
           *// 调用失败，返回OCRError子类SDKError对象*
6. }
7. }, getApplicationContext(), "您的应用AK", "您的应用SK");

        虽然SDK对网络传输的敏感数据进行了二次加密，但由于AK/SK是明文填写在代码中，在移动设备中可能会存在AK/SK被盗取的风险。有安全考虑的开发者可使用第二种授权方案。

方案二、授权文件（安全模式）获取Token

        此种身份验证方案使用授权文件获得AccessToken，缓存在本地。建议有安全考虑的开发者使用此种身份验证方式。在您的移动APP分发出去之后，APP存在被反编译的可能，所以直接将AK / SK 置于APP源码之中，存在被盗取的风险。采用授权文件的身份验证方法，可有效保护AK/SK在移动设备中的安全。攻击者即使拦截了流量，盗取了授权文件，也难以盗用您的配额。

这时候应该有人就在想 什么是授权文件？如何获取？



* 在应用列表找到对应的应用
* 下载对应应用的授权文件
* 将授权文件添加至工程assets文件夹，文件名必须为aip.license
* 调用initAccessToken方法，初始化OCR单例，来获取AccessToken：

1. OCR.getInstance().initAccessToken(new OnResultListener<AccessToken>() {  
       @Override
2. public void onResult(AccessToken result) {  
           *// 调用成功，返回AccessToken对象*
3. String token = result.getAccessToken();
4. }      
       @Override
5. public void onError(OCRError error) {          
           *// 调用失败，返回OCRError子类SDKError对象*
6. }
7. }, getApplicationContext());

        通过这种方式，你就不需要填写你的ak和sk了，这种授权文件的方式较为安全，推荐使用。

3、自助AccessToken管理(不推荐)

        此种身份验证方案直接使用开发者提供的AccessToken，鉴于安全性考虑不推荐此方式，使用此模式将不能开启身份证本地质量控制能力。  
        若开发者的应用有自行搭建的服务端，也可在自己的服务端进行token的获取与管理，然后分配给移动客户端使用。token的获得可以参考API鉴权认证机制。此种授权方案在移动客户端上没有任何AK/SK信息，风险系数低，但需要开发者自行管理token的获取与分配，适合有条件的开发者使用。

使用步骤：  
        将您需要使用的token传入对应的构造函数，初始化OCR单例：

OCR.getInstance().initWithToken(getApplicationContext(), "自行获取的access\_token");

        到这里，核心部分就讲完了，你只需要按照文档上面的SDK导入就好，注释很清楚，在此提供一个官方的接入文档的地址

[http://ai.baidu.com/docs#/OCR-Android-SDK/top](http://mp.weixin.qq.com/s?__biz=MzI0MjE3OTYwMg==&mid=2649550710&idx=1&sn=a39775baa4e7a71bd8493fb9cdfb6702&chksm=f118040bc66f8d1de418bdd36d482828f4c0afe6e22e4064a8f32a09c3c72a162af229b34def&scene=21#wechat_redirect)

Demo地址：[https://github.com/zhouxu88/OCRDemo](http://mp.weixin.qq.com/s?__biz=MzI0MjE3OTYwMg==&mid=2649550696&idx=2&sn=1edcfd9bdb27e9d2a605776e93e18801&chksm=f1180415c66f8d03b5c4b10063951af60192830c8b66a87f8dde76ab10205dd41e2346f10031&scene=21#wechat_redirect)