

# 图像分析

## SDK 文档

### 产品文档



腾讯云

**【版权声明】**

©2013-2019 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

**【商标声明】**

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

**【服务声明】**

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或模式的承诺或保证。

## 文档目录

### SDK 文档

SDK 概览

Java-SDK-图像识别

Python-SDK-图像识别

CPP-SDK-图像识别

PHP-SDK-图像识别

Node-SDK-图像识别

# SDK 文档

## SDK 概览

最近更新时间：2019-03-21 15:33:43

除了直接使用 API 接口以外，腾讯云智能图像识别还提供了丰富多样的 SDK 供开发者使用。

### 图像识别 SDK

SDK	描述
<a href="#">Java-SDK- 图像识别</a>	提供图片标签 Java SDK 的操作指引
<a href="#">Python-SDK- 图像识别</a>	提供图片标签 Python SDK 的操作指引
<a href="#">CPP-SDK- 图像识别</a>	提供图片标签 CPP SDK 的操作指引
<a href="#">PHP-SDK- 图像识别</a>	提供图片标签 PHP SDK 的操作指引
<a href="#">Node-SDK-图像识别</a>	提供图片标签 Node SDK 的操作指引

# Java-SDK-图像识别

最近更新时间：2019-06-14 18:42:03

## SDK 获取

智能图像的 Java SDK 下载地址：[Java-SDK-V2.0](#)。

## 使用前准备

1. 前往注册：[腾讯云账号注册](#)（详细指引见[注册腾讯云](#)）。
2. 取得 **APPID**、**SecretId**、**SecretKey**：请前往 [云 API 密钥](#)，单击【新建密钥】（目前只支持主账号及密钥进行调用）。

## 集成方式

### 获得 SDK jar 文件

您可以通过以下两种方式获得文件：

- 直接使用 `release/*-with-dependencies.jar`
- 自行编译：在工程根目录下执行命令 `mvn assembly:assembly`，编译结果见 `target/*-with-dependencies.jar`

### 导入 jar 到项目中

根据项目具体情况导入 `*-with-dependencies.jar`

## 使用简介

### 初始化

```
ImageClient imageClient = new ImageClient(APPID, SecretId, SecretKey);
```

### 设置代理

根据实际网络环境设置代理，例如：

```
Proxy proxy = new Proxy(Type.HTTP, new InetSocketAddress("127.0.0.1", 8080));
imageClient.setProxy(proxy);
```

## 使用

SDK 提供功能如下：

**图像识别**：鉴黄、标签

**文字识别 (OCR)**：身份证、名片、通用、驾驶证行驶证、营业执照、银行卡、车牌号

```
// 调用车牌识别 API 示例
String imageUrl = "http://youtu.qq.com/app/img/experience/char_general/icon_ocr_license_3.jpg";
String result = imageClient.ocrPlate(new OcrPlateRequest("bucketName", imageUrl));
System.out.println(result);
```

更多例子详情可参见 [Demo.java](#) 的代码。

## Demo 工程

1. 修改文件 `src/main/java/com/qcloud/image/demo/Demo.java` 的 `main()` 方法，填入上述申请到的 **APPID**、**SecretId**、**SecretKey**
2. 导入到 IDE：这个 Demo 工程是用 Maven 构建的，以 IntelliJ IDEA 为例，导入方式为：Import Project > 选择工程目录 > Import project from external model > Maven
3. 运行：Demo.java 右键，Run Demo.main()

# Python-SDK-图像识别

最近更新时间：2019-06-14 18:42:10

## 开发准备

### SDK 获取

智能图像的 Python SDK 下载地址：[Python-SDK-V2.0](#)。

### 开发准备

#### 使用 pip

Python 2：

```
pip install qcloud_image
```

Python 3：

```
pip3 install qcloud_image
```

## 快速入门

### 在腾讯云申请业务的授权

授权包括：APPID、SecretId、SecretKey，目前只支持主账号及密钥进行调用。

#### ⚠ 注意：

BUCKET 为历史遗留字段, 无需修改。

### 创建对应操作类的对象

如果要使用图片，需要创建图片操作类对象：

```
from qcloud_image import Client
from qcloud_image import CIUrl, CIFile, CIBuffer, CIUrls, CIFiles, CIBuffers
appid = 'APP_ID'
secret_id = 'SECRET_ID'
secret_key = 'SECRET_KEY'
bucket = 'BUCKET'
client = Client(appid, secret_id, secret_key, bucket)
client.use_http()
client.set_timeout(30)
```

## 调用对应的方法

创建完对象后，您可以根据实际需求调用对应的操作方法。

图片识别包括：图片鉴黄、图像分析、OCR - 身份证识别及 OCR - 名片识别。

### 图片鉴黄

```
//单个或多个图片 Url
print (client.porn_detect(CIUrls(['http://jiangsu.china.com.cn/uploadfile/2015/1102/1446443026382534.jpg','http://n.sinaimg.cn/fashion/transform/20160704/flgG-fxtspsa6612705.jpg'])))
//单个或多个图片 File
print (client.porn_detect(CIFiles(['./test.jpg'])))
```

### 图像分析

```
//单个图片 Url
print (client.tag_detect(CIUrl('http://img3.a0bi.com/upload/ttq/20160814/1471155260063.png'))
//单个图片 File
print (client.tag_detect(CIFile('./hot2.jpg')))
```

### OCR - 身份证识别

```
//单个或多个图片 Url,识别身份证正面
print (client.idcard_detect(CIUrls(['http://imgs.focus.cn/upload/sz/5876/a_58758051.jpg']), 0))
//单个或多个图片 File,识别身份证正面
print (client.idcard_detect(CIFiles(['./id4zheng.jpg','./id1zheng.jpg']), 0))
//单个或多个图片 Url,识别身份证反面
print (client.idcard_detect(CIUrls(['http://www.csx.gov.cn/cfwf/bszn/201403/W020121030349825312574.jpg', 'http://www.4009951551.com/upload/image/20151026/1445831136187479.png']), 1))
//单个或多个图片 File,识别身份证反面
print (client.idcard_detect(CIFiles(['./id5_fan.jpg']), 1))
```

### OCR - 名片识别

```
//单个或多个图片 Url
print (client.namecard_detect(CIUrls(['http://pic1.nipic.com/2008-12-03/2008123181119306_2.jpg', 'http://pic.58pic.com/58pic/12/49/04/80k58PICzYP.jpg'])))
//单个或多个图片 File
print (client.namecard_detect(CIFiles(['./name1.jpg'])))
```



# CPP-SDK-图像识别

最近更新时间：2019-06-14 18:42:16

## 开发准备

### ⚠ 注意：

以下为 Linux 等类 UNIX 系统使用手册，暂不支持 Windows 系统。

## SDK 获取

智能图像 C++ SDK 下载地址：[cpp-SDK-V2.0](#)。

## 开发准备

依赖静态库：curl jsoncpp (在 lib 文件夹下)；

依赖动态库：ssl crypto rtz (需要安装)；

(1)安装 openssl 的库和头文件 <http://www.openssl.org/source/>；

(2)安装 curl 的库和头文件 <http://curl.haxx.se/download/curl-7.43.0.tar.gz>；

(3)安装 jsoncpp 的库和头文件 <https://github.com/open-source-parsers/jsoncpp>；

(4)安装 cmake 工具 <http://www.cmake.org/download/>。

## SDK 配置

直接下载 github 上提供的源代码，集成到您的开发环境。

执行下面的命令：

```
cd ${image-cpp-sdk-v2.0}
mkdir -p build
cd build
cmake ..
make
```

image\_demo.cpp 里面有常见 API 的例子。生成的 image\_demo 可直接运行，生成的静态库名称为：

libimagesdk.a。生成的 libimagesdk.a 放到用户自己的工程里 lib 路径下，include 目录拷贝到用户的工程的 include 路径下。

## 快速入门

## 在腾讯云申请业务的授权

授权包括：APPID、SecretId、SecretKey，目前只支持主账号及密钥进行调用。

## 创建对应操作类的对象

如果要使用图片，需要创建图片操作类对象：

```
//设置全局参数（非必须）
ImageSysConfig::setAuthExpiredTime(300); //设置签名超时时长300s
//生成ImageAPI对象
ImageConfig config(APP_ID, SECRET_ID, SECRET_KEY);
ImageAPI image(config);
```

## 调用对应的方法

在创建完对象后，根据实际需求，调用对应的操作方法就可以了。

## 图片识别

图片识别包括：图片鉴黄、图像分析、OCR - 身份证识别及 OCR - 名片识别。

### 图片鉴黄

```
//单个或多个图片 Url
vector<string> pornUrls; pornUrls.push_back("http://hearthstone.nos.netease.com/1/artworkGvG/Gob
linBlastmagel.jpg");
pornUrls.push_back("http://hearthstone.nos.netease.com/1/artworknaxx/Faerlinal.jpg");
pornUrls.push_back("http://hearthstone.nos.netease.com/1/artworknaxx/KelThuzadl.jpg");
PornDetectReq pornReq(BUCKET, pornUrls);
ret = image.PornDetect(pornReq);
cout<<ret<<endl;
//单个或多个图片 File
map<string, string> pornImages;
pornImages["1.jpg"] = FileUtil::getFileContent("pic/1.jpg");
pornImages["2.jpg"] = FileUtil::getFileContent("pic/2.jpg");
pornImages["3.jpg"] = FileUtil::getFileContent("pic/3.jpg");
PornDetectReq pornReq2(BUCKET, pornImages);
ret = image.PornDetect(pornReq);
cout<<ret<<endl;
```

### 图像分析

```
//单个图片 Url
TagDetectReq tagReq(BUCKET);
```

```
tagReq.SetUrl("http://img3.a0bi.com/upload/ttq/20160814/1471155260063.png");
ret = image.TagDetect(tagReq);
cout<<ret<<endl;
//单个图片 File
TagDetectReq tagReq(BUCKET);
tagReq.SetImage("hot1.jpg");
ret = image.TagDetect(tagReq);
cout<<ret<<endl;
```

## OCR - 身份证识别

```
//单个或多个图片 Url,识别身份证正面
vector<string> idZUrls;
idZUrls.push_back("http://imgs.focus.cn/upload/sz/5876/a_58758051.jpg");
idZUrls.push_back("http://img5.iqilu.com/c/u/2013/0530/1369896921237.jpg");
IdCardOcrReq idReq(BUCKET, idZUrls,0);
ret = image.IdCardOcr(idReq);
cout<<ret<<endl;
//单个或多个图片 File,识别身份证正面
map<string, string> idZImages;
idZImages["id6zheng.jpg"] = FileUtil::getFileContent("id6zheng.jpg");
idZImages["id2zheng.jpg"] = FileUtil::getFileContent("id2zheng.jpg");
IdCardOcrReq idReq2(BUCKET, idZImages, 0);
ret = image.IdCardOcr(idReq2);
cout<<ret<<endl;
//单个或多个图片 Url,识别身份证反面
vector<string> idFUrls; idFUrls.push_back("http://www.csx.gov.cn/cwfw/bszn/201403/W020121030349
825312574.jpg"); idFUrls.push_back("http://www.4009951551.com/upload/image/20151026/14458311
36187479.png");
IdCardOcrReq idReq3(BUCKET, idFUrls,1);
ret = image.IdCardOcr(idReq3);
cout<<ret<<endl;
//单个或多个图片 File,识别身份证反面
map<string, string> idFImages;
idFImages["id5fan.jpg"] = FileUtil::getFileContent("id5fan.jpg");
idFImages["id7fan.jpg"] = FileUtil::getFileContent("id7fan.jpg");
IdCardOcrReq idReq4(BUCKET, idFImages, 1);
ret = image.IdCardOcr(idReq4);
cout<<ret<<endl;
```

## OCR - 名片识别

```
//单个或多个图片 Url
vector<string> nameUrls;
```

```
nameUrls.push_back("http://pic1.nipic.com/2008-12-03/2008123181119306_2.jpg");
nameUrls.push_back("http://pic.58pic.com/58pic/12/49/04/80k58PICzYP.jpg");
NameCardOcrReq nameReq(BUCKET, nameUrls, 0);
ret = image.NameCardOcr(nameReq);
cout<<ret<<endl;
//单个或多个图片 File
map<string, string> nameImages;
nameImages["r.jpg"] = FileUtil::getFileContent("r.jpg");
nameImages["name2.jpg"] = FileUtil::getFileContent("name2.jpg");
NameCardOcrReq nameReq2(BUCKET, nameImages, 0);
ret = image.NameCardOcr(nameReq2);
cout<<ret<<endl;
```

# PHP-SDK-图像识别

最近更新时间：2019-06-14 18:42:23

## 开发准备

### SDK 获取

智能图像的 PHP SDK 下载地址：[PHP-SDK-V2.0](#)。

## 快速入门

### 在腾讯云申请业务的授权

授权包括：APPID、SecretId、SecretKey，目前只支持主账号及密钥进行调用。

#### ⚠ 注意：

BUCKET 为历史遗留字段，无需修改。

### 创建对应操作类的对象

如果要使用图片，需要创建图片操作类对象：

```
require_once DIR . '/index.php';
use QcloudImage\CIClient;
$client = new CIClient('APP_ID', 'SECRET_ID', 'SECRET_KEY', 'BUCKET');
$client->setTimeout(30);
```

### 调用对应的方法

创建完对象后，您可以根据实际需求调用对应的操作方法。

### 图片识别

图片识别包括：图片鉴黄、图像分析、OCR - 身份证识别及 OCR - 名片识别。

#### 图片鉴黄

```
//单个或多个图片 URL
var_dump($client->pornDetect(array('urls'=>array('http://img3.a0bi.com/upload/ttq/20160814/1471155260063.png',
```

```
"http://jiangsu.china.com.cn/uploadfile/2015/1102/1446443026382534.jpg"))));  
//单个或多个图片 file  
var_dump ($client->pornDetect(array('files'=>array('F:\pic\您好.jpg','G:\pic\test2.jpg'))));
```

## 图像分析

```
//单个图片 URL  
var_dump ($client->tagDetect(array('url'=>'http://img3.a0bi.com/upload/ttq/20160814/1471155260  
063.png')));  
//单个图片 file  
var_dump ($client->tagDetect(array('file'=>'G:\pic\hot1.jpg')));  
//单个图片内容  
var_dump ($client->tagDetect(array('buffer'=>file_get_contents('G:\pic\hot1.jpg'))));
```

## OCR - 身份证识别

```
//单个或多个图片 URL , 识别身份证正面  
var_dump ($client->idcardDetect(array('urls'=>array('http://imgs.focus.cn/upload/sz/5876/a_587580  
51.jpg', 'http://img5.iqilu.com/c/u/2013/0530/1369896921237.jpg'), 0));  
//单个或多个图片 file , 识别身份证正面  
var_dump ($client->idcardDetect(array('files'=>array('F:\pic\id6zheng.jpg', 'F:\pic\id2zheng.jpg'), 0  
));  
//单个或多个图片内容 , 识别身份证正面  
var_dump ($client->idcardDetect(array('buffers'=>array(file_get_contents('F:\pic\id6_zheng.jpg'),  
file_get_contents('F:\pic\id2_zheng.jpg')), 0));  
//单个或多个图片 URL , 识别身份证反面  
var_dump ($client->idcardDetect(array('urls'=>array('http://www.csx.gov.cn/cwfw/bszn/201403/W02  
0121030349825312574.jpg',  
'http://www.4009951551.com/upload/image/20151026/1445831136187479.png')), 1));  
//单个或多个图片 file , 识别身份证反面  
var_dump ($client->idcardDetect(array('files'=>array('F:\pic\id5fan.jpg', 'F:\pic\id7fan.png')), 1));  
//单个或多个图片内容 , 识别身份证反面  
var_dump ($client->idcardDetect(array('buffers'=>array(file_get_contents('F:\pic\id5_fan.jpg'),  
file_get_contents('F:\pic\id7_fan.jpg')), 1));
```

## OCR - 名片识别

```
//单个或多个图片 HRL  
var_dump ($client->namecardV2Detect(array('urls'=>array('http://open.youtu.qq.com/app/img/exp  
erience/char_general/ocr_namecard_01.jpg'))));  
//单个或多个图片 file  
var_dump ($client->namecardV2Detect(array('files'=>array('assets/ocr_namecard_01.jpg'))));  
//单个或多个图片内容
```

```
var_dump ($client->namecardV2Detect(array('buffers'=>array(file_get_contents('assets/ocr_namecard_01.jpg'))));
```

# Node-SDK-图像识别

最近更新时间：2019-03-21 15:34:30

## 开发准备

### SDK 获取

智能图像的 Node SDK 下载地址：[Node-SDK](#)。

## 快速入门

### 在腾讯云申请业务的授权

授权包括：APPID、SecretId、SecretKey，目前只支持主账号及密钥进行调用。

### 安装使用

```
npm i --save image-node-sdk
```

以 OCR-身份证识别 为例，一般支持外链 url 或者本地读取图片文件这两种方式。

- 外链 url

```
const {
  ImageClient
} = require('image-node-sdk');

let AppId = ''; // 腾讯云 AppId
let SecretId = ''; // 腾讯云 SecretId
let SecretKey = ''; // 腾讯云 SecretKey

let idCardImageUrl = 'http://images.cnitblog.com/blog/454646/201306/07090518-029ff26fac014d72a7786937e8319c78.jpg';
let imgClient = new ImageClient({ AppId, SecretId, SecretKey });
imgClient.ocrIdCard({
  data: {
    url_list: [idCardImageUrl]
  }
}).then((result) => {
  console.log(result.body)
}).catch((e) => {
```



```
console.log(e);
});
```

- 读取本地文件

```
const fs = require('fs');
const path = require('path');
const {
  ImageClient
} = require('image-node-sdk');

let AppId = ""; // 腾讯云 AppId
let SecretId = ""; // 腾讯云 SecretId
let SecretKey = ""; // 腾讯云 SecretKey

let imgClient = new ImageClient({ AppId, SecretId, SecretKey });
imgClient.ocrIdCard({
  formData: {
    card_type: 0,
    image: fs.createReadStream(path.join(__dirname, './idcard.jpg'))
  },
  headers: {
    'content-type': 'multipart/form-data'
  }
}).then((result) => {
  console.log(result.body)
}).catch((e) => {
  console.log(e);
});
```

如果想运行，example/index.js 下面的例子，请先在项目根目录新建 config/index.js 文件，并按以下格式写下配置：

```
const ProxyUrl = ""; // 可填公司代理
const AppId = ""; // 腾讯云 AppId
const SecretId = ""; // 腾讯云 SecretId
const SecretKey = ""; // 腾讯云 SecretKey

exports.ProxyUrl = ProxyUrl;
exports.AppId = AppId;
exports.SecretId = SecretId;
exports.SecretKey = SecretKey;
```

然后运行：

```
npm run example
```

## 支持功能

- 文字识别 OCR
  - [手写体识别](#) - ocrHandWriting
  - [身份证识别](#) - ocrIdCard
  - [营业执照识别](#) - ocrBizLicense
  - [行驶证驾驶证识别](#) - ocrDrivingLicence
  - [车牌号识别](#) - ocrPlate
  - [通用印刷体识别](#) - ocrGeneral
  - [银行卡识别](#) - ocrBankCard
  - [名片识别 \(V2\)](#) - ocrBizCard
- 图片识别
  - [图片标签](#) - imgTagDetect
  - [图片鉴黄](#) - imgPornDetect