## 图像识别

SDK 参考

文档版本01发布日期2024-12-05





#### 版权所有 © 华为技术有限公司 2024。保留一切权利。

非经本公司书面许可,任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部,并不得以任何形式传播。

#### 商标声明

#### 注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束,本文档中描述的全部或部分产品、服务或 特性可能不在您的购买或使用范围之内。除非合同另有约定,华为公司对本文档内容不做任何明示或暗示的声 明或保证。

由于产品版本升级或其他原因,本文档内容会不定期进行更新。除非另有约定,本文档仅作为使用指导,本文 档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

### 华为技术有限公司

地址: 深圳市龙岗区坂田华为总部办公楼 邮编: 518129

网址: <u>https://www.huawei.com</u>

客户服务邮箱: <u>support@huawei.com</u>

客户服务电话: 4008302118

## 安全声明

#### 漏洞处理流程

华为公司对产品漏洞管理的规定以"漏洞处理流程"为准,该流程的详细内容请参见如下网址: https://www.huawei.com/cn/psirt/vul-response-process 如企业客户须获取漏洞信息,请参见如下网址: https://securitybulletin.huawei.com/enterprise/cn/security-advisory

| 1 图像识别 SDK 间介      |    |
|--------------------|----|
| 2 申请服务             | 2  |
| 3 获取认证信息           | 3  |
| 4 准备环境             | 4  |
| 5 SDK 列表及指导文档(通用版) | 5  |
| 6 使用 SDK(Java)     | 6  |
| 6.1 Java 开发环境配置    | 6  |
| 6.2 媒资图像标签示例       | 7  |
| 6.3 名人识别示例         | 10 |
| 6.4 主体识别示例         | 11 |
| 6.5 翻拍识别示例         | 12 |
| 6.6 图像标签示例         | 14 |
| 7 使用 SDK(Python)   | 16 |
| 7.1 Python 开发环境配置  | 16 |
| 7.2 媒资图像标签示例       |    |
| 7.3 名人识别示例         | 19 |
| 7.4 主体识别示例         | 20 |
| 7.5 翻拍识别示例         | 21 |
| 7.6 图像标签示例         | 22 |
| 8 使用 SDK(.NET)     | 23 |
| 8.1 .NET 开发环境配置    |    |
| 8.2 媒资图像标签示例       |    |
| 8.3 名人识别示例         | 25 |
| 8.4 主体识别示例         | 27 |
| 8.5 翻拍识别示例         |    |
| 8.6 图像标签示例         |    |
| 9 使用 SDK(GO)       |    |
| 9.1 GO 开发环境配置      |    |
| 9.2 媒资图像标签示例       |    |
| 9.3 名人识别示例         | 34 |
|                    |    |

| 9.4 主体识别示例      | 35 |
|-----------------|----|
| 9.5 翻拍识别示例      |    |
| 9.6 图像标签示例      | 37 |
| 10 使用 SDK(PHP)  |    |
| 10.1 PHP 开发环境配置 |    |
| 10.2 媒资图像标签示例   | 41 |
| 10.3 翻拍识别示例     |    |
| 10.4 图像标签示例     |    |
| 11 使用 SDK(C++)  |    |
| 11.1 C++开发环境配置  | 46 |
| 11.2 媒资图像标签示例   | 50 |
| 11.3 翻拍识别示例     | 51 |
| 11.4 图像标签示例     |    |
|                 |    |

目录

# 图像识别 SDK 简介

#### 图像识别概述

图像识别(Image Recognition ),是指利用计算机对图像进行分析和理解,以识别各 种不同模式的目标和对象的技术,包括媒资图像标签,名人识别,主体识别,翻拍识 别,图像标签等。

图像识别以开放API(Application Programming Interface,应用程序编程接口)的方 式提供给用户,用户通过实时访问和调用API获取推理结果,帮助用户自动采集关键数 据,打造智能化业务系统,提升业务效率。

#### SDK 概述

图像识别软件开发工具包(Image Recognition Software Development Kit,简称 Image SDK)是对图像识别提供的REST API进行的封装,以简化用户的开发工作。用 户通过添加依赖或下载的方式调用API即可实现使用图像识别业务能力的目的。

#### 接口与 API 对应关系

图像识别接口与API对应关系请参见表1-1。

表 1-1 接口与 API 对应关系表

| 接口     | ΑΡΙ   |
|--------|---|
| 媒资图像标签 | POST /v2/{project_id}/image/media-tagging             |
| 名人识别   | POST /v2/{project_id}/image/celebrity-recognition     |
| 主体识别   | POST /v3/{project_id}/image/main-object-<br>detection |
| 翻拍识别   | POST /v2/{project_id}/image/recapture-detect          |
| 图像标签   | POST /v2/{project_id}/image/tagging                   |



申请图像识别服务的具体操作步骤请参见《图像识别API参考》的"如何调用API>申 请服务"章节。

# 3 获取认证信息

使用服务API需要进行认证,具体方法如下:

- 步骤1 注册并登录华为云管理控制台。
- 步骤2 鼠标移动至用户名处,在下拉列表中单击"我的凭证"。
- 步骤3 选择"访问密钥"页签,单击"新增访问密钥"按钮。
- 步骤4 通过邮箱或者手机进行验证,输入对应的验证码。
- **步骤5** 单击"确定",下载认证账号的AK/SK,AK/SK数据会以本地文件的形式保存,请妥善保管。

-----结束



在使用图像识别SDK时,各语言需要准备的环境参照表4-1。

| 开发语言           | 准备项      | 说明   |
|----------------|----------|--|
| JAV<br>A       | 安装JDK    | JAVA环境开发配置,支持 Java JDK 1.8 及以上版本,<br>推荐通过Maven 安装依赖的方式使用JAVA版本SDK。 |
| PYT<br>HO<br>N | 安装python | python版本sdk支持python3.3及以上版本。                                       |
| .NE<br>T       | 安装.NET   | .NET Standard 2.0 及其以上版本或C# 4.0 及其以上版<br>本 。                       |
| GO             | 安装go     | 支持 go 1.14 及以上版本。  |
| PH<br>P        | 安装PHP    | 支持PHP 5.6 及以上版本。   |
| C++            | 安装C++    | 支持 C++ 14 及以上版本,要求安装 CMake 3.10 及以<br>上版本 。                        |

**表 4-1** 开发环境

# 5 SDK 列表及指导文档(通用版)

API Explorer能根据需要动态生成SDK代码功能,降低您使用SDK的难度,推荐使用。

SDK中心 https://sdkcenter.developer.huaweicloud.com/zh-cn提供服务SDK依赖引入的方式。

表5-1 提供了各个云服务支持的SDK列表,您可以在GitHub仓库查看SDK更新历史、获取安装包以及查看指导文档进行配置。

| 编程语言   | github地址                      | 指导文档               | 视频指导               |
|--------|-------------------------------|--------------------|--------------------|
| JAVA   | huaweicloud-<br>sdk-java-v3   | Java SDK使用指导       | Java SDK视频指导       |
| PYTHON | huaweicloud-<br>sdk-python-v3 | Python SDK使用<br>指导 | Python SDK视频<br>指导 |
| .NET   | huaweicloud-<br>sdk-net-v3    | .Net SDK使用指导       | -                  |
| GO     | huaweicloud-<br>sdk-go-v3     | Go SDK使用指导         | -                  |
| РНР    | huaweicloud-<br>sdk-php-v3    | PHP SDK使用指导        | -                  |
| C++    | huaweicloud-<br>sdk-cpp-v3    | C++ SDK使用指导        | -                  |

#### 表 5-1 SDK 列表

## **6** 使用 SDK (Java)

## 6.1 Java 开发环境配置

#### 环境配置

使用图像识别Java版本SDK包,要求的JDK版本必须高于JDK8版本,以下步骤以win7环境配置JDK8 64位为例,若已经下载JDK并配置好环境请忽略本章节。

- 1. 下载JDK文件。
- 2. 下载完成后按照提示安装,位置自选,比如安装到本地C:\Program Files\Java \jdk1.8.0\_131。
- 配置Java环境变量:右键"计算机>属性>高级系统设置>环境变量",进行如下操作。
  - a. 新建系统变量JAVA\_HOME,变量值为实际JDK安装位置。
  - b. 在Path中添加%JAVA\_HOME%\bin;%JAVA\_HOME%\jre\bin(注意用英文 分号分隔 )。
  - c. 新建系统变量CLASSPATH,变量值为%JAVA\_HOME%\lib \dt.jar;%JAVA\_HOME%\lib\tools.jar 。
- 4. 打开命令行窗口,输入"java -version",显示如<mark>图6-1</mark>表示配置成功。

图 6-1 Java 版本信息

```
C:\>java -version
java version "1.8.0_131"
Java(TM) SE Runtime Environment (build 1.8.0_131-b11)
Java HotSpot(TM) 64-Bit Server VM (build 25.131-b11, mixed mode)
```

SDK 获取和安装

JAVA版本图像识别的SDK推荐您通过 Maven 安装依赖的方式使用华为云 Java SDK:

步骤1 首先您需要在您的操作系统中下载并安装Maven。

步骤2 安装配置完成后,输入命令"mvn -v",显示如下图 Maven版本信息 表示成功。

:∖>mvn −v

);\ Ppache Maven 3.3.9 (bb52d8502b132ec0a5a3f4c09453c07478323dc5; 2015-11-11T00:41:47+08:00) Maven home: D:\maven\apache-maven-3.3.9\bin\.. Java version: 1.8.0\_262, vendor: Huawei Technologies Co., Ltd Java home: D:\develop\jdk\_1.8\jre Default locale: zh\_CN, platform encoding: GBK DS name: "windows 10", version: "10.0", arch: "amd64", family: "dos"

**步骤3** Maven安装完成后,可根据开发需要,直接引入依赖到已有的Maven工程或先用开发工具创建Maven工程。

创建项目以idea开发工具为例(已有Maven 工程可跳过此步骤):

- 1. 打开IntelliJ IDEA 开发工具。
- 2. 单击File New project...
- 3. 在New Project弹窗单击-Maven-单击Next。
- 4. 输入GroupId和ArtifactId,单击Next。
- 5. 输入Project name 和 Project location,单击Finish。

步骤4 在Maven 项目的 pom.xml 文件加入相应的依赖项即可。

以引入最新版本的图像识别SDK为例,请在**SDK中心**获取最新的sdk包版本,替换代码中版本。

<dependency>

<groupId>com.huaweicloud.sdk</groupId>

<artifactId>huaweicloud-sdk-image</artifactId> <version>3.1.8</version> ///此处需要您输入最新的sdk包版本

</dependency>

示例代码因需要输出结果更直观,所以新增fastjson的依赖(如接入不涉及此项,可忽略并在示例代码中删除相关内容),依赖如下:

<dependency>

<groupId>com.alibaba</groupId>
<artifactId>fastjson</artifactId>
<version>1.2.70</version>

</dependency>

----结束

### 6.2 媒资图像标签示例

本章节对媒资图像标签AK/SK方式使用SDK进行示例说明。

- 媒资图像标签示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用 ImageMediaTaggingReq中的withUrl或withImage方法配置图像信息(image和 url参数二选一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端
   节点。
- 媒资图像标签调用示例代码如下: package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential; import com.huaweicloud.sdk.core.auth.BasicCredentials; import com.huaweicloud.sdk.core.exception.ConnectionException; import com.huaweicloud.sdk.core.exception.RequestTimeoutException; import com.huaweicloud.sdk.core.exception.ServiceResponseException; import com.huaweicloud.sdk.image.v2.region.ImageRegion; import com.huaweicloud.sdk.image.v2.\*;

```
import com.huaweicloud.sdk.image.v2.model.*;
     public class RunImageMediaTaggingSolution {
       public static void main(String[] args) {
         // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量
     中密文存放,使用时解密,确保安全
         // 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境
     变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
         String ak = System.getenv("HUAWEICLOUD_SDK_AK");
         String sk = System.getenv("HUAWEICLOUD_SDK_SK");
         ICredential auth = new BasicCredentials()
              .withAk(ak)
              .withSk(sk);
         ImageClient client = ImageClient.newBuilder()
              .withCredential(auth)
              .withRegion(ImageRegion.valueOf("cn-north-4")) //此处替换为您开通服务的区域,详情请查
     看地区和终端节点
              .build();
         RunImageMediaTaggingRequest request = new RunImageMediaTaggingRequest();
         ImageMediaTaggingReq body = new ImageMediaTaggingReq();
         body.withThreshold(10f);
         body.withLanguage("zh");
         body.withUrl("https://XXX.jpg"); //此处替换为公网可以访问的图片地址
         request.withBody(body);
         try {
            RunImageMediaTaggingResponse response = client.runImageMediaTagging(request);
            System.out.println(response.toString());
         } catch (ConnectionException e) {
            e.printStackTrace();
         } catch (RequestTimeoutException e) {
            e.printStackTrace();
         } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
         }
       }
     }
2.
    执行示例代码文件,控制台输出媒资图像标签结果即表示程序执行成功。相关参
     数说明请参考<mark>媒资图像标签</mark>API。
     class RunImageMediaTaggingResponse {
       result: class ImageMediaTaggingResponseResult {
         tags: [class ImageMediaTaggingItemBody {
            confidence: 83.15
            type: 社会
            tag: 火灾
            i18nTag: class ImageMediaTaggingItemBodyI18nTag {
              zh: 火灾
              en: Fire
            i18nType: class ImageMediaTaggingItemBodyI18nType {
              zh: 社会
              en: Society
            instances: []
         }, class ImageMediaTaggingItemBody {
            confidence: 70.21
            type: 其他
            taq:火
            i18nTag: class ImageMediaTaggingItemBodyI18nTag {
              zh: 火
              en: Fire
            i18nType: class ImageMediaTaggingItemBodyI18nType {
              zh: 其他
              en: Other
```

instances: [] }, class ImageMediaTaggingItemBody { confidence: 53.63 type: 学科科学 tag: 火球 i18nTag: class ImageMediaTaggingItemBodyI18nTag { zh: 火球 en: Fireball i18nType: class ImageMediaTaggingItemBodyI18nType { zh: 学科科学 en: Subject science } instances: [] }, class ImageMediaTaggingItemBody { confidence: 53.45 type: 其他 tag: 营火 i18nTag: class ImageMediaTaggingItemBodyI18nTag { zh: 营火 en: Campfire } i18nType: class ImageMediaTaggingItemBodyI18nType { zh: 其他 en: Other } instances: [] }, class ImageMediaTaggingItemBody { confidence: 50.73 type: 人工场景 tag: 林业局 i18nTag: class ImageMediaTaggingItemBodyI18nTag { zh: 林业局 en: Forestry bureau i18nType: class ImageMediaTaggingItemBodyI18nType { zh: 人工场景 en: Manual scenario instances: [] }, class ImageMediaTaggingItemBody { confidence: 50.34 type: 其他 tag: 烧毁的 i18nTag: class ImageMediaTaggingItemBodyI18nTag { zh: 烧毁的 en: Burned } i18nType: class ImageMediaTaggingItemBodyI18nType { zh: 其他 en: Other } instances: [] }, class ImageMediaTaggingItemBody { confidence: 46.63 type: 场景 tag: 火焰 i18nTag: class ImageMediaTaggingItemBodyI18nTag { zh: 火焰 en: Flame i18nType: class ImageMediaTaggingItemBodyI18nType { zh: 场景 en: Scene } instances: [] }, class ImageMediaTaggingItemBody { confidence: 45.27

| type: 其他  |
|---|
| tag: 燃烧   |
| i18nTag: class ImageMediaTaggingItemBodyI18nTag {                     |
| zh: 燃烧  |
| en: Burning   |
| }   |
| i18nType: class ImageMediaTaggingItemBodyI18nType {                   |
| zh·甘他   |
| en: Other   |
| l   |
| j<br>instances: []  |
| Class ImageMediaTaggingItemBody {                                     |
| s, class intragenteura raggingitembouy (                              |
| tuno: IIIIIII/位合  |
| type. 职业/用巴   |
| ldy: )月初页<br>i10Ten: alege lage an Madia Tenesia altera Dadul10Tene ( |
| TISHIAG: Class Imageiviedia laggingitembodyi ishiag {                 |
| ZN: 消防灾   |
| en: Fireman   |
|   |
| 118hType: class ImageMediaTaggingItemBody[18hType {                   |
| zh: 职业/角色   |
| en: Career/Role   |
| }   |
| instances: []   |
| <pre>}, class ImageMediaTaggingItemBody {</pre>                       |
| confidence: 39.97   |
| type: 动作  |
| tag: 烧香   |
| i18nTag: class ImageMediaTaggingItemBodyI18nTag {                     |
| zh: 烧香  |
| en: Incense burning   |
| }   |
| i18nType: class ImageMediaTaggingItemBodyI18nType {                   |
| zh: 动作  |
| en: Action  |
| }   |
| instances: []   |
| }]  |
|   |
|   |

## 6.3 名人识别示例

} }

本章节对名人识别AK/SK方式使用SDK进行示例说明。

- 名人识别示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用
   CelebrityRecognitionReq中的withUrl或withImage方法配置图像信息(image和url参数二选一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端 节点。
- 名人识别调用示例代码如下: package com.huaweicloud.sdk.test;

import com.alibaba.fastjson.JSON;

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.image.v2.ImageClient;
import com.huaweicloud.sdk.image.v2.model.RunCelebrityRecognitionRequest;
import com.huaweicloud.sdk.image.v2.region.ImageRegion;
import com.huaweicloud.sdk.image.v2.model.CelebrityRecognitionReq;
import com.huaweicloud.sdk.image.v2.model.RunCelebrityRecognitionReq;
```

```
public class RunCelebrityRecognitionSolution {
       public static void main(String[] args) {
        // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量
     中密文存放,使用时解密,确保安全
        // 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境
     变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
         String ak = System.getenv("HUAWEICLOUD_SDK_AK");
         String sk = System.getenv("HUAWEICLOUD_SDK_SK");
         ICredential auth = new BasicCredentials()
              .withAk(ak)
              .withSk(sk);
         ImageClient client = ImageClient.newBuilder()
              .withCredential(auth)
              .withRegion(ImageRegion.valueOf("cn-north-1")) //此处替换为您开通服务的区域,详情请查
     看地区和终端节点
              .build():
         RunCelebrityRecognitionRequest request = new RunCelebrityRecognitionRequest();
         CelebrityRecognitionReq body = new CelebrityRecognitionReq();
         body.withThreshold(0f);
         body.withUrl("https://XXX.jpg"); //此处替换为公网可以访问的图片地址
         request.withBody(body);
         try {
           RunCelebrityRecognitionResponse response = client.runCelebrityRecognition(request);
           System.out.println(response.getHttpStatusCode());
           System.out.println(JSON.toJSONString(response));
         } catch (ConnectionException e) {
           e.printStackTrace();
         } catch (RequestTimeoutException e) {
           e.printStackTrace();
         } catch (ServiceResponseException e) {
           e.printStackTrace();
           System.out.println(e.getHttpStatusCode());
           System.out.println(e.getErrorCode());
           System.out.println(e.getErrorMsg());
         }
      }
    }
    执行示例代码文件,控制台输出名人识别结果即表示程序执行成功。相关参数说
2.
```

明请参考名人识别API。

200 {"httpStatusCode":200,"result":[{"confidence":0.9990679,"faceDetail": {"w":357,"h":510,"x":260,"y":201},"label":"迈克尔·杰克逊"}]} Process finished with exit code 0

## 6.4 主体识别示例

本章节对主体识别AK/SK方式使用SDK进行示例说明。

- 主体识别示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用 ImageMainObjectDetectionReq中的withUrl或withImage方法配置图像信息 (image和url参数二选一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端
   节点。
- 主体识别调用示例代码如下: package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential; import com.huaweicloud.sdk.core.auth.BasicCredentials; import com.huaweicloud.sdk.core.exception.ConnectionException;



## 6.5 翻拍识别示例

本章节对翻拍识别AK/SK方式使用SDK进行示例说明。

翻拍识别示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用 RunRecaptureDetectReg中的withUrl或withImage方法配置图像信息(image和 url参数二选一 ),示例中以url为例,配置完成后运行即可。 endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看<mark>地区和终端</mark> 节点。 翻拍识别调用示例代码如下: 1. package com.huaweicloud.sdk.test; import com.huaweicloud.sdk.core.auth.ICredential; import com.huaweicloud.sdk.core.auth.BasicCredentials; import com.huaweicloud.sdk.core.exception.ConnectionException; import com.huaweicloud.sdk.core.exception.RequestTimeoutException; import com.huaweicloud.sdk.core.exception.ServiceResponseException; import com.huaweicloud.sdk.image.v2.region.ImageRegion; import com.huaweicloud.sdk.image.v2.\*; import com.huaweicloud.sdk.image.v2.model.\*; public class RunRecaptureDetectSolution { public static void main(String[] args) { // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量 中密文存放,使用时解密,确保安全 // 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境 变量HUAWEICLOUD\_SDK\_AK和HUAWEICLOUD\_SDK\_SK String ak = System.getenv("HUAWEICLOUD\_SDK\_AK"); String sk = System.getenv("HUAWEICLOUD\_SDK\_SK"); ICredential auth = new BasicCredentials() .withAk(ak) .withSk(sk); ImageClient client = ImageClient.newBuilder() .withCredential(auth) .withRegion(ImageRegion.valueOf("cn-north-4")) //此处替换为您开通服务的区域,详情请查 看地区和终端节点 .build(); RunRecaptureDetectRequest request = new RunRecaptureDetectRequest(); RecaptureDetectReq body = new RecaptureDetectReq(); body.withThreshold(0f); body.withUrl("https://bucketname.obs.cn-north-4.myhuaweicloud.com/image/test1.jpg"); //替换 为公网可以访问的图片地址 request.withBody(body); try { RunRecaptureDetectResponse response = client.runRecaptureDetect(request); System.out.println(response.toString()); } catch (ConnectionException e) { e.printStackTrace(); } catch (RequestTimeoutException e) { e.printStackTrace(); } catch (ServiceResponseException e) { e.printStackTrace(); System.out.println(e.getHttpStatusCode()); System.out.println(e.getRequestId()); System.out.println(e.getErrorCode()); System.out.println(e.getErrorMsg()); } } 3 执行示例代码文件,控制台输出翻拍识别结果即表示程序执行成功。 2. class RunRecaptureDetectResponse { result: class RecaptureDetectResponseResult { suggestion: false category: recapture score: 0.9998 detail: [class RecaptureDetectResponseResultDetail { label: recapture

confidence: 0.9998 }] }

## 6.6 图像标签示例

本章节对图像标签AK/SK方式使用SDK进行示例说明。

- 图像标签示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用 ImageTaggingReq中的withUrl或withImage方法配置图像信息(image和url参数 二选一),示例中以url为例配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端
   节点。
- 1. 图像标签调用示例代码如下:

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.image.v2.region.ImageRegion;
import com.huaweicloud.sdk.image.v2.*;
import com.huaweicloud.sdk.image.v2.model.*;
public class RunImageTaggingSolution {
  public static void main(String[] args) {
    // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量
中密文存放,使用时解密,确保安全
    // 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境
变量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
    String ak = System.getenv("HUAWEICLOUD_SDK_AK");
    String sk = System.getenv("HUAWEICLOUD_SDK_SK");
    ICredential auth = new BasicCredentials()
         .withAk(ak)
         .withSk(sk);
    ImageClient client = ImageClient.newBuilder()
         .withCredential(auth)
         .withRegion(ImageRegion.valueOf("XXXX")) //此处替换为您开通服务的区域,详情请查看<mark>地</mark>
区和终端节点
         .build();
    RunImageTaggingRequest request = new RunImageTaggingRequest();
    ImageTaggingReq body = new ImageTaggingReq();
    body.withLimit(50);
    body.withThreshold(95f);
    body.withLanguage("zh");
    body.withUrl("https://XXX.jpg"); //此处替换为公网可以访问的图片地址
    request.withBody(body);
    try {
       RunImageTaggingResponse response = client.runImageTagging(request);
       System.out.println(response.toString());
    } catch (ConnectionException e) {
       e.printStackTrace();
    } catch (RequestTimeoutException e) {
       e.printStackTrace();
    } catch (ServiceResponseException e) {
       e.printStackTrace();
       System.out.println(e.getHttpStatusCode());
       System.out.println(e.getErrorCode());
       System.out.println(e.getErrorMsg());
    }
  }
```

}

执行示例代码,控制台返回200即表示程序执行成功。相关参数说明请参考图像标 2. 签API。

```
class RunImageTaggingResponse {
  result: class ImageTaggingResponseResult {
     tags: [class ImageTaggingItemBody {
       confidence: 98.00
        type: 树
        tag: 树
        i18nTag: class ImageTaggingItemBodyI18nTag {
          zh: 树
          en: Tree
        }
       i18nType: class ImageTaggingItemBodyI18nType {
zh: 树
          en: Tree
        3
        instances: []
     }]
  }
```

Process finished with exit code 0

}

## **7** 使用 SDK(Python)

## 7.1 Python 开发环境配置

#### 环境配置

使用图像识别Python版本SDK包,需要您配置Python开发环境。

- 1. 从**Python官网**下载并安装合适的Python版本。请使用Python3.3以上版本,如下 以Python3.7 版本为例进行说明。
- 2. 从PyCharm官网下载并安装最新版本。
- 3. 在PyCharm开发工具中配置Python环境,在菜单依次选择"File > Settings > Project Interpreter"。
- 4. 在页面上方选择您的Python安装路径,如图 **PyCharm配置python环境所示**。选择好目标Python之后单击页面下方"Apply"完成配置。

| Q,*                                 | Project: python → Project Inter   | oreter 🛛 🗟 For current project      |                |     |
|-------------------------------------|-----------------------------------|-------------------------------------|----------------|-----|
| > Appearance & Behavior             | Project Interpreter: 📑 Python 3.7 | (python) (1) D:\Python37\python.exe |                | ~ 🕸 |
| Keymap                              |                                   |                                     |                |     |
| Editor                              | Package                           | Version                             | Latest version | +   |
| Pluging                             | Click                             | 7.0                                 | 7.0            |     |
| Flugins                             | Flask                             | 1.0.2                               | ▲ 1.1.1        |     |
| > Version Control                   | Jinja2                            | 2.10.1                              | 2.10.1         |     |
| <ul> <li>Project: python</li> </ul> | MarkupSafe                        | 1.1.1                               | 1.1.1          | 0   |
| Project Interpreter                 | Werkzeug                          | 0.15.2                              | ▲ 0.15.6       |     |
|                                     | boto3                             | 1.9.147                             | ▲ 1.9.231      |     |
| Project Structure                   | botocore                          | 1.12.147                            | ▲ 1.12.231     |     |
| > Build, Execution, Deployment      | t certifi                         | 2019.6.16                           | ▲ 2019.9.11    |     |
| > Languages & Frameworks            | chardet                           | 3.0.4                               |                |     |
| Tools                               | docutils                          | 0.14                                |                |     |
| 7 10015                             | idna                              | 2.8                                 |                |     |
|                                     | itsdangerous                      | 1.1.0                               |                |     |
|                                     | jmespath                          | 0.9,4                               |                |     |
|                                     | numpy                             | 1.15.3                              |                |     |
|                                     | pip<br>authors distantil          | 20.0                                |                |     |
|                                     | python-dateutil                   | 2.0.0                               |                |     |
|                                     | setrensfer                        | 0.2.0                               |                |     |
|                                     | setuntools                        | 39.0.1                              |                |     |
|                                     | six                               | 1.12.0                              |                |     |
|                                     | urllib3                           | 1.24.3                              |                |     |
|                                     |                                   |                                     |                |     |
|                                     |                                   |                                     |                |     |
|                                     |                                   |                                     |                |     |
|                                     |                                   |                                     |                |     |
|                                     |                                   |                                     |                |     |
|                                     | Python packaging tools not found  | d. Install packaging tools          |                |     |
| 0                                   |                                   |                                     | OK Cancel      |     |

#### 图 7-1 PyCharm 配置 python 环境

#### SDK 获取和安装

python语言SDK的获取和安装推荐使用pip命令进行安装或使用pycharm进行安装,需 要安装huaweicloudsdkcore包以及huaweicloudsdkimage,参考方法如下:

pip 安装: # 安装核心库 pip install huaweicloudsdkcore

# 安装image服务库 pip install huaweicloudsdkimage

使用pycharm安装,步骤如下:

1. 打开pycharm,单击File -> Settings... 2. 单击Python Interpreter -> 单击+

3. 单击+,分别搜索huaweicloudsdkcore及huaweicloudsdkimage,搜索到包内容单击左下角Install Package完 成安装

| PC Settings                    |                                 |                                |  |        | ×   |
|--------------------------------|---------------------------------|--------------------------------|--|--------|-----|
| Q.                             | Project: spider > Python Int    | terpreter                      | project  |        |     |
| > Appearance & Behavior        | Python Interpreter: 🍦 Pytho     | n 3.7.2 (D:\Python\python37\pj | /thon.exe)   |        | - ¢ |
| Keymap                         |                                 |                                |  |        |     |
| > Editor                       | Package                         | Version                        | Latest ve  | rsion  | +   |
| Plugins                        | Cython                          | 0.29.14                        | ▲ 0.29.2   | 3      | -   |
| > Version Control              | IPy                             | 1.0                            | ▲ 1.01   |        |     |
| Project: spider                | ImageHash                       | 4.0                            | ▲ 4.2.0  |        |     |
| Python Interpreter 🛛 📾         | Available Packages              |                                |  |        | ×   |
| Project Structure              | Q+ huaweicloudsdkcore           |                                |  |        | ×   |
| > Build, Execution, Deployment | huawaislaudedkeara              |                                |  |        |     |
| > Languages & Frameworks       | nuaweicioudsdkcore              |                                | G Description  |        |     |
| > Tools                        |                                 |                                |  |        |     |
|                                | ☑ Install to user's site packag | yes directory (C:\Users\hwx621 | Specify version<br>Options<br>1781/AppData\Reaming/Pythc | 3.0.50 | v   |
| ?                              | Install Package Mana            | ge Repositories                |  |        |     |

#### 图 7-2 pycharm 安装图像识别 python 版本 sdk 包

## 7.2 媒资图像标签示例

本章节对媒资图像标签AK/SK方式使用SDK进行示例说明。

- 媒资图像标签示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用 ImageMediaTaggingReq中的url或image方法配置图像信息(image和url参数二 选一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端 节点。
- 媒资图像标签调用示例代码如下: # coding: utf-8

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkimage.v2.region.image_region import ImageRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkimage.v2 import *
if __name__ == "__main__":
  密文存放,使用时解密,确保安全;
  #本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
  ak = os.getenv("HUAWEICLOUD SDK AK")
 sk = os.getenv("HUAWEICLOUD_SDK_SK")
  credentials = BasicCredentials(ak, sk) \
  client = ImageClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ImageRegion.value_of("cn-north-4")) \ //此处替换为您开通服务的区域
    .build()
  try:
    request = RunImageMediaTaggingRequest()
    request.body = ImageMediaTaggingReq(
      threshold=10,
      language="zh"
      url="https://XXX.jpg" //此处替换为公网可以访问的图片地址
```

response = client.run\_image\_media\_tagging(request)
print(response)
except exceptions.ClientRequestException as e:
print(e.status\_code)
print(e.request\_id)
print(e.error\_code)
print(e.error\_msg)

执行示例代码文件,控制台输出媒资图像标签结果即表示程序执行成功。相关参数说明请参考媒资图像标签API。

数况明请参考保负国家你金AFT。 {"result": {"tags": [{"confidence": "83.15", "type": "社会", "tag": "火灾", "i18n\_tag": {"zh": "火灾", "en": "Fire"}, "i18n\_type": {"zh": "社会", "en": "Society"}, "instances": []}, {"confidence": "70.21", "type": "其 他", "tag": "火", "i18n\_tag": {"zh": "火", "en": "Fire"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "53.63", "type": "学科科学", "tag": "火球", "i18n\_tag": {"zh": "火球", "en": "Fireball"}, "i18n\_type": {"zh": "火", "en": "Subject science"}, "instances": []}, {"confidence": "53.45", "type": "其他", "tag": "营火", "i18n\_tag": {"zh": "支水", "en": "Campfire"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "50.73", "type": "人工场景", "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "50.73", "type": "人工场景", "tag": "林业局", "i18n\_tag": {"zh": "林业局", "en": "Forestry bureau"}, "i18n\_type": {"zh": "人工场景", "i18n\_type": {"zh": "烧毁的", "en": "Burned"}, "i18n\_type": {"zh": "大城", "tag": "K毁毁的", "i18n\_tag": {"zh": "烧毁的", "en": "Burned"}, "i18n\_type": {"zh": "火焰", "en": "Flame"}, "i18n\_type": {"zh": "城陽", "en": "Scene"}, "i18n\_type": {"zh": "大城", "tag": "大%", "en": "Gume"}, "i18n\_type": {"zh": "城陽", "en": "Scene"}, "i18n\_type": {"zh": "火焰", "en": "Flame"}, "i18n\_type": {"zh": "城陽", "en": "Scene"}, "i18n\_type": {"zh": "大焰", "en": "Flame"}, "i18n\_type": {"zh": "城陽", "en": "Scene"}, "i18n\_type": {"zh": "大焰", "en": "Flame"}, "i18n\_type": {"zh": "城陽", "en": "Scene"}, "i18n\_type": {"zh": "大焰", "en": "Flame"}, "i18n\_type": {"zh": "城B", "en": "Scene"}, "i18n\_type": {"zh": "jthe", "en": "Gume"}, "i18n\_type": {"zh": "城B", "en": "Scene"}, "i18n\_type": {"zh": "jthe", "en": "Gume"}, "i18n\_type": {"zh": "城B", "en": "Scene"}, "i18n\_type": {"zh": "jthe", "en": "Gume"}, "i18n\_type": {"zh": "jthe", "en": "Scene"}, "i18n\_type": {"zh": "jthe", "en": "Gume"}, "i18n\_type": {"zh": "jthe", "en": "Scene"}, "i18n\_tag": {"zh": "jthe", "i18n\_tag": {"zh": "jthe", "en": "Scene"}, "instances": []}, {"confidence": "161.27", "jthe"; "jthe", "i18n\_tag": {"zh": "jthe", "en"

## 7.3 名人识别示例

本章节对名人识别AK/SK方式使用SDK进行示例说明。

- 名人识别示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用初始化 CelebrityRecognitionReq中的url或image配置图像信息(image和url参数二选 一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端 节点。
- 名人识别调用示例代码如下: # coding: utf-8

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcore.http.http_config import HttpConfig
from huaweicloudsdkimage.v2 import
ImageClient,RunCelebrityRecognitionRequest,CelebrityRecognitionReq
from huaweicloudsdkimage.v2.region.image_region import ImageRegion
if __name__ == "__main__":
  # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量中
密文存放,使用时解密,确保安全;
  #本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
  ak = os.getenv("HUAWEICLOUD_SDK_AK")
  sk = os.getenv("HUAWEICLOUD_SDK_SK")
  credentials = BasicCredentials(ak, sk)
  client = ImageClient.new_builder() \
    .with_credentials(credentials)
    .with_region(ImageRegion.value_of("cn-north-1")) \ //此处替换为您开通服务的区域,详情请查看
地区和终端节点
    .build()
  try:
    request = RunCelebrityRecognitionRequest()
```



明请参考**名人识别API**。 200 {'result': [{'confidence': 0.9990678852332598, 'face\_detail': {'h': 510, 'w': 357, 'x': 260, 'y': 201}, 'label': '迈克尔·杰克逊'}]}

## 7.4 主体识别示例

2.

本章节对主体识别AK/SK方式使用SDK进行示例说明。

- 主体识别示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用初始化 ImageMainObjectDetectionReq中的url或image配置图像信息(image和url参数 二选一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端 节点。

#### 1. 主体识别调用示例代码如下:

```
# coding: utf-8
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkimage.v2.region.image_region import ImageRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkimage.v2 import *
```

```
if __name__ == "__main__":
  # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量中
密文存放,使用时解密,确保安全;
  #本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
  ak = os.getenv("HUAWEICLOUD_SDK_AK")
  sk = os.getenv("HUAWEICLOUD_SDK_SK")
  credentials = BasicCredentials(ak, sk) \
 client = ImageClient.new builder() \
    .with_credentials(credentials) \
    .with_region(ImageRegion.value_of("cn-north-4")) \ //此处替换为您开通服务的区域,详情请查看
地区和终端节点
    .build()
  try:
    request = RunImageMainObjectDetectionReguest()
    request.body = ImageMainObjectDetectionReq(
      threshold=30,
      url="https://XXX.jpg" //替换为公网可以访问的图片地址
    )
    response = client.run_image_main_object_detection(request)
    print(response)
  except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
```

```
print(e.error_code)
print(e.error_msg)
```

 执行示例代码文件,控制台输出主体识别结果即表示程序执行成功。相关参数说 明请参考主体识别API。

```
{
    "result": [{
    "label": "bounding_box",
    "location": {"height": 327.77, "top_left_x": 3.7, "top_left_y": 1.72, "width": 463.03},
    "confidence": "95.73"}, {"label": "main_object_box",
    "location": {"height": 328.35, "top_left_x": 3.22, "top_left_y": 1.5, "width": 456.63},
    "confidence": "88.35"}]
}
```

## 7.5 翻拍识别示例

本章节对翻拍识别AK/SK方式使用SDK进行示例说明。

- 翻拍识别示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用初始化 RunRecaptureDetectReq中的url或image配置图像信息(image和url参数二选 一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端
   节点。
- 翻拍识别调用示例代码如下: # coding: utf-8

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
    from huaweicloudsdkimage.v2.region.image_region import ImageRegion
    from huaweicloudsdkcore.exceptions import exceptions
    from huaweicloudsdkimage.v2 import *
    if __name__ == "__main__":
       # 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量中
    密文存放,使用时解密,确保安全;
       # 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
    量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
      ak = os.getenv("HUAWEICLOUD_SDK_AK")
      sk = os.getenv("HUAWEICLOUD_SDK_SK")
      credentials = BasicCredentials(ak, sk) \
      client = ImageClient.new_builder() \
         .with_credentials(credentials) \
         .with_region(ImageRegion.value_of("cn-north-4")) \ //此处替换为您开通服务的区域,详情请查看
    地区和终端节点.
         .build()
      try:
         request = RunRecaptureDetectRequest()
         request.body = RecaptureDetectReq(
           threshold=0,
           url="https://bucketname.obs.cn-north-4.myhuaweicloud.com/image/test1.jpg"//替换为公网可
    以访问的图片地址
         response = client.run_recapture_detect(request)
         print(response)
      except exceptions.ClientRequestException as e:
         print(e.status_code)
         print(e.request_id)
         print(e.error_code)
         print(e.error_msg)
   执行示例代码文件,控制台输出翻拍识别结果即表示程序执行成功。
2.
```

{"result":{"suggestion":"false","category":"recapture","score":"0.9998","detail": [{"label":"recapture","confidence":"0.9998"}]}}

## 7.6 图像标签示例

本章节对图像标签AK/SK方式使用SDK进行示例说明。

- 图像标签示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用初始化 ImageTaggingReq中的url或image配置图像信息(image和url参数二选一),示 例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端 节点。
- 图像标签调用示例代码如下: # coding: utf-8

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcore.http.http_config import HttpConfig
from huaweicloudsdkimage.v2 import ImageClient,RunImageTaggingRequest,ImageTaggingReq
from huaweicloudsdkimage.v2.region.image_region import ImageRegion
if __name__ == "__main__":
  密文存放,使用时解密,确保安全;
  #本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
  ak = os.getenv("HUAWEICLOUD_SDK_AK")
  sk = os.getenv("HUAWEICLOUD_SDK_SK")
  credentials = BasicCredentials(ak, sk)
  client = ImageClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ImageRegion.value_of("cn-north-1")) \ //此处替换为您开通服务的区域,详情请查看
地区和终端节点
    .build()
  try:
    request = RunImageTaggingRequest()
    request.body = ImageTaggingReq(
      limit=50,
      threshold=95,
      language="zh"
      url="https://XXX.jpg" //此处替换为公网可以访问的图片地址
    )
    response = client.run_image_tagging(request)
    print(response.status_code)
    print(response)
  except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

 执行示例代码,控制台返回200即表示程序执行成功。相关参数说明请参考图像标 签API。
 200

```
{"result": {"tags": [{"confidence": "98.00", "type": "树", "tag": "树", "i18n_tag": {"zh": "树", "en":
"Tree"}, "i18n_type": {"zh": "树", "en": "Tree"}, "instances": []}]}}
Process finished with exit code 0
```

# **8** 使用 SDK(.NET)

## 8.1 .NET 开发环境配置

#### 环境配置

使用图像识别.NET版本SDK包,需要先配置.Net开发环境。

- 1. 从Visual Studio官网,下载最新版Visual Studio。
- 2. 下载完毕后,单击exe文件,开始安装。
- 3. 安装过程中选择工作负荷ASP.NET和Web开发,安装完成后启动即可。

#### 图 8-1 选择安装.NET 环境依赖

| ASP.NET 和 Web 开発<br>使用 ASP.NET Core, ASP.NET, HTML/JavaScript 和包括<br>Docker 支持的容器生成 Web 应用程序。 | Azure 开設<br>用于使用 AIET 和 NET Framework 开放云应用和创建洗源<br>的 Azure SDK、工具和语言。 E包含用于实现应用程序号           | <ul> <li>ASP.NET 和 Web 开发</li> <li>已含音</li> <li>MET 开次工具</li> <li>MET Framework 4.7.2 开茨工具</li> </ul>   |
|---|---|---|
| Python 开发<br>分 Python 进行编辑、调试、交互式开发和源代码管理。  | Nodejs 开发<br>使用 Nodeja (一个曲用參學件驱动的 JavaScript 运行的)生<br>成可過始的同時在回程序。                           | <ul> <li>✓ ASP.NET 和 Web 开放工具</li> <li>✓ ASP.NET 和 Web 开放工具</li> <li>▼可达</li> <li>□.NET Framework 4 - 4.6 开放工具</li> <li>□.MET Framework 4 - 4.6 开放工具</li> </ul> |
| 面应用和移动应用 (5)  |   | IntelliTrace<br>.NET 分析工具   |
| NET 進動开发<br>守 C4、Visual Basic 和 F# 5、NET 和 NET Framework 一起<br>使用、生成 WPF、Windows 紫花和控制台应用程序。  | ** 使用 Fx+ 的桌面开发<br>使用 Fk注 II K(活路 MSVC, Clang, CMake 近 MSBuild)生成<br>送用 F Windows 的现代 C++ 空用。 | Entity Framework 6 工具<br>Live Unit Testing<br>高級ASP.NET 功能<br>Developer Analytics Tools<br>Web 部署   |
| <ul> <li>通用 Windows 平台开枝</li> <li>使用 C#、VB、並 C++ (可認)为通用 Windows 干台創建应用<br/>程序。</li> </ul>    | 使用 .NET 的修动开发<br>使用 Xemarin 对 iOS. Android 或 Windows 生成語平台应<br>用程序。                           | Live Share<br>IntelliCode<br>Windows Communication Foundation<br>年初时间IIS 主持   |

#### SDK 获取和安装

.NET语言SDK的获取和安装可以使用可以通过命令方式安装或使用Visual Studio开发工具安装第三方引用,参考方法如下:

使用 .NET CLI 工具 dotnet add package HuaweiCloud.SDK.Core dotnet add package HuaweiCloud.SDK.Image

使用 Package Manager Install-Package HuaweiCloud.SDK.Core Install-Package HuaweiCloud.SDK.Image 使用Visual Studio开发工具安装第三方引用:单击工具->选择NuGet包管理器->单击 管理解决方案的NuGet程序包,选择浏览中搜索并安装HuaweiCloud.SDK.Core及 HuaweiCloud.SDK.Image,如下图所示:

| 图 8-2 | 安装图像识别.NET | 版本 sdk 包 | IJ, |
|-------|------------|----------|-----|
|-------|------------|----------|-----|

| NuGet - 解决方案 + × NuGet ConsoleApp Program.cs      |        |             |  | ConsoleApp.csproj 🖬 🗙 👻 |
|---|--------|-------------|--|-------------------------|
| (112) 已安装 更新 合并                                   |        |             |  | 管理解决方案包                 |
| HuawwiCloudSDK.Image × - C (19255396736           |        |             |  | 程序包括 全部 · 〇             |
|   |        |             |  |                         |
| 1 HuaweiCloud.SDK.Image 作者 HarveiCloud. 5.73K 小下歌 | 3.0.52 | 2           | HuaweiCloud.SDK.Image  | 🍄 nugetorg              |
| HuaweiCloud .net SDK                              |        | 版書          | - 0  |                         |
|   |        |             | 项目   | 版本 已安装                  |
|   |        | 2           | ConsoleApp   |                         |
|   |        |             |  |                         |
|   |        |             |  |                         |
|   |        |             |  |                         |
|   |        | 1           |  | •                       |
|   |        | 89          | <b>2</b> +++4  | 509X                    |
|   |        | 15-8        | : 3.0.52   | • 9:00                  |
|   |        |             |  |                         |
|   |        |             | 建模   |                         |
|   |        |             |  |                         |
|   |        | Bird<br>Hua | weiCloud net SDK   |                         |
|   |        | 16.8        | 3052   |                         |
|   |        | 6.0         | t HuaweiCloud  |                         |
|   |        | 許可          | 证: 查看许可证   |                         |
|   |        | 下級          | 5,727  |                         |
|   |        | 3(6<br>8/5  | 日期: 2021年7月16日 (2021/7/16)<br>第冊間段: https://www.nupet.org/packages/Humme | Cloud SDK Image/3.0.52/ |
|   |        |             | ReportAbuse  |                         |
|   |        | 4 (630      | 硬<br>NETStandard,Version=v2.0  |                         |
|   |        |             | HuawelCloud.SDK.Core (>= 3.0.52)<br>Newtonsoft.Json (>= 9.0.1)           |                         |
|   |        | *           |  |                         |
| 每个包都由其所有者许可地均。NuGet 既不对第三方包改善,也不被予其许可证。<br>一      |        |             |  |                         |
|   |        | 1           |  |                         |

## 8.2 媒资图像标签示例

本章节对媒资图像标签AK/SK方式使用SDK进行示例说明。

- 媒资图像标签示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用 ImageMediaTaggingReq中的url或image方法配置图像信息(image和url参数二 选一),示例中以url为例,配置完成后运行即可。
- endnoint早化为一名服名应田区域和各服名的终端节占,详情请查看 地区和终端

| 12-( | 95) 版权所有 © 华为技术有限公司 24  |
|------|---|
|      | var client = ImageClient.NewBuilder()   |
|      | var autn = new BasicCredentials(aK, sk);  |
|      | config.lgnoreSslVerification = true;  |
|      | var config = HttpConfig.GetDefaultConfig();   |
|      | EnvironmentVariableTarget.Machine);   |
|      | EnvironmentVariableTarget.Machine);<br>string_secretKey= Environment GetEnvironmentVariable("HUAWEICLOUD_SDK_SK". |
|      | string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",  |
|      | //本示例以ak和lsk保存在坏境受量中来买现身份验证为例,运行本示例前请先在本地环境中设置环境变<br>量HUAWEICLOUD SDK AK和HUAWEICLOUD SDK SK                       |
|      |   |
|      | {<br>//认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量  |
|      | static void Main(string[] args)   |
|      | class Program<br>{  |
|      | {<br>   |
|      | namespace RunImageMediaTaggingSolution  |
|      | using HuaweiCloud.SDK.Image.V2.Model;   |
|      | using HuaweiCloud.SDK.Image.V2;   |
|      | using HuaweiCloud.SDK.Core.Auth;  |
|      | using HuaweiCloud.SDK.Core;   |
|      | using System;<br>using System.Collections.Generic:  |
| 1.   | 媒资图像标签调用示例代码如下:   |
|      | 节点。<br>节点。  |
|      | endpoint走千万云石服为应用区域相合服为的终端问点,许同间亘有 地区相终端  |

```
.WithCredential(auth)
            .WithRegion(ImageRegion.ValueOf("cn-north-4")) //此处替换为您开通服务的区域,详情
请查看地区和约
            .WithHttpConfig(config)
            .Build();
       var req = new RunImageMediaTaggingRequest
       ł
       };
       req.Body = new ImageMediaTaggingReq()
       Ł
          Threshold = 10,
          Language = "zh",
          Url = "https://XXX.jpg" //此处替换为公网可以访问的图片地址
       };
       try
       {
          var resp = client.RunImageMediaTagging(req);
          var respStatusCode = resp.HttpStatusCode;
          Console.WriteLine(respStatusCode);
Console.WriteLine(JsonConvert.DeserializeObject(resp.HttpBody));
       ł
       catch (RequestTimeoutException requestTimeoutException)
       {
          Console.WriteLine(requestTimeoutException.ErrorMessage);
       }
       catch (ServiceResponseException clientRequestException)
       {
          Console.WriteLine(clientRequestException.HttpStatusCode);
          Console.WriteLine(clientRequestException.ErrorCode);
          Console.WriteLine(clientRequestException.ErrorMsg);
       }
       catch (ConnectionException connectionException)
       Ł
          Console.WriteLine(connectionException.ErrorMessage);
       }
    }
 }
ļ
```

执行示例代码文件,控制台输出媒资图像标签结果即表示程序执行成功。相关参数说明请参考媒资图像标签API。
 200

{"result": {"tags": [{"confidence": "83.15", "type": "社会", "tag": "火灾", "i18n\_tag": {"zh": "火灾", "en": "Fire"}, "i18n\_type": {"zh": "社会", "en": "Society"}, "instances": []}, {"confidence": "70.21", "type": "其 他", "tag": "火", "i18n\_tag": {"zh": "火", "en": "Fire"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "53.63", "type": "学科科学", "tag": "火球", "i18n\_tag": {"zh": "火球", "en": "Fireball"}, "i18n\_type": {"zh": "学科科学", "en": "Subject science"}, "instances": []}, {"confidence": "53.45", "type": "其他", "tag": "营火", "i18n\_tag": {"zh": "支环", "en": "Campfire"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "50.73", "type": "人工场景", "tag": "林业局", "i18n\_tag": {"zh": "林业局", "en": "Forestry bureau"}, "i18n\_type": {\zh": "人工场景", "en": "Manual scenario"}, "instances": []}, {"confidence": "50.34", "type": "人工场景", "i18n\_tag": {"zh": "烧毁的", "en": "Burned"}, "i18n\_type": {\zh": "其他", "en": "Cher"}, "instances": []}, {"confidence": "46.63", "type": "场景", "tag": "火焰", "i18n\_tag": {"zh": "大焰", "en": "Flame"}, "i18n\_type": {\zh": "场景", "en": "Scene"}, "i18n\_tge": {\zh": "大焰", "en": "Flame"}, "i18n\_tag": {\zh": "场景", "en": "Burning"}, "i18n\_type": {\zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "46.63", "type": "其他", "tag": "沙焰", "i18n\_tge": {\zh": "其他", "en": "Flame"}, "i18n\_type": {\zh": "场景", "en": "Scene"}, "instances": []}, {"confidence": "45.27", "type": "其他", "tag": "燃烧", "i18n\_tag": {\zh": "太%," en": "Burning"}, "i18n\_tge": {\zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "41.32", "type": "职业/角色", "tag": "消防员", "i18n\_tag": {\zh": "其": "消防员", "i18n\_tag": {\zh": "其他", "en": "消防员", "i18n\_tge": {\zh": "其他", "tag": "於你,"en": "Career/Role"}, "i18n\_tag": {\zh": "消防员", "i18n\_tge": {\zh": "這 ", "en": "Fireman"}, "i18n\_type": {\zh": "动作", "tag": "烧香", "i18n\_tag": {\zh": "烧香", "en": "Incense burning"}, "i18n\_type": {\zh": "动作", "tag": "烧香", "i18n\_tag": {\zh": "Kē香", "en": "Incense burning"}, "i18n\_type": {\zh": "动作", "tag": "烧香", "i18n\_tag": {\zh":

## 8.3 名人识别示例

#### 本章节对名人识别AK/SK方式使用SDK进行示例说明。

```
名人识别示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用
    CelebrityRecognitionReg中的url或image方法配置图像信息(image和url参数二
    选一),示例中以url为例,配置完成后运行即可。
    endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看 地区和终端
    节点。
    名人识别调用示例代码如下:
1.
    using System;
    using System.Collections.Generic;
    using HuaweiCloud.SDK.Core;
    using HuaweiCloud.SDK.Core.Auth;
    using HuaweiCloud.SDK.Image;
    using HuaweiCloud.SDK.Image.V2;
    using HuaweiCloud.SDK.Image.V2.Model;
    using Newtonsoft.Json;
    namespace RunCelebrityRecognitionSolution
      class Program
      {
         static void Main(string[] args){
        //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量
    中密文存放,使用时解密,确保安全
        //本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
    量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
         string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
    EnvironmentVariableTarget.Machine);
         string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD SDK SK",
    EnvironmentVariableTarget.Machine);
           var config = HttpConfig.GetDefaultConfig();
           config.IgnoreSslVerification = true;
           var auth = new BasicCredentials(ak, sk);
           var client = ImageClient.NewBuilder()
               .WithCredential(auth)
                .WithRegion(ImageRegion.ValueOf("cn-north-1")) //此处替换为您开通服务的区域,详情
    请查看地区和终端节点
               .WithHttpConfig(config)
               .Build();
           var req = new RunCelebrityRecognitionRequest
           Ł
           };
           req.Body = new CelebrityRecognitionReq()
           {
             Threshold = 0.48,
             Url = "https://XXX.jpg" //此处替换为公网可以访问的图片地址
           };
           try
           {
             var resp = client.RunCelebrityRecognition(req);
             var respStatusCode = resp.HttpStatusCode;
             Console.WriteLine(respStatusCode);
             Console.WriteLine(JsonConvert.DeserializeObject(resp.HttpBody));
           }
           catch (RequestTimeoutException requestTimeoutException)
             Console.WriteLine(requestTimeoutException.ErrorMessage);
           }
           catch (ServiceResponseException clientRequestException)
           {
             Console.WriteLine(clientRequestException.HttpStatusCode);
             Console.WriteLine(clientRequestException.ErrorCode);
             Console.WriteLine(clientRequestException.ErrorMsq);
           }
           catch (ConnectionException connectionException)
```



## 8.4 主体识别示例

本章节对主体识别AK/SK方式使用SDK进行示例说明。

- 主体识别示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用 ImageMainObjectDetectionReq中的url或image方法配置图像信息(image和url 参数二选一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看 地区和终端 节点。

#### 1. 主体识别调用示例代码如下:

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Image;
using HuaweiCloud.SDK.Image.V2;
using HuaweiCloud.SDK.Image.V2.Model;
namespace RunImageMainObjectDetectionSolution
{
  class Program
    static void Main(string[] args) {
    //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量
中密文存放,使用时解密,确保安全
   //本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
    string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
EnvironmentVariableTarget.Machine);
    string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
EnvironmentVariableTarget.Machine);
      var config = HttpConfig.GetDefaultConfig();
      config.IgnoreSslVerification = true;
      var auth = new BasicCredentials(ak, sk);
      var client = ImageClient.NewBuilder()
           .WithCredential(auth)
           .WithRegion(ImageRegion.ValueOf("cn-north-4")) //此处替换为您开通服务的区域,详情
请查看地区和终端节点
           .WithHttpConfig(config)
           .Build();
```

```
var req = new RunImageMainObjectDetectionRequest
       ł
       };
       req.Body = new ImageMainObjectDetectionReq()
       {
          Threshold = 30,
          Url = "https://XXX.jpg" //替换为公网可以访问的图片地址
       };
       try
       {
          var resp = client.RunImageMainObjectDetection(req);
          var respStatusCode = resp.HttpStatusCode;
          Console.WriteLine(respStatusCode);
       }
       catch (RequestTimeoutException requestTimeoutException)
       {
          Console.WriteLine(requestTimeoutException.ErrorMessage);
       }
       catch (ServiceResponseException clientRequestException)
       {
          Console.WriteLine(clientRequestException.HttpStatusCode);
          Console.WriteLine(clientRequestException.ErrorCode);
          Console.WriteLine(clientRequestException.ErrorMsg);
       }
       catch (ConnectionException connectionException)
       ł
          Console.WriteLine(connectionException.ErrorMessage);
       }
    }
  }
}
执行示例代码文件,控制台输出主体识别结果即表示程序执行成功。相关参数说
明请参考主体识别API。
"result": [{
"confidence": "95.73",
"label": "bounding_box",
"location": {
"height": 327.77,
"top_left_x": 3.7,
"top_left_y": 1.72,
"width": 463.03}},
"confidence": "88.35",
"label": "main_object_box",
"location": {
"height": 328.35,
"top_left_x": 3.22,
"top_left_y": 1.5,
"width": 456.63}
}]
}
```

## 8.5 翻拍识别示例

2.

本章节对翻拍识别AK/SK方式使用SDK进行示例说明。

- 翻拍识别示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用 RunRecaptureDetectReq中的url或image方法配置图像信息(image和url参数二 选一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看 地区和终端 节点。

```
翻拍识别调用示例代码如下:
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Image;
using HuaweiCloud.SDK.Image.V2;
using HuaweiCloud.SDK.Image.V2.Model;
namespace RunRecaptureDetectSolution
{
  class Program
  ł
    static void Main(string[] args)
    //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量
中密文存放,使用时解密,确保安全
   //本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
    string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
EnvironmentVariableTarget.Machine);
    string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
EnvironmentVariableTarget.Machine);
       var config = HttpConfig.GetDefaultConfig();
       config.IgnoreSslVerification = true;
       var auth = new BasicCredentials(ak, sk);
       var client = ImageClient.NewBuilder()
            .WithCredential(auth)
            .WithRegion(ImageRegion.ValueOf("cn-north-4")) //此处替换为您开通服务的区域,详情
请查看地区和终端节点
            .WithHttpConfig(config)
           .Build();
       var req = new RunRecaptureDetectRequest
       Ł
       };
       req.Body = new RecaptureDetectReq()
       {
         Threshold = 0,
         Url = "https://bucketname.obs.cn-north-4.myhuaweicloud.com/image/test1.jpg"//替换为公
网可以访问的图片地址
       };
       try
       {
         var resp = client.RunRecaptureDetect(req);
         var respStatusCode = resp.HttpStatusCode;
         Console.WriteLine(respStatusCode);
          Console.WriteLine(resp.HttpBody);
       }
       catch (RequestTimeoutException requestTimeoutException)
       {
         Console.WriteLine(requestTimeoutException.ErrorMessage);
       }
       catch (ServiceResponseException clientRequestException)
       {
         Console.WriteLine(clientRequestException.HttpStatusCode);
         Console.WriteLine(clientRequestException.RequestId);
         Console.WriteLine(clientRequestException.ErrorCode);
         Console.WriteLine(clientRequestException.ErrorMsg);
       }
       catch (ConnectionException connectionException)
       ł
         Console.WriteLine(connectionException.ErrorMessage);
       }
    }
```

} }

2. 执行示例代码文件,控制台输出翻拍识别结果即表示程序执行成功。 200

{"result":{"suggestion":"false","category":"recapture","score":"0.9998","detail": [{"label":"recapture","confidence":"0.9998"}]}}

## 8.6 图像标签示例

本章节对图像标签AK/SK方式使用SDK进行示例说明。

- 图像标签示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用初始化 ImageTaggingReq中的url或image配置图像信息(image和url参数二选一),示 例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看 地区和终端 节点。
- 1. 图像标签调用示例代码如下:

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Image;
using HuaweiCloud.SDK.Image.V2;
using HuaweiCloud.SDK.Image.V2.Model;
using Newtonsoft.Json;
namespace RunImageTaggingSolution
{
  class Program
  {
    static void Main(string[] args) {
    //认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量
中密文存放,使用时解密,确保安全
   //本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK
    string accessKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK",
EnvironmentVariableTarget.Machine);
    string secretKey= Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK",
EnvironmentVariableTarget.Machine);
      var config = HttpConfig.GetDefaultConfig();
      config.IgnoreSslVerification = true;
      var auth = new BasicCredentials(ak, sk);
      var client = ImageClient.NewBuilder()
           .WithCredential(auth)
           .WithRegion(ImageRegion.ValueOf("cn-north-1")) //此处替换为您开通服务的区域,详情
请查看地区和终端节点.
           .WithHttpConfig(config)
           .Build();
      var req = new RunImageTaggingRequest
      Ł
      };
      req.Body = new ImageTaggingReq()
      {
         Limit = 50,
         Threshold = 95,
         Language = "zh",
         Url = "https://XXX.jpg" //此处替换为公网可以访问的图片地址
      };
      try
```

{



## **9** 使用 SDK(GO)

## 9.1 GO 开发环境配置

#### 准备工作

使用图像识别GO版本SDK包,需要您配置Go语言开发环境。如已经安装配置go开发 环境,可跳过该步骤。

- 1. 在golang 官网下载并安装对应的go版本,sdk支持 go 1.14 及以上版本。
- 2. 下载完成后按照提示安装,位置自选,比如安装到本地D:\develop\go。
- 3. 配置GO环境变量:右键"计算机>属性>高级系统设置>环境变量",进行如下操作。
  - a. 新建系统变量GOROOT,变量值为实际go安装位置。新增环境变量 GOPATH,变量为go语言工程所在目录。
  - b. 修改PATH变量,末尾增加GO的编译器路径 ";%GOROOT%\bin",中间分隔 符是【;】。
  - c. 修改CLASSPATH变量,末尾增加GO的编译器路径";%GOROOT%\lib",中间分隔符是【;】。
- 4. 在vs code官网,下载安装操作系统的vs code。
- 5. 安装go语言的插件Go和 Go Critic。

#### SDK 获取和安装

GO语言SDK的获取和安装使用命令进行安装,参考方法如下:

go get -u github.com/huaweicloud/huaweicloud-sdk-go-v3

## 9.2 媒资图像标签示例

本章节对媒资图像标签AK/SK方式使用SDK进行示例说明。

 媒资图像标签示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用 ImageMediaTaggingReq中的Url或Image方法配置图像信息(image和url参数二 选一),示例中以url为例,配置完成后运行即可。

```
endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看<mark>地区和终端</mark>
    节点。
1.
    媒资图像标签调用示例代码如下:
    package main
    import (
      "fmt'
      "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
      image "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2"
      "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2/model"
      region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2/region"
    )
    func main() {
      // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量中
    密文存放,使用时解密,确保安全;
      // 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
    量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
      ak := os.Getenv("HUAWEICLOUD_SDK_AK")
      sk := os.Getenv("HUAWEICLOUD_SDK_SK")
      auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()
      client := image.NewImageClient(
         image.ImageClientBuilder().
           WithRegion(region.ValueOf("cn-north-4")). //此处替换为您开通服务的区域,详情请查看<mark>地区和</mark>
     冬端节点
           WithCredential(auth).
           Build())
      request := &model.RunImageMediaTaggingRequest{}
      thresholdImageMediaTaggingReq:= float32(10)
      languageImageMediaTaggingReq:= "zh"
      urlImageMediaTaggingReq:= "https://XXX.jpg" //此处替换为公网可以访问的图片地址
      request.Body = &model.ImageMediaTaggingReg{
        Threshold: &thresholdImageMediaTaggingReq,
         Language: &languageImageMediaTaggingReq,
        Url: &urlImageMediaTaggingReq,
      response, err := client.RunImageMediaTagging(request)
      if err == nil {
        fmt.Printf("%+v\n", response)
      } else {
        fmt.Println(err)
      }
    }
    执行示例代码文件,控制台输出媒资图像标签结果即表示程序执行成功。相关参
```

#### 2. 数说明请参考媒资图像标签API。

RunImageMediaTaggingResponse {"result": {"tags": [{"confidence": "83.15", "type": "社会", "tag": "火 灾", "i18n\_tag": {"zh": "火灾", "en": "Fire"}, "i18n\_type": {"zh": "社会", "en": "Society"}, "instances": []}, {"confidence": "70.21", "type": "其他", "tag": "火", "i18n\_tag": {"zh": "火", "en": "Fire"}, "i18n\_type": {"zh": "其他", "en": "Fire"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "53.63", "type": "学科科学", "tag": "火球", "i18n\_tag": {"zh": "文球", "en": "Fireball"}, "i18n\_type": {"zh": "学科科学", "en": "Subject science"}, "i18n\_tag": {"zh": "火球", "en": "Fireball"}, "i18n\_type": {"zh": "字科科字", "en": "Subject science"}, "instances": []}, {"confidence": "53.45", "type": "其他", "tag": "营火", "i18n\_tag": {"zh": "营火", "en": "Campfire"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "50.73", "type": "人工场景", "tag": "林业局", "i18n\_tag": {"zh": "林业局", "en": "Forestry bureau"}, "i18n\_type": {"zh": "人工场景", "en": "Manual scenario"}, "instances": []}, {"confidence": "50.34", "type": "其他", "tag": "k 毁的", "i18n\_tag": {"zh": "烧毁的", "en": "Burned"}, "i18n\_type": {"zh": "其他", "tag": "其他", "tag": "人工场景", "en": "Other"}, "instances": []}, {"confidence": "50.34", "type": "其他", "tag": "烧 毁的", "i18n\_tag": {"zh": "烧毁的", "en": "Burned"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "46.63", "type": "场景", "tag": "火焰", "i18n\_tag": {"zh": "火焰", "en": "Flame"}, "i18n\_type": {"zh": "场景", "en": "Scene"}, "instances": []}, {"confidence": "45.27", "type": "其 ""utag": "i18n\_type": "i14", "i14", "witag": "j14", twpe", "i14", "utag": "i14", "ap": "I14", "utag": "I14", "Ene", "Ene", "Ene", "Enem", "Enem, "K Frame Y, fran\_type : { Zh : 场景 , eff : Scene Y, instances : [], { confidence : 45.27 , type : 他", "tag": "燃烧", "i18n\_tag": {"zh": "燃烧", "en": "Burning"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "41.32", "type": "职业/角色", "tag": "消防员", "i18n\_tag": {"zh": "消防员", "en": "Fireman"}, "i18n\_type": {"zh": "职业/角色", "en": "Career/Role"}, "instances":

[]}, {"confidence": "39.97", "type": "动作", "tag": "烧香", "i18n\_tag": {"zh": "烧香", "en": "Incense burning"}, "i18n\_type": {"zh": "动作", "en": "Action"}, "instances": []}]}}

## 9.3 名人识别示例

本章节对名人识别AK/SK方式使用SDK进行示例说明。

- 名人识别示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用初始化 CelebrityRecognitionReq中的Image或Url配置图像信息(image和url参数二选 一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端 节点。
- 1. 名人识别调用示例代码如下: package main

```
import (
       "fmt"
       "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
       image "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2"
       "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2/model"
       region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2/region"
     func main() {
     // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量中
密文存放,使用时解密,确保安全;
      // 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
     量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
      ak := os.Getenv("HUAWEICLOUD_SDK_AK")
      sk := os.Getenv("HUAWEICLOUD_SDK_SK")
       auth := basic.NewCredentialsBuilder().
         WithAk(ak).
         WithSk(sk).
         Build()
       client := image.NewImageClient(
         image.ImageClientBuilder().
            WithRegion(region.ValueOf("cn-north-1")). //此处替换为您开通服务的区域,详情请查看地区和
     终端节点
            WithCredential(auth).
            Build())
       request := &model.RunCelebrityRecognitionRequest{}
       thresholdCelebrityRecognitionReq:= float32(0.48)
       urlCelebrityRecognitionReq:= "https://XXX.jpg" //此处替换为公网可以访问的图片地址
       request.Body = &model.CelebrityRecognitionReq{
         Threshold: &thresholdCelebrityRecognitionReg,
         Url: &urlCelebrityRecognitionReq,
       }
       response, err := client.RunCelebrityRecognition(request)
       if err == nil {
         fmt.Printf("%+v\n", response)
       } else {
         fmt.Println(err)
       }
     }
    识别结果输出到控制台表示执行成功。相关参数说明请参考<mark>名人识别API</mark>。
2.
     RunCelebrityRecognitionResponse {"result":[{"confidence":0.9990679,"face_detail":
     {"h":510,"w":357,"x":260,"y":201},"label":"迈克尔·杰克逊"}]}
     Process exiting with code: 0
```

## 9.4 主体识别示例

本章节对主体识别AK/SK方式使用SDK进行示例说明。

- 主体识别示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用初始化 ImageMainObjectDetectionReq中的Image或Url配置图像信息(image和url参数 二选一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端 节点。
- 1. 主体识别调用示例代码如下: package main

```
import (
  "fmt"
  "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
  image "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2"
  "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2/model"
  region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2/region"
)
func main() {
 // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量中
密文存放,使用时解密,确保安全;
 // 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
 ak := os.Getenv("HUAWEICLOUD SDK AK")
 sk := os.Getenv("HUAWEICLOUD_SDK_SK")
  auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()
  client := image.NewImageClient(
    image.ImageClientBuilder().
       WithRegion(region.ValueOf("cn-north-4")). //此处替换为您开通服务的区域,详情请查看<mark>地区和</mark>
 §端节点
       WithCredential(auth).
       Build())
  request := &model.RunImageMainObjectDetectionRequest{}
  thresholdImageMainObjectDetectionReg:= float32(30)
  urlImageMainObjectDetectionReq:= "https://XXX.jpg"
                                                  //此处替换为公网可以访问的图片地址
  request.Body = &model.ImageMainObjectDetectionReq{
    Threshold: &thresholdImageMainObjectDetectionReg,
    Url: &urlImageMainObjectDetectionReq,
  }
  response, err := client.RunImageMainObjectDetection(request)
  if err == nil {
    fmt.Printf("%+v\n", response)
  } else {
    fmt.Println(err)
  }
}
识别结果输出到控制台表示执行成功。相关参数说明请参考主体识别API。
RunImageMainObjectDetectionResponse {"result":[{"label":"bounding_box","location":
{"height":327.77,"top_left_x":3.7,"top_left_y":1.72,"width":463.03},"confidence":"95.73"},
{"label":"main_object_box","location":
```

```
{"height":328.35,"top_left_x":3.22,"top_left_y":1.5,"width":456.63},"confidence":"88.35"}]}
```

## 9.5 翻拍识别示例

本章节对翻拍识别AK/SK方式使用SDK进行示例说明。

- 翻拍识别示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用初始化 RunRecaptureDetectReq中的Image或Url配置图像信息(image和url参数二选 一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端 节点。
- 1. 翻拍识别调用示例代码如下: package main

```
import (
  "fmt"
  "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
  image "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2"
  "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2/model"
  region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2/region"
)
func main() {
 // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量中
密文存放,使用时解密,确保安全;
 // 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
 ak := os.Getenv("HUAWEICLOUD SDK AK")
 sk := os.Getenv("HUAWEICLOUD_SDK_SK")
  auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()
  client := image.NewImageClient(
    image.ImageClientBuilder().
       WithRegion(region.ValueOf("cn-north-4")). //此处替换为您开通服务的区域,详情请查看<mark>地区和</mark>
 §端节点
       WithCredential(auth).
       Build())
  request := &model.RunRecaptureDetectRequest{}
  thresholdRecaptureDetectReg:= float32(0)
  urlRecaptureDetectReq:= "https://bucketname.obs.cn-north-4.myhuaweicloud.com/image/
test1.jpg"//替换为公网可以访问的图片地址
  request.Body = &model.RecaptureDetectReq{
    Threshold: &thresholdRecaptureDetectReq,
    Url: &urlRecaptureDetectReq,
  }
  response, err := client.RunRecaptureDetect(request)
  if err == nil {
    fmt.Printf("%+v\n", response)
  } else {
    fmt.Println(err)
  }
}
识别结果输出到控制台表示执行成功。
```

```
{"result":{"suggestion":"false","category":"recapture","score":"0.9998","detail":
[{"label":"recapture","confidence":"0.9998"}]}}
```

### 9.6 图像标签示例

本章节对图像标签AK/SK方式使用SDK进行示例说明。

- 图像标签示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用初始化 ImageTaggingReq中的Image或Url配置图像信息(image和url参数二选一),示 例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端 节点。
- 图像标签调用示例代码如下: package main

```
import (
  "fmt"
  "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
  image "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2"
  "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2/model"
  region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/image/v2/region"
)
func main() {
 // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量中
密文存放,使用时解密,确保安全;
 // 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
量HUAWEICLOUD_SDK_AK和HUAWEICLOUD_SDK_SK。
 ak := os.Getenv("HUAWEICLOUD SDK AK")
 sk := os.Getenv("HUAWEICLOUD_SDK_SK")
  auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()
  client := image.NewImageClient(
    image.ImageClientBuilder().
      WithRegion(region.ValueOf("cn-north-1")). //此处替换为您开通服务的区域,详情请查看<mark>地区和</mark>
 端节点
      WithCredential(auth).
      Build())
  request := &model.RunImageTaggingRequest{}
  limitImageTaggingReg:= int32(50)
  thresholdImageTaggingReq:= float32(95)
  languageImageTaggingReq:= "zh"
  urlImageTaggingReq:= "https://XXX.jpg" //此处替换为公网可以访问的图片地址
  request.Body = &model.ImageTaggingReq{
    Limit: &limitImageTaggingReq,
    Threshold: &thresholdImageTaggingReq,
    Language: &languageImageTaggingReq,
    Url: &urlImageTaggingReq,
  }
  response, err := client.RunImageTagging(request)
  if err == nil {
    fmt.Printf("%+v\n", response)
  } else {
    fmt.Println(err)
  }}
识别结果输出到控制台表示执行成功。相关参数说明请参考<mark>图像标签API</mark>。
```

# **10** 使用 SDK(PHP)

## 10.1 PHP 开发环境配置

#### 准备工作

- 注册华为账号并开通华为云,并完成实名认证,账号不能处于欠费或冻结状态。
- 已具备开发环境,PHP 5.6 及以上版本,可执行php --version检查当前的版本信息。
- 登录"我的凭证>访问密钥"页面,获取Access Key(AK)和Secret Access Key (SK)。

#### 图 10-1 获取 AK、SK



 登录"我的凭证"页面,获取"IAM用户名""、账号名"以及待使用区域的 "项目ID"。调用服务时会用到这些信息,请提前保存。

本样例以"华北-北京四"区域为例,获取对应的项目ID(project\_id )。

#### **图 10-2** 我的凭证

| 我的凭证   | APIțte ©  |                   |             |        |              |   |
|--------|---|-------------------|-------------|--------|--------------|---|
| APPELE | ● 和关472228号, MM子把^422目的把关信息, 请求未过至                          |                   |             |        |              | × |
|        | imaipa «J <sup>a</sup><br>Imaida (Marina) (Marina) (Marina) |                   | 账号码<br>账号ID | et     |              |   |
|        | 8ПИВ<br>8ПО ⊕<br>95ee                                       | 1811 <del>0</del> |             | MEXM 0 | 「朝鮮入び日本が担行地家 | Q |

#### 安装 SDK

推荐使用 Composer 安装 SDK 。

Composer 是 php 的依赖管理工具,允许您在项目中声明依赖关系并安装这些依赖:

// 安装 Composer curl -sS https://getcomposer.org/installer | php // 安装 PHP SDK composer require huaweicloud/huaweicloud-sdk-php

#### 安装完毕后,你需要引入 Composer 的自动加载文件:

require 'path/to/vendor/autoload.php';

#### 开始使用 SDK

1. 导入依赖模块

<?php namespace HuaweiCloud\SDK\Image\V2\Model; require\_once "vendor/autoload.php"; use HuaweiCloud\SDK\Core\Auth\BasicCredentials; use HuaweiCloud\SDK\Core\Exceptions\ConnectionException; use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException; use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException; use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException; use HuaweiCloud\SDK\Image\V2\ImageClient;

#### 2. 配置客户端连接参数

\_

- 默**认配<u>置</u> // 使用默认配置 \$config = HttpConfig::getDefaultConfig();**
- 网络代理(可选)
   //使用代理服务器
   \$config->setProxyProtocol('http');
   \$config->setProxyHost('proxy.huawei.com');
   \$config->setProxyPort(8080);
   \$config->setProxyUser('username');
   \$config->setProxyPassword('password');
- 超时配置(可选)
   // 默认连接超时时间为60秒,读取超时时间为120秒。可根据需要修改默认值。
   \$config->setTimeout(120);
   \$config->setConnectionTimeout(60);
  - SSL配置(可选) // 配置跳过服务端证书验证 \$config->setIgnoreSslVerification(true); // 配置服务器端CA证书,用于SDK验证服务端证书合法性 \$config->setCertFile("{yourCertFile}");
- 3. 配置认证信息

配置AK、SK、projectld信息。华为云通过AK识别用户的身份,通过SK对请求数 据进行签名验证,用于确保请求的机密性、完整性和请求者身份的正确性。 <sup>\$ak =</sup> getenv('HUAWEICLOUD\_SDK\_AK'); <sup>\$sk =</sup> getenv('HUAWEICLOUD\_SDK\_SK') <sup>\$projectld =</sup> getenv("PROJECT\_ID");; <sup>\$credentials =</sup> new BasicCredentials(\$ak,\$sk,\$projectId);

#### 认证参数说明:

- ak、sk:访问密钥信息,获取方法请参见<mark>准备工作</mark>。
- · projectld:华为云项目lD,获取方法请参见<mark>准备工作</mark>。
- endpoint: 华为云各服务应用区域和各服务的终端节点,详情请查看地区和
   终端节点。

⚠ 注意

- 认证用的 ak 和sk 硬编码到代码中或者明文存储都有很大的安全风险,建 议在配置文件或者环境变量中密文存放,使用时解密,确保安全。
- 本示例以 ak 和 sk 保存在环境变量中来实现身份验证为例,运行本示例前 请先在本地环境中设置环境变量HUAWEICLOUD\_SDK\_AK, HUAWEICLOUD\_SDK\_SK和PROJECT\_ID。

#### 图 10-3 Windows 环境新建环境变量

| 新建用户变量  |                    | × |
|---------|--------------------|---|
| 变量名(N): | HUAWEICLOUD_SDK_AK |   |
| 变量值(V): |                    |   |
| 浏览目录(D) | 浏览文件(F) 确定 取消      |   |

#### 4. 初始化客户端

5.

#### 指定云服务endpoint,初始化客户端

```
$endpoint = "https://image.cn-north-4.myhuaweicloud.com";
$client = ImageClient::newBuilder(new ImageClient)
 ->withHttpConfig($config)
 ->withEndpoint($endpoint)
 ->withCredentials($credentials)
->build();
发送并查看响应
$request = new RunImageMediaTaggingRequest();
$body = new ImageMediaTaggingReq();
$body->setLimit(10);
$body->setThreshold(60);
$body->setLanguage("en");
$body->setImage("图片Base64");
$request->setBody($body);
try {
 $response = $client->RunImageMediaTagging($request);
 echo "\n";
 echo $response;
} catch (ConnectionException $e) {
 $msg = $e->getMessage();
 echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
 $msg = $e->getMessage();
 echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
 echo "\n";
 echo $e->getHttpStatusCode(). "\n";
 echo $e->getRequestId(). "\n";
 echo $e->getErrorCode() . "\n";
 echo $e->getErrorMsg() . "\n";
}
```

6. 异常处理

#### **表 10-1** 异常处理

| 一级分类                         | 一级分类说明      | 二级分类                         | 二级分类说明                             |
|------------------------------|-------------|------------------------------|------------------------------------|
| ConnectionExcepti<br>on      | 连接类异常       | HostUnreachableE<br>xception | 网络不可达、被拒<br>绝。                     |
|                              |             | SslHandShakeExce<br>ption    | SSL认证异常。                           |
| RequestTimeoutEx ception     | 响应超时异常      | CallTimeoutExcept<br>ion     | 单次请求,服务器<br>处理超时未返回。               |
|                              |             | RetryOutageExcep<br>tion     | 在重试策略消耗完<br>成后,仍无有效的<br>响应。        |
| ServiceResponseEx<br>ception | 服务器响应异<br>常 | ServerResponseEx<br>ception  | 服务端内部错误,<br>Http响应码:<br>[500,] 。   |
|                              |             | ClientRequestExce<br>ption   | 请求参数不合法,<br>Http响应码: [400,<br>500) |

```
// 捕获和处理不同类型的异常
try {
 $response = $client->RunImageMediaTagging($request);
 echo "\n";
 echo $response;
} catch (ConnectionException $e) {
 $msg = $e->getMessage();
 echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
 $msg = $e->getMessage();
 echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
 echo "\n";
 echo $e->getHttpStatusCode(). "\n";
echo $e->getRrequestId(). "\n";
echo $e->getErrorCode(). "\n";
echo $e->getErrorMsg(). "\n";
}
```

#### 🗀 说明

使用异步客户端,配置日志等操作请参见SDK中心、PHP SDK使用指导、PHP SDK使用视频。

## 10.2 媒资图像标签示例

本章节对媒资图像标签AK/SK方式使用SDK进行示例说明。

- 媒资图像标签示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用 ImageMediaTaggingReq中的Url或Image方法配置图像信息(image和url参数二 选一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端 节点。

#### 1. 媒资图像标签调用示例代码如下:

<?php namespace HuaweiCloud\SDK\Image\V2\Model; require\_once "vendor/autoload.php"; use HuaweiCloud\SDK\Core\Auth\BasicCredentials; use HuaweiCloud\SDK\Core\Http\HttpConfig; use HuaweiCloud\SDK\Core\Exceptions\ConnectionException; use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException; use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException; use HuaweiCloud\SDK\Image\V2\ImageClient; // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量中 密文存放,使用时解密,确保安全; // 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变 量CLOUD\_SDK\_AK和CLOUD\_SDK\_SK。 \$ak = getenv('CLOUD\_SDK\_AK'); \$sk = getenv('CLOUD\_SDK\_SK'); //image.cn-north-4.myhuaweicloud.com替换为您开通服务区域的终端节点,详情请查看<mark>地区和终端节点</mark>. \$endpoint = "https://image.cn-north-4.myhuaweicloud.com"; \$projectId = ""; \$credentials = new BasicCredentials(\$ak,\$sk,\$projectId); \$config = HttpConfig::getDefaultConfig(); \$config->setIgnoreSslVerification(true); \$client = ImageClient::newBuilder(new ImageClient) ->withHttpConfig(\$config) ->withEndpoint(\$endpoint) ->withCredentials(\$credentials) ->build(); \$request = new RunImageMediaTaggingRequest(); \$body = new ImageMediaTaggingReq(); \$body->setLimit(10); \$body->setThreshold(60); \$body->setLanguage("zh"); \$body->setImage("输入图片,使用base64编码"); \$request->setBody(\$body); try { \$response = \$client->RunImageMediaTagging(\$request); echo "\n"; echo \$response; } catch (ConnectionException \$e) { \$msg = \$e->getMessage(); echo "\n". \$msg ."\n"; } catch (RequestTimeoutException \$e) { \$msg = \$e->getMessage(); echo "\n". \$msg ."\n"; } catch (ServiceResponseException \$e) { echo "\n"; echo \$e->getHttpStatusCode(). "\n"; echo \$e->getRequestId(). "\n"; echo \$e->getErrorCode() . "\n"; echo \$e->getErrorMsg() . "\n";

## 执行示例代码文件,控制台输出媒资图像标签结果即表示程序执行成功。相关参数说明请参考媒资图像标签API。

{"result": {"tags": [{"confidence": "83.15", "type": "社会", "tag": "火灾", "i18n\_tag": {"zh": "火灾", "en": "Fire"}, "i18n\_type": {"zh": "社会", "en": "Society"}, "instances": []}, {"confidence": "70.21", "type": "其 他", "tag": "火", "i18n\_tag": {"zh": "火", "en": "Fire"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []], {"confidence": "53.63", "type": "学科科学", "tag": "火球", "i18n\_tag": {"zh": "火球", "en": "Fireball"}, "i18n\_type": {"zh": "文", "en": "Fire"}, "i18n\_tag": {"zh": "其述", "y球", "instances": []], {"confidence": "53.63", "type": "学科科学", "en": "Subject science"}, "instances": []], {"confidence": "53.45", "type": "其他", "tag": "营火", "i18n\_tag": {"zh": "营火", "en": "Campfire"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []], {"confidence": "50.73", "type": "人工场景", "tag": "林业局", "i18n\_tag": {"zh": "林业局", "en": "Forestry bureau"}, "i18n\_type": {"zh": "人工场景", "i18n\_type": {"zh": "烧毁的", "en": "Burned"}, "i18n\_type": {"zh": "其他", "tag": "烧毁的", "i18n\_tag": {"zh": "烧毁的", "en": "Burned"}, "i18n\_type": {"zh": "火焰", "en": "Fieme"}, "i18n\_type": {"zh": "%%%", "en": "Burned"}, "i18n\_type": {"zh": "火焰", "en": "Fieme"}, "i18n\_type": {"zh": "%%%", "en": "Scene"}, "instances": []}, {"confidence": "45.27", "type": "其他", "tag": "%%%, "i18n\_tag": {"zh": "燃%, "en": "Burning"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "45.27", "type": "其他", "tag": "%%%, "i18n\_tag": {"zh": "%%%, "en": "Burning"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "41.32", "type": "Ruu/角色", "tag": "%% "i18n\_tag": {"zh": "jtm"; "instances": []}, {"confidence": "41.32", "type": "Ruu/角色", "tag": "jtm; "jtm; "i18n\_tag": {"zh": "jtm; "jtm; "instances": []}, {"confidence": "41.32", "type": "Ruu/角色", "tag": "jtm; "jtm; "jtm; "jtm; "instances": []}, {"confidence": "41.32", "type": "Ruu/角色", "tag": "jtm; "jtm; "jtm; "jtm; "jtm; "jtm; "jtm;", "i18n\_type": {"zh": "jtm;";"jtm;";"jtm;";"jtm;";"jtm;";"jtm;";"jtm;";"jtm;";"jtm;";"jtm;";"jtm;";"jtm;";"jtm;";"jtm;";"jtm;";"jtm;";"jtm;";"jtm {"confidence": "39.97", "type": "动作", "tag": "烧香", "i18n\_tag": {"zh": "烧香", "en": "Incense burning"}, "i18n\_type": {"zh": "动作", "en": "Action"}, "instances": []}]}}

## 10.3 翻拍识别示例

本章节对翻拍识别AK/SK方式使用SDK进行示例说明。

- 翻拍识别示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用初始化 RunRecaptureDetectReq中的Image或Url配置图像信息(image和url参数二选 一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端
   节点。
- 1. 翻拍识别调用示例代码如下:

```
<?php
```

```
namespace HuaweiCloud\SDK\Image\V2\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Image\V2\ImageClient;
 // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量中
密文存放,使用时解密,确保安全;
 // 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
量CLOUD_SDK_AK和CLOUD_SDK_SK。
$ak = getenv('CLOUD_SDK_AK');
$sk = getenv('CLOUD_SDK_SK');
//image.cn-north-4.myhuaweicloud.com替换为您开通服务区域的终端节点
$endpoint = "https://image.cn-north-4.myhuaweicloud.com";
$projectId = "";
$credentials = new BasicCredentials($ak,$sk,$projectId);
$config = HttpConfig::getDefaultConfig();
$config->setIgnoreSslVerification(true);
$client = ImageClient::newBuilder(new ImageClient)
 ->withHttpConfig($config)
 ->withEndpoint($endpoint)
 ->withCredentials($credentials)
 ->build();
$request = new RunRecaptureDetectRequest();
$body = new RecaptureDetectReq();
$body->setThreshold(0.9);
$body->setImage("输入图片,使用base64编码");
$request->setBody($body);
try {
 $response = $client->RunRecaptureDetect($request);
 echo "\n";
 echo $response;
} catch (ConnectionException $e) {
 $msg = $e->getMessage();
 echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
 $msg = $e->getMessage();
 echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
 echo "\n";
 echo $e->getHttpStatusCode(). "\n";
 echo $e->getRequestId(). "\n";
 echo $e->getErrorCode() . "\n";
echo $e->getErrorMsg() . "\n";
}
识别结果输出到控制台表示执行成功。
{"result":{"suggestion":"false","category":"recapture","score":"0.9998","detail":
[{"label":"recapture","confidence":"0.9998"}]}}
```

### 10.4 图像标签示例

本章节对图像标签AK/SK方式使用SDK进行示例说明。

- 图像标签示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用初始化 ImageTaggingReq中的Image或Url配置图像信息(image和url参数二选一),示 例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端
   节点。
- 图像标签调用示例代码如下:
   <?php</p>

namespace HuaweiCloud\SDK\Image\V2\Model; require\_once "vendor/autoload.php"; use HuaweiCloud\SDK\Core\Auth\BasicCredentials; use HuaweiCloud\SDK\Core\Http\HttpConfig; use HuaweiCloud\SDK\Core\Exceptions\ConnectionException; use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException; use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException; use HuaweiCloud\SDK\Image\V2\ImageClient; // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量中 密文存放,使用时解密,确保安全; // 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变 量CLOUD\_SDK\_AK和CLOUD\_SDK\_SK。 \$ak = getenv('CLOUD\_SDK\_AK'); \$sk = getenv('CLOUD\_SDK\_SK'); //image.cn-north-4.myhuaweicloud.com替换为您开通服务区域的终端节点 \$endpoint = "https://image.cn-north-4.myhuaweicloud.com"; \$projectId = ""; \$credentials = new BasicCredentials(\$ak,\$sk,\$projectId); \$config = HttpConfig::getDefaultConfig(); \$config->setIgnoreSslVerification(true); \$client = ImageClient::newBuilder(new ImageClient) ->withHttpConfig(\$config) ->withEndpoint(\$endpoint) ->withCredentials(\$credentials) ->build(); \$request = new RunImageTaggingRequest(); \$body = new ImageTaggingReq(); \$body->setLimit(10); \$body->setThreshold(60); \$body->setLanguage("zh"); \$body->setImage("输入图片,使用base64编码"); \$request->setBody(\$body); try { \$response = \$client->RunImageTagging(\$request); echo "\n"; echo \$response; } catch (ConnectionException \$e) { \$msg = \$e->getMessage(); echo "\n". \$msg ."\n"; } catch (RequestTimeoutException \$e) { \$msg = \$e->getMessage(); echo "\n". \$msg ."\n"; } catch (ServiceResponseException \$e) { echo "\n"; echo \$e->getHttpStatusCode(). "\n"; echo \$e->getRequestId(). "\n"; echo \$e->getErrorCode() . "\n"; echo \$e->getErrorMsg() . "\n";

2. 识别结果输出到控制台表示执行成功。相关参数说明请参考图像标签API。

RunImageTaggingResponse {"result": {"tags": [{"confidence": "98.01", "type": "树", "tag": "树", "i18n\_tag": {"zh": "树", "en": "Tree"}, "i18n\_type": {"zh": "树", "en": "Tree"}, "instances": []}]}} Process exiting with code: 0

## **11** 使用 SDK(C++)

## 11.1 C++开发环境配置

#### 准备工作

- 注册华为账号并开通华为云,并完成实名认证,账号不能处于欠费或冻结状态。
- 已具备开发环境,支持 C++ 14 及以上版本,要求安装 CMake 3.10 及以上版本。
- 登录"我的凭证 > 访问密钥"页面,获取Access Key(AK)和Secret Access Key (SK)。

#### 图 11-1 获取 AK、SK

| 我的凭证          | <b>நிலை பிடைய பிடிய பிட</b> |       |      |                      |  |
|---------------|---|-------|------|----------------------|--|
| APU先近<br>访问题明 | ① 刘果动将老规算器, 会带来数据装置风险, 且每个物何是研究所下到一次, 为了保持全全性, 建议您定规要当件装备保存的用意用.  |       |      |                      |  |
|               | ● 教授的方向本明已丢完、您可然運動的方向本明并得无限有的方向本明。  |       |      |                      |  |
|               | 新潮汕河市市  |       |      |                      |  |
|               | ianginanin ⊖  | 新EE ⊖ | 秋四 ⊖ | 698899160 O          |  |
|               | And the second second   |       | ◎風用  | helps privile by the |  |

 登录"我的凭证"页面,获取"IAM用户名""、账号名"以及待使用区域的 "项目ID"。调用服务时会用到这些信息,请提前保存。

本样例以"华北-北京四"区域为例,获取对应的项目ID(project\_id )。

#### **图 11-2** 我的凭证

| 我的死证              | API先証 ③                         |       |     |                |       |
|-------------------|---------------------------------|-------|-----|----------------|-------|
| APIDEE<br>(SIDER) | ● 有关中力公司号, MM子用户和双目的用关信息, 请告承公司 |       |     |                | ×     |
|                   | MARIA 4.2                       |       | 账号属 | Color (1994)   |       |
|                   | 1448,440 768 19 C               |       | 肥明ロ | a60 7004 🖓     |       |
|                   | 309%                            |       |     |                | <br>٩ |
|                   | No e                            | 第□ 0  |     | MERIC O        |       |
|                   | 05ee -                          | 1.000 |     | Set 17 Barrier |       |

#### 安装 SDK

- 在Linux系统上安装SDK
  - a. 获取依赖包 所需的这些第三方软件包在大部分系统的包管理工具中都有提供,例如基于 Debian/Ubuntu 的系统。

图像识别

SDK 参考

sudo apt-get install libcurl4-openssl-dev libboost-all-dev libssl-dev libcpprest-dev

#### spdlog 需要从源码进行安装。

git clone https://github.com/gabime/spdlog.git cd spdlog mkdir build cd build cmake -DCMAKE\_POSITION\_INDEPENDENT\_CODE=ON .. // 用以生成动态库 make sudo make install

#### b. 编译安装

git clone https://github.com/huaweicloud/huaweicloud-sdk-cpp-v3.git cd huaweicloud-sdk-cpp-v3 mkdir build cd build cmake .. make sudo make install

完成上述操作后,C++ SDK 安装目录为 /usr/local 。

- 在Windows系统上安装SDK
  - a. 安装 vcpkg 并使用 vcpkg 安装所需软件包 vcpkg install curl cpprestsdk boost openssl spdlog
  - b. 使用CLion进行编译
    - i. 使用CLion打开huaweicloud-sdk-cpp-v3 目录。
    - ii. 选择"File > Settings"。
    - iii. 选择"Build, Execution, Deployment > > CMake"。
    - iv. 在CMake options中加入: -DCMAKE\_TOOLCHAIN\_FILE={your vcpkg install dir}/scripts/buildsystems/vcpkg.cmake
    - v. 右键 CMakeLists.txt 选择 Load CMake Project。
    - vi. 选择Build开始编译。
  - c. 安装C++ SDK

编译完成后选择"Build > Install"。

完成上述操作后,C++ SDK 安装目录为 C:\Program File (x86)\huaweicloud-sdk-cpp-v3。

#### 开始使用 SDK

#### 1. 导入依赖模块

- #include <cstdlib>
  #include <iostream>
  #include <iostream>
  #include <string>
  #include <memory>
  #include <huaweicloud/core/exception/Exceptions.h>
  #include <huaweicloud/core/Client.h>
  #include <huaweicloud/image/v2/ImageClient.h>
  using namespace HuaweiCloud::Sdk::Image::V2;
  using namespace HuaweiCloud::Sdk::Core;
  using namespace HuaweiCloud::Sdk::Core;
  using namespace HuaweiCloud::Sdk::Core::Exception;
  using namespace std;
- 2. 配置客户端连接参数
  - · 默认配置 // 使用默认配置 HttpConfig httpConfig = HttpConfig();
  - 网络代理(可选)

// 根据需要配置网络代理 httpConfig.setProxyProtocol("http"); httpConfig.setProxyHost("proxy.huawei.com"); httpConfig.setProxyPort("8080"); httpConfig.setProxyUser("username"); httpConfig.setProxyPassword("password");

- 超时配置(可选)
   // 默认连接超时为60秒,默认读取超时为120秒。可根据需求修改该默认值 httpConfig.setConnectTimeout(60); httpConfig.setReadTimeout(120);
- SSL配置(可选)
   // 配置跳过服务端证书验证
   httpConfig.setIgnoreSslVerification(true);
- 3. 配置认证信息

配置AK、SK、projectld信息。华为云通过AK识别用户的身份,通过SK对请求数 据进行签名验证,用于确保请求的机密性、完整性和请求者身份的正确性。 string ak = getenv("HUAWEICLOUD\_SDK\_AK"); string sk = getenv("HUAWEICLOUD\_SDK\_SK"); string projectld = getenv("PROJECT\_ID"); auto auth = std::make\_unique<BasicCredentials>(); auth->withAk(ak) .withSk(sk) .withProjectld(projectld); 认证参数说明:

- ak、sk:访问密钥信息,获取方法请参见<mark>准备工作</mark>。
- projectld:华为云项目ID,获取方法请参见<mark>准备工作</mark>。



- 认证用的 ak 和sk 硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量中密文存放,使用时解密,确保安全。
- 本示例以 ak 和 sk 保存在环境变量中来实现身份验证为例,运行本示例前 请先在本地环境中设置环境变量HUAWEICLOUD\_SDK\_AK, HUAWEICLOUD\_SDK\_SK和PROJECT\_ID。

#### 图 11-3 Windows 环境新建环境变量

| 义主信(*).<br>浏览目录(D) |                    |  |
|--------------------|--------------------|--|
| ☆最值(\/):           |                    |  |
| 变量名(N):            | HUAWEICLOUD_SDK_AK |  |
|                    |                    |  |
| 新建用户变量             | ×                  |  |

4. 初始化客户端

| 指定云服务endpoint,初始化客户端  |
|---|
| string endpoint = "https://image.cn-north-4.myhuaweicloud.com";   |
| auto client = ImageClient::newBuilder()   |
| .withCredentials(std::unique_ptr <credentials>(auth.release()))<br/>.withHttpConfig(httpConfig)</credentials> |
| .withEndPoint(endpoint)   |
| .build();   |
|   |

endpoint:华为云各服务应用区域和各服务的终端节点,详情请查看<mark>地区和终端</mark> 节点 。

#### 5. 发送请求并查看响应

// 以调用接口RunImageMediaTagging为例 RunImageMediaTaggingRequest request; ImageMediaTaggingReq body; body.setLimit(10); body.setThreshold(60); body.setLanguage("zh"); body.setImage("输入图片,使用base64编码"); request.setBody(body);

6. 异常处理

#### 表 11-1 异常处理

| 一级分类  | 一级分类说明      | 二级分类                       | 二级分类说明                             |
|---|-------------|----------------------------|------------------------------------|
| ConnectionExcepti 连接类异常 HostUnreachablel<br>on xception |             | HostUnreachableE xception  | 网络不可达、被拒<br>绝。                     |
|   |             | SslHandShakeExce<br>ption  | SSL认证异常。                           |
| RequestTimeoutEx ception                                | 响应超时异常      | CallTimeoutExcept<br>ion   | 单次请求,服务器<br>处理超时未返回。               |
|   |             | RetryOutageExcep<br>tion   | 在重试策略消耗完<br>成后,仍无有效的<br>响应。        |
| ServiceResponseEx<br>ception                            | 服务器响应异<br>常 | ServerResponseEx ception   | 服务端内部错误,<br>Http响应码:<br>[500,] 。   |
|   |             | ClientRequestExce<br>ption | 请求参数不合法,<br>Http响应码: [400,<br>500) |

#### // 捕获和处理不同类型的异常

#### try {

auto reponse = client->runImageMediaTagging(request); std::cout << response->getHttpBody() << std::endl;</pre> } catch (HostUnreachableException& e) { std::cout << "host unreachable:" << e.what() << std::endl;</pre> } catch (SslHandShakeException& e) { std::cout << "ssl handshake error:" << e.what() << std::endl;</pre> } catch (RetryOutageException& e) { std::cout << "retryoutage error:" << e.what() << std::endl; } catch (CallTimeoutException& e) { std::cout << "call timeout:" << e.what() << std::endl;</pre> } catch (ServiceResponseException& e) { std::cout << "http status code:" << e.getStatusCode() << std::endl;</pre> std::cout << "error code:" << e.getErrorCode() << std::endl; std::cout << "error msg:" << e.getErrorMsg() << std::endl;</pre> std::cout << "RequestId:" << e.getRequestId() << std::endl;</pre> } catch (exception& e) { std:cout << "unknown exception:" << e.what() << std::endl;</pre> } std::cout << "-----request finished------" << std::endl;

```
🗀 说明
```

使用异步客户端,配置日志等操作请参见SDK中心、C++ SDK使用指导。

## 11.2 媒资图像标签示例

本章节对媒资图像标签AK/SK方式使用SDK进行示例说明。

- 媒资图像标签示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用 ImageMediaTaggingReq中的Url或Image方法配置图像信息(image和url参数二 选一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端
   节点。
- 1. 媒资图像标签调用示例代码如下:

```
#include <cstdlib>
#include <iostream>
#include <string>
#include <memory>
#include <huaweicloud/core/exception/Exceptions.h>
#include <huaweicloud/core/Client.h>
#include <huaweicloud/image/v2/ImageClient.h>
using namespace HuaweiCloud::Sdk::Image::V2;
using namespace HuaweiCloud::Sdk::Image::V2::Model;
using namespace HuaweiCloud::Sdk::Core;
using namespace HuaweiCloud::Sdk::Core::Exception;
using namespace std;
int main(void)
 // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量中
密文存放,使用时解密,确保安全
 // 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
量CLOUD SDK AK和CLOUD SDK SK。
  string ak = getenv("CLOUD_SDK_AK");
  string sk = getenv("CLOUD_SDK_SK");
  string projectId = "";
//image.cn-north-4.myhuaweicloud.com替换为您开通服务区域的终端节点
  string endpoint = "https://image.cn-north-4.myhuaweicloud.com";
  auto auth = std::make_unique<BasicCredentials>();
  auth->withAk(ak)
     .withSk(sk)
    .withProjectId(projectId);
  HttpConfig httpConfig = HttpConfig();
  auto client = ImageClient::newBuilder()
       .withCredentials(std::unique_ptr<Credentials>(auth.release()))
       .withHttpConfig(httpConfig)
       .withEndPoint(endpoint)
       .build();
  RunImageMediaTaggingRequest request;
  ImageMediaTaggingReq body;
  body.setLimit(10);
  body.setThreshold(60);
  body.setLanguage("zh");
  body.setImage("输入图片,使用base64编码");
  request.setBody(body);
  std::cout << "-----begin execute request------" << std::endl;
  try {
    auto reponse = client->runImageMediaTagging(request);
    std::cout << reponse->getHttpBody() << std::endl;</pre>
  } catch (HostUnreachableException& e) {
    std::cout << "host unreachable:" << e.what() << std::endl;</pre>
  } catch (SslHandShakeException& e) {
    std::cout << "ssl handshake error:" << e.what() << std::endl;</pre>
  } catch (RetryOutageException& e) {
```

std::cout << "retryoutage error:" << e.what() << std::endl;</pre>

```
} catch (CallTimeoutException& e) {
   std::cout << "call timeout:" << e.what() << std::endl;
} catch (ServiceResponseException& e) {
   std::cout << "http status code:" << e.getStatusCode() << std::endl;
   std::cout << "error code:" << e.getErrorCode() << std::endl;
   std::cout << "error msg:" << e.getErrorMsg() << std::endl;
   std::cout << "RequestId:" << e.getRequestId() << std::endl;
} catch (exception& e) {
   std::cout << "unknown exception:" << e.what() << std::endl;
}
std::cout << "-----request finished------" << std::endl;
return 0:</pre>
```

reu

执行示例代码文件,控制台输出媒资图像标签结果即表示程序执行成功。相关参数说明请参考媒资图像标签API。

RunImageMediaTaggingResponse {"result": {"tags": [{"confidence": "83.15", "type": "社会", "tag": "火 灾", "i18n\_tag": {"zh": "火灾", "en": "Fire"}, "i18n\_type": {"zh": "社会", "en": "Society"}, "instances": []}, {"confidence": "70.21", "type": "其他", "tag": "火", "i18n\_tag": {"zh": "火", "en": "Fire"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "53.63", "type": "学科科学", "tag": "火球", "i18n\_tag": {"zh": "火球", "en": "Fireball"}, "i18n\_type": {"zh": "学科科学", "en": "Subject science"}, "instances": []}, {"confidence": "53.45", "type": "其他", "tag": "营火", "i18n\_tag": {"zh": "营火", "en": "Campfire"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "50.73", "type": "人工场景", "tag": "林业局", "i18n\_tag": {"zh": "林业局", "en": "Forestry bureau"}, "i18n\_type": {"zh": "人工场景", "en": "Manual scenario"}, "instances": []}, {"confidence": "50.34", "type": "其他", "tag": "烧 毀的", "i18n\_tag": {"zh": "烧毁的", "en": "Burned"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "instances": []}, {"confidence": "46.63", "type": "场景", "tag": "火焰", "i18n\_tag": {"zh": "火焰", "en": "Flame"}, "i18n\_tag": {"zh": "燃烧", "en": "Suened"}, "instances": []}, {"confidence": "45.27", "type": "其 他", "tag": "燃烧", "i18n\_tag": {"zh": "燃烧", "en": "Burning"}, "i18n\_type": {"zh": "其他", "en": "Other"}, "i18n\_tag": {"zh": "K%," "en": "Burning"}, "i18n\_type": {"zh": "j16m," unitariag": "Cife", "instances": []}, {"confidence": "41.32", "type": "职业/角色", "en": "Career/Role"}, "instances": []}, {"confidence": "39.97", "type": "动作", "tag": "烧香", "i18n\_tag": {"zh": "K春", "en": "Incense burning"}, "i18n\_type": {"zh": "动作", "en": "Action"}, "i18n\_tag": {"zh": "K春", "en": "Incense burning"}, "i18n\_type": {"zh": "动作", "en": "Action"}, "i18n\_tag": {"zh": "K<sup>\*</sup></sup>, "en": "Incense

## 11.3 翻拍识别示例

本章节对翻拍识别AK/SK方式使用SDK进行示例说明。

- 翻拍识别示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用初始化 RunRecaptureDetectReq中的Image或Url配置图像信息(image和url参数二选 一),示例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端
   节点。
- 1. 翻拍识别调用示例代码如下:

```
#include <cstdlib>
#include <iostream>
#include <strina>
#include <memory>
#include <huaweicloud/core/exception/Exceptions.h>
#include <huaweicloud/core/Client.h>
#include <huaweicloud/image/v2/ImageClient.h>
using namespace HuaweiCloud::Sdk::Image::V2;
using namespace HuaweiCloud::Sdk::Image::V2::Model;
using namespace HuaweiCloud::Sdk::Core;
using namespace HuaweiCloud::Sdk::Core::Exception;
using namespace std;
int main(void)
ł
 // 认证用的ak和sk硬编码到代码中或者明文存储都有很大的安全风险,建议在配置文件或者环境变量中
密文存放,使用时解密,确保安全;
 // 本示例以ak和sk保存在环境变量中来实现身份验证为例,运行本示例前请先在本地环境中设置环境变
量CLOUD_SDK_AK和CLOUD_SDK_SK。
  string ak = getenv("CLOUD_SDK_AK");
  string sk = getenv("CLOUD_SDK_SK");
  string projectId = "";
```

```
//image.cn-north-4.myhuaweicloud.com替换为您开通服务区域的终端节点
  string endpoint = "https://image.cn-north-4.myhuaweicloud.com";
  auto auth = std::make_unique<BasicCredentials>();
  auth->withAk(ak)
     .withSk(sk)
     .withProjectId(projectId);
  HttpConfig httpConfig = HttpConfig();
  auto client = ImageClient::newBuilder()
        .withCredentials(std::unique_ptr<Credentials>(auth.release()))
        .withHttpConfig(httpConfig)
        .withEndPoint(endpoint)
        .build();
  RunRecaptureDetectRequest request;
  RecaptureDetectReq body;
  body.setThreshold(0.95);
  body.setImage("输入图片,使用base64编码");
  request.setBody(body);
  std::cout << "-----begin execute request------" << std::endl;
  try {
     auto reponse = client->runRecaptureDetect(request);
     std::cout << reponse->getHttpBody() << std::endl;</pre>
  } catch (HostUnreachableException& e) {
     std::cout << "host unreachable:" << e.what() << std::endl;</pre>
  } catch (SslHandShakeException& e) {
     std::cout << "ssl handshake error:" << e.what() << std::endl;</pre>
  } catch (RetryOutageException& e) {
     std::cout << "retryoutage error:" << e.what() << std::endl;
  } catch (CallTimeoutException& e) {
     std::cout << "call timeout:" << e.what() << std::endl;</pre>
  } catch (ServiceResponseException& e) {
     std::cout << "http status code:" << e.getStatusCode() << std::endl;</pre>
     std::cout << "error code:" << e.getErrorCode() << std::endl;</pre>
     std::cout << "error msg:" << e.getErrorMsg() << std::endl;</pre>
     std::cout << "RequestId:" << e.getRequestId() << std::endl;</pre>
  } catch (exception& e) {
     std:cout << "unknown exception:" << e.what() << std::endl;</pre>
  std::cout << "-----request finished------" << std::endl;
  return 0;
}
识别结果输出到控制台表示执行成功。
```

```
{"result":{"suggestion":"false","category":"recapture","score":"0.9998","detail":
[{"label":"recapture","confidence":"0.9998"}]}}
```

## 11.4 图像标签示例

2.

本章节对图像标签AK/SK方式使用SDK进行示例说明。

- 图像标签示例代码只需将AK/SK信息替换为实际AK/SK,代码中可以使用初始化 ImageTaggingReq中的Image或Url配置图像信息(image和url参数二选一),示 例中以url为例,配置完成后运行即可。
- endpoint是华为云各服务应用区域和各服务的终端节点,详情请查看地区和终端
   节点。
- 1. 图像标签调用示例代码如下:

#include <cstdlib>
#include <iostream>
#include <iostream>
#include <string>
#include <memory>
#include <huaweicloud/core/exception/Exceptions.h>
#include <huaweicloud/core/Client.h>
#include <huaweicloud/image/v2/ImageClient.h>
using namespace HuaweiCloud::Sdk::Image::V2;
using namespace HuaweiCloud::Sdk::Image::V2::Model;
using namespace HuaweiCloud::Sdk::Core;

| using namespace HuaweiCloud::Sdk::Core::Exception;   |
|--|
| using namespace std;   |
| int main(void)   |
| {  |
| 7/ 以证用的品种品质硬编码到10词干线目的文件储邮件低入的文主风险,建设在能量文件或目外境文量干密文存放。使用时解密。确保安全:  |
| 二人了场,这个时间,你这个之中。   |
| THE ACTION AND AND A AND |
| string ak = getenv("CLOUD_SDK_AK");  |
| string sk = getenv("CLOUD_SDK_SK");  |
| string projectId = "";   |
| //image.cn-north-4.myhuaweicloud.com替换为您开通服务区域的终端节点  |
| string endpoint = "https://image.cn-north-4.myhuaweicloud.com";  |
| auto auth = std::make_unique <basiccredentials>();</basiccredentials>  |
| auth->withAk(ak)   |
| withSk(sk)   |
| .withProjectId(projectId);   |
| HttpConfig httpConfig = HttpConfig();  |
| auto cuent = ImageClient::newBuilder()   |
| with Uth Condition (http://www.secondition.com/secondition   |
| withEndPoint(endpoint)   |
|  |
| .outo(),<br>RunhaaeTaagingReguest reguest  |
| ImageTaggingReg body:  |
| body.setLimit(10):   |
| body.setThreshold(60):   |
| body.setLanguage("zh");  |
| body.setImage("输入图片,使用base64编码");  |
| request.setBody(body);   |
| std::cout << "begin execute request" << std::endl;   |
| try {  |
| auto reponse = client->runImageTagging(request);   |
| std::cout << reponse->getHttpBody() << std::endl;  |
| } catch (HostUnreachableException& e) {  |
| std::cout << "host unreachable:" << e.what() << std::endl;   |
| <pre>} catch (SSIHandShakeException&amp; e) {     stduesut cc "sel bandshake arror" cc a what() cc at dues due </pre>  |
| stuttout << "sst nandsnake error:" << e.wnat() << std::endl;   |
| <pre>std::cout &lt;&lt; "retryoutage error" &lt;&lt; e what() &lt;&lt; std::endl:</pre>  |
| Succours <= religious end () << successful,  |
| std:rout << "call timeout:" << e what() << std:rendl:  |
| } catch (ServiceResponseException& e) {  |
| std::cout << "http://status.code:" << e.getStatusCode() << std::endl:  |
| std::cout << "error code:" << e.getErrorCode() << std::endl;   |
| std::cout << "error msg:" << e.getErrorMsg() << std::endl;   |
| std::cout << "RequestId:" << e.getRequestId() << std::endl;  |
| } catch (exception& e) {   |
| <pre>std:cout &lt;&lt; "unknown exception:" &lt;&lt; e.what() &lt;&lt; std::endl;</pre>  |
| }  |
| std::cout << "request finished" << std::endl;  |
| return 0;  |
| }  |
| 识别结果输出到控制台表示执行成功。相关参数说明请参考 <mark>图像标签AP</mark> L。  |
|  |

2. 识别结果输出到控制台表示执行成功。相关参数说明请参考<mark>图像标签API</mark>。 RunImageTaggingResponse {"result": {"tags": [{"confidence": "98.01", "type": "树", "tag": "树", "i18n\_tag": {"zh": "树", "en": "Tree"}, "i18n\_type": {"zh": "树", "en": "Tree"}, "instances": []}]}} Process exiting with code: 0