

图像识别

## SDK 参考

文档版本 01  
发布日期 2021-09-06



版权所有 © 华为技术有限公司 2021。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



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

## 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： <https://www.huawei.com>

客户服务邮箱： [support@huawei.com](mailto:support@huawei.com)

客户服务电话： 4008302118

---

# 目录

---

<b>1 图像识别 SDK 简介</b> .....	<b>1</b>
<b>2 申请服务</b> .....	<b>2</b>
<b>3 获取认证信息</b> .....	<b>3</b>
<b>4 准备环境</b> .....	<b>4</b>
<b>5 SDK 列表及指导文档（通用版）</b> .....	<b>5</b>
<b>6 使用 SDK（Java）</b> .....	<b>6</b>
6.1 Java 开发环境配置.....	6
6.2 SDK 获取和安装.....	6
6.3 图像标签示例.....	7
6.4 名人识别示例.....	8
<b>7 使用 SDK（Python）</b> .....	<b>10</b>
7.1 Python 开发环境配置.....	10
7.2 Python 语言 SDK 获取和安装.....	11
7.3 图像标签示例.....	11
7.4 名人识别示例.....	12
<b>8 使用 SDK（.NET）</b> .....	<b>14</b>
8.1 .NET 开发环境配置.....	14
8.2 .Net 语言 SDK 获取和安装.....	14
8.3 图像标签示例.....	15
8.4 名人识别示例.....	17
<b>9 使用 SDK（GO）</b> .....	<b>19</b>
9.1 GO 开发环境配置.....	19
9.2 GO 语言 SDK 获取和安装.....	19
9.3 图像标签示例.....	19
9.4 名人识别示例.....	20

# 1 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，包括图像标签，名人识别等。

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	API
图像标签	POST /v2/{project_id}/image/tagging
名人识别	POST /v2/{project_id}/image/celebrity-recognition

# 2 申请服务

---

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

# 4 准备环境

在使用图像识别SDK时，各语言需要准备的环境 参照[表4-1](#)。

**表 4-1** 开发环境

开发语言	准备项	说明
JAVA	安装JDK	JAVA环境开发配置，支持 Java JDK 1.8 及以上版本，推荐通过Maven 安装依赖的方式使用JAVA版本SDK。
PYTHON	安装python	python版本sdk支持python3.3及以上版本。
.NET	安装.NET	.NET Standard 2.0 及其以上版本或C# 4.0 及其以上版本。
GO	安装go	支持 go 1.14 及以上版本。

# 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。

**表 5-1** SDK 列表

编程语言	github地址	指导文档	视频指导
JAVA	<a href="#">huaweicloud-sdk-java-v3</a>	<a href="#">Java SDK使用指导</a>	<a href="#">Java SDK视频指导</a>
PYTHON	<a href="#">huaweicloud-sdk-python-v3</a>	<a href="#">Python SDK使用指导</a>	<a href="#">Python SDK视频指导</a>
.NET	<a href="#">huaweicloud-sdk-net-v3</a>	<a href="#">.Net SDK使用指导</a>	-
GO	<a href="#">huaweicloud-sdk-go-v3</a>	<a href="#">Go SDK使用指导</a>	-



# 6 使用 SDK (Java)

## 6.1 Java 开发环境配置

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。

- 步骤1** 下载[JDK文件](#)。
- 步骤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。
- 步骤3** 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。
1. 新建系统变量JAVA\_HOME，变量值为实际JDK安装位置。
  2. 在Path中添加%JAVA\_HOME%\bin;%JAVA\_HOME%\jre\bin（注意用英文分号分隔）。
  3. 新建系统变量CLASSPATH，变量值为%JAVA\_HOME%\lib\dt.jar;%JAVA\_HOME%\lib\tools.jar。
- 步骤4** 打开命令行窗口，输入“java -version”，显示如[图6-1](#)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)
```

----结束

## 6.2 SDK 获取和安装

JAVA版本图像识别的SDK推荐推荐您通过 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。

```
D:\>mvn -v
D:\
Apache 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
OS 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. 点击File - New - project...
3. 在New Project弹窗点击-Maven-点击Next。
4. 输入GroupId和ArtifactId，点击Next。
5. 输入Project name 和 Project location,点击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3.0.55版本的图像识别SDK为例：

```
<dependency>
  <groupId>com.huaweicloud.sdk</groupId>
  <artifactId>huaweicloud-sdk-image</artifactId>
  <version>3.0.55</version>
</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

## 6.3 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类的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。

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) {
    String ak = "<YOUR AK>";
    String sk = "<YOUR SK>";
    ICredential auth = new BasicCredentials()
        .withAk(ak)
        .withSk(sk);
    ImageClient client = ImageClient.newBuilder()
        .withCredential(auth)
        .withRegion(ImageRegion.valueOf("cn-north-4"))
        .build();
    RunImageTaggingRequest request = new RunImageTaggingRequest();
    ImageTaggingReq body = new ImageTaggingReq();
    body.withLimit(50);
    body.withThreshold(95f);
    body.withLanguage("zh");
    body.withUrl("https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/tagging-normal.jpg");
    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());
    }
}
}
}

```

2. 执行示例代码，控制台返回200即表示程序执行成功。[图像标签识别结果](#)输出到控制台。

```

SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
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
            }
            instances: []
        }
    }
}
Process finished with exit code 0

```

## 6.4 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类的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。

1. 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.RunCelebrityRecognitionResponse;

public class RunCelebrityRecognitionSolution {

    public static void main(String[] args) {
        String ak = "<YOUR AK>";
        String sk = "<YOUR SK>";

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ImageClient client = ImageClient.newBuilder()
            .withCredential(auth)
            .withRegion(ImageRegion.valueOf("cn-north-1"))
            .build();

        RunCelebrityRecognitionRequest request = new RunCelebrityRecognitionRequest();
        CelebrityRecognitionReq body = new CelebrityRecognitionReq();
        body.withThreshold(0f);
        body.withUrl("https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/celebrity-
recognition.jpg");
        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());
        }
    }
}
```

2. 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。**名人识别结果**输出到控制台。

```
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
200
{"statusCode":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
```

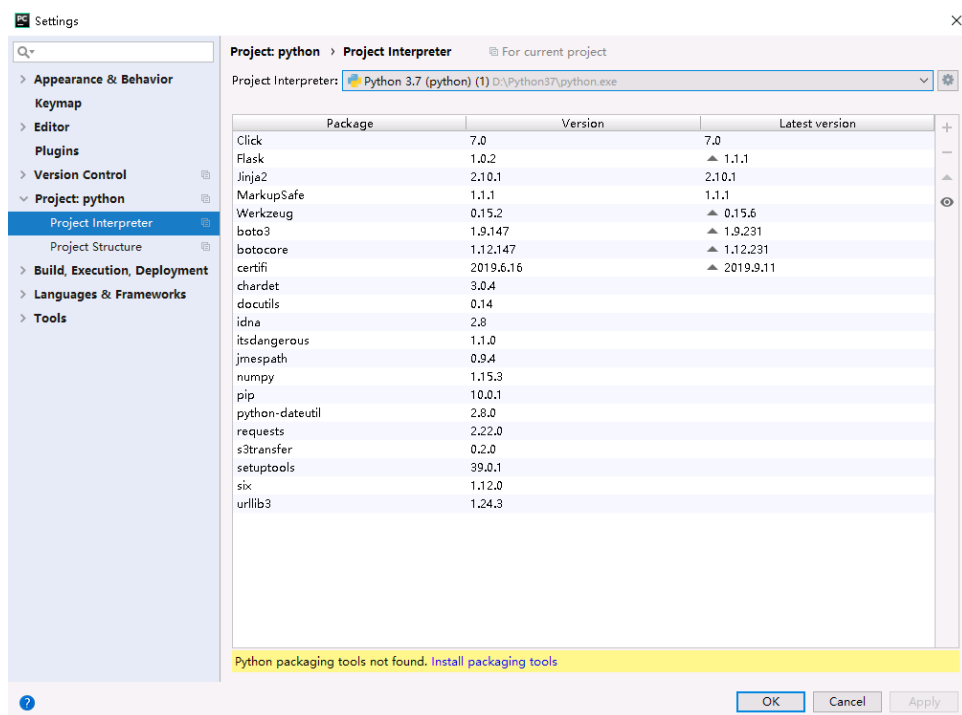
# 7 使用 SDK ( Python )

## 7.1 Python 开发环境配置

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。

图 7-1 PyCharm 配置 python 环境



## 7.2 Python 语言 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：

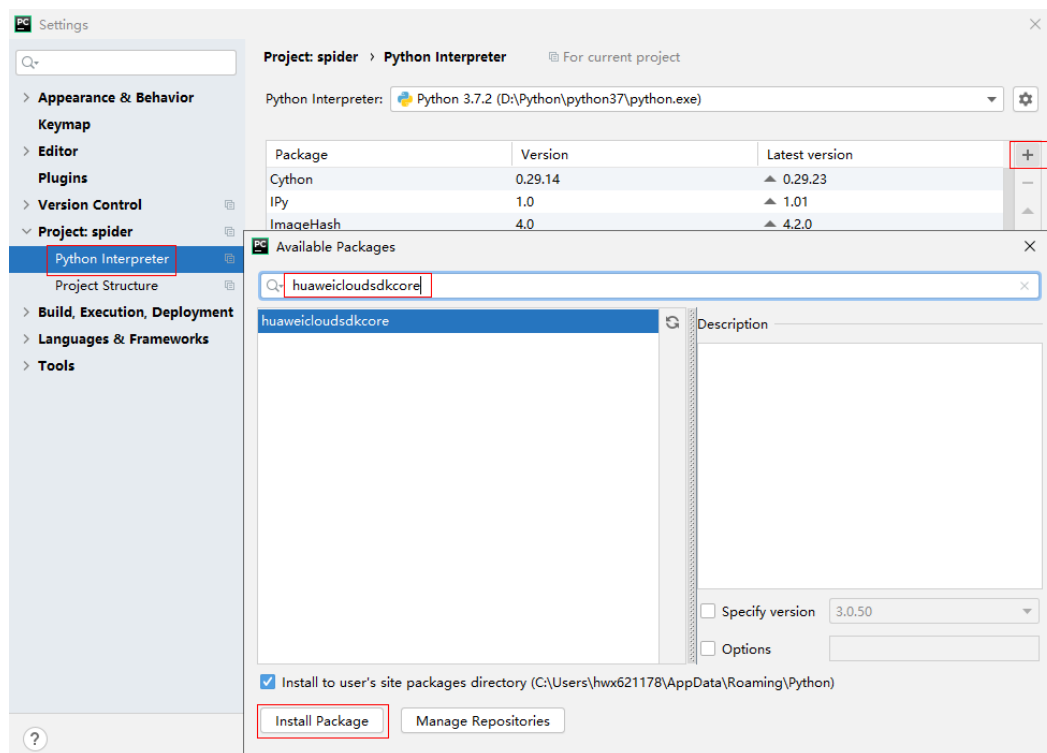
```
pip 安装：
# 安装核心库
pip install huaweicloudsdkcore

# 安装image服务库
pip install huaweicloudsdkimage
```

使用pycharm安装，步骤如下：

1. 打开pycharm,点击File -> Settings...
2. 点击Python Interpreter -> 点击+
3. 点击+, 分别搜索huaweicloudsdkcore及huaweicloudsdkimage，搜索到包内容点击左下角Install Package完成安装

图 7-2 pycharm 安装图像识别 python 版本 sdk 包



## 7.3 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。

1. 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__":
    ak = "<YOUR AK>"
    sk = "<YOUR SK>"

    credentials = BasicCredentials(ak, sk)

    client = ImageClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ImageRegion.value_of("cn-north-1")) \
        .build()

    try:
        request = RunImageTaggingRequest()
        request.body = ImageTaggingReq(
            limit=50,
            threshold=95,
            language="zh",
            url="https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/tagging-normal.jpg"
        )
        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)

```

2. 执行示例代码，控制台返回200即表示程序执行成功。[图像标签识别结果](#)输出到控制台。

```

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

```

## 7.4 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。

1. 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__":
    ak = "<YOUR AK>"
    sk = "<YOUR SK>"

    credentials = BasicCredentials(ak, sk)

    client = ImageClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ImageRegion.value_of("cn-north-1")) \

```

```
.build()

try:
    request = RunCelebrityRecognitionRequest()
    request.body = CelebrityRecognitionReq(
        threshold=0,
        url="https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/celebrity-recognition.jpg"
    )
    response = client.run_celebrity_recognition(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)
```

2. 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。[名人识别结果](#)输出到控制台。

```
200
{'result': [{'confidence': 0.9990678852332598,
             'face_detail': {'h': 510, 'w': 357, 'x': 260, 'y': 201},
             'label': '迈克尔·杰克逊'}]}
```



# 8 使用 SDK (.NET)

## 8.1 .NET 开发环境配置

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，点击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



## 8.2 .Net 语言 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：

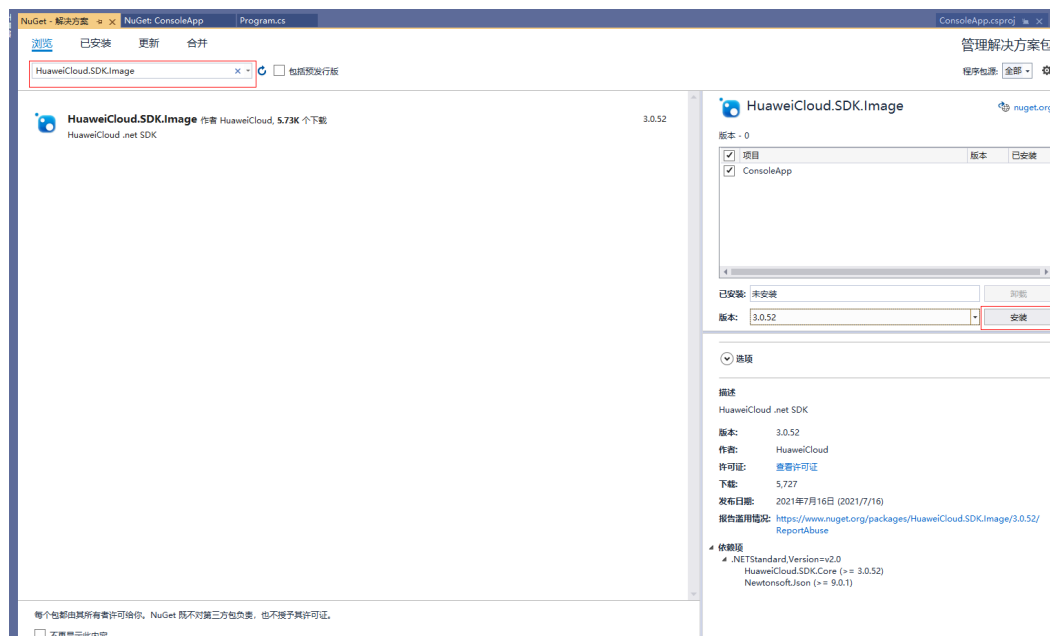
```
使用 .NET CLI 工具
dotnet add package HuaweiCloud.SDK.Core
dotnet add package HuaweiCloud.SDK.Image

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：点击工具->选择NuGet包管理器->点击管理解决方案的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



## 8.3 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。

### 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)
        {
            const string ak = "<YOUR AK>";
            const string sk = "<YOUR SK>";

            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"))
    .WithHttpConfig(config)
    .Build();

var req = new RunImageTaggingRequest
{
};
req.Body = new ImageTaggingReq()
{
    Limit = 50,
    Threshold = 95,
    Language = "zh",
    Url = "https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/tagging-normal.jpg"
};

try
{
    var resp = client.RunImageTagging(req);
    var respStatusCode = resp.HttpStatusCode;
    Console.WriteLine(respStatusCode);
    Console.WriteLine(JsonConvert.SerializeObject(resp.HttpBody));
}
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)
{
    Console.WriteLine(connectionException.ErrorMessage);
}
}
}
}

```

2. 执行示例代码，控制台返回200即表示程序执行成功。图像标签识别结果输出到控制台。

```

200
{
  "result": {
    "tags": [
      {
        "confidence": "98.01",
        "type": "树",
        "tag": "树",
        "i18n_tag": {
          "zh": "树",
          "en": "Tree"
        },
        "i18n_type": {
          "zh": "树",
          "en": "Tree"
        },
        "instances": [ ]
      }
    ]
  }
}

```

## 8.4 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类的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。

### 1. 名人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;
using Newtonsoft.Json;

namespace RunCelebrityRecognitionSolution
{
    class Program
    {
        static void Main(string[] args)
        {
            const string ak = "<YOUR AK>";
            const string sk = "<YOUR SK>";

            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"))
                .WithHttpConfig(config)
                .Build();

            var req = new RunCelebrityRecognitionRequest
            {
            };
            req.Body = new CelebrityRecognitionReq()
            {
                Threshold = 0.48,
                Url = "https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/celebrity-
recognition.jpg"
            };

            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.ErrorMessage);
            }
            catch (ConnectionException connectionException)
            {
            }
        }
    }
}
```

```

        Console.WriteLine(connectionException.ErrorMessage);
    }
}
}

```

2. 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。**名人识别结果**输出到控制台。

```

200
{
  'result': [{
    'confidence': 0.9990678852332598,
    'face_detail': {
      'h': 510,
      'w': 357,
      'x': 260,
      'y': 201
    },
    'label': '迈克尔·杰克逊'
  }]
}

```

# 9 使用 SDK (GO)

## 9.1 GO 开发环境配置

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。

## 9.2 GO 语言 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
```

## 9.3 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。

1. 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() {
    ak := "<YOUR AK>"
    sk := "<YOUR SK>"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := image.NewImageClient(
        image.ImageClientBuilder().
            WithRegion(region.ValueOf("cn-north-1")).
            WithCredential(auth).
            Build())

    request := &model.RunImageTaggingRequest{}
    limitImageTaggingReq:= int32(50)
    thresholdImageTaggingReq:= float32(95)
    languageImageTaggingReq:= "zh"
    urlImageTaggingReq:= "https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/tagging-normal.jpg"
    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)
    }
}

```

2. 识别结果输出到控制台表示执行成功。

```

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

```

## 9.4 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。

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() {
    ak := "<YOUR AK>"
    sk := "<YOUR SK>"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := image.NewImageClient(
        image.ImageClientBuilder().
            WithRegion(region.ValueOf("cn-north-1")).
            WithCredential(auth).
            Build())

    request := &model.RunCelebrityRecognitionRequest{}
    thresholdCelebrityRecognitionReq:= float32(0.48)
    urlCelebrityRecognitionReq:= "https://sdk-obs-source-save.obs.cn-north-4.myhuaweicloud.com/
celebrity-recognition.jpg"
    request.Body = &model.CelebrityRecognitionReq{
        Threshold: &thresholdCelebrityRecognitionReq,
        Url: &urlCelebrityRecognitionReq,
    }
    response, err := client.RunCelebrityRecognition(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

2. 识别结果输出到控制台表示执行成功。

```

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

```