Face Recognition

SDK Reference

Issue   01
Date   2024-03-05

Huawei Technologies Co., Ltd.

Address:	Huawei Industrial Base Bantian, Longgang Shenzhen 518129 People's Republic of China
Website:	https://www.huawei.com
Email:	support@huawei.com

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process.* For details about this process, visit the following web page:
https://www.huawei.com/en/psirt/vul-response-process
For vulnerability information, enterprise customers can visit the following web page:
https://securitybulletin.huawei.com/enterprise/en/security-advisory

Contents

1 Overview

FRS SDKs use FRS RESTful APIs to simplify development. Currently, the FRS has a Java SDK.

Mappings Between Services and APIs

Table 1-1 lists the mappings between the FRS sub-services and APIs.

Table 1-1 Mappings between services and APIs

Function		API
Face Detection		POST /v2/{project_id}/face-detect
Face Verification		POST /v2/{project_id}/face-compare
Face Retrieval		POST /v2/{project_id}/face-sets/{face_set_name}/search
Face LiveDetect	Action live detecting	POST /v1/{project_id}/live-detect
Facial Image Library Management	Creating a facial image library	POST /v2/{project_id}/face-sets
	Querying all facial image libraries	GET /v2/{project_id}/face-sets
	Querying a facial image library	GET /v2/{project_id}/face-sets/{face_set_name}
	Deleting a facial image library	DELETE /v2/{project_id}/face-sets/{face_set_name}
Facial Resource Management	Adding faces	POST /v2/{project_id}/face-sets/{face_set_name}/faces

Function		API
	Querying a face	GET /v2/{project_id}/face-sets/{face_set_name}/faces?offset=xxx&limit=xxx
	Updating a face	PUT /v2/{project_id}/face-sets/{face_set_name}/faces
	Deleting a face	DELETE /v2/{project_id}/face-sets/{face_set_name}/faces?field_name=field_value
	Deleting faces in batches	DELETE /v2/{project_id}/face-sets/{face_set_name}/faces/batch

2 Java SDK

2.1 Quick Start

Environment Preparation

1. Enable the Face Recognition service: Log in to the **Face Recognition management console**, select a sub-service, and click **Enable Service** on the right.

   You only need to enable the service once.

2. Obtain the endpoints of Face Recognition. For details, see **Regions and Endpoints**.

3. For details about Face Recognition APIs, see the **Face Recognition API Reference**.

4. Obtain the AK/SK of your HUAWEI CLOUD account.

   If no AK/SK has been generated, log in to , enter the **My Credentials** page, and choose **Access Keys** > **Create Access Key** to obtain the access keys.

5. Install the JDK. The Java SDK can run on JDK 1.8 or later.

   📖 **NOTE**

   - SDKs provided by Face Recognition do not support access retries. If the access is abnormal, the SDK directly returns the request results. Therefore, the client needs to establish a retry mechanism.

   - When an Android system uses this Java SDK, the main thread is not supported. You are advised to use the sub-thread.

   - When using the demo in the Java SDK, you are advised to use JDK 1.9+ because **readAllBytes()** is a method in JDK 1.9+.

   - For details about how to use the demo in the SDK, see demoV2().

SDK Acquisition and Installation

Take Eclipse as an example. After creating a Java project, perform the following steps to import the JAR file to the new project:

1. Download the **SDK JAR file**.

2. Copy the downloaded JAR file to the project folder.

3. Open the project in Eclipse, right-click the project, and choose **Properties**.

4. In the displayed dialog box, click **Java Build Path**. On the **Libraries** tab, click **Add JARs** to add the downloaded JAR file.

5. Click **OK**.

Services involved in this document use the same JAR file.

2.2 SDK API Example

SDK API Example

1. Construct a service client.
```
String ak = System.getenv("HUAWEICLOUD_SDK_AK");
String sk = System.getenv("HUAWEICLOUD_SDK_SK");
String projectId = "projectId";
//Proxy information. If you need to use a proxy to access the service, add the proxy information here.
ProxyHostInfo proxyHostInfo = new ProxyHostInfo("proxy",
8080,
"UserName",
"Password"
);
//Service info
String url = "https://face.ap-southeast-1.myhuaweicloud.com";
String region = "ap-southeast-1";
AuthInfo authInfo = new AuthInfo(url, region, ak, sk);
FrsClient frsClient = new FrsClient(authInfo, projectId);
//FrsClient frsClient = new FrsClient(authInfo, projectId, proxyHostInfo);//If you need to use a proxy to access the service, add the proxy information to the third parameter.
```

2. Create a face set.
```
CreateExternalFields createExternalFields = new CreateExternalFields();
createExternalFields.addField("testInteger", FieldType.INTEGER);
CreateFaceSetResult createFaceSetResult =
frsClient.getV2().getFaceSetService().createFaceSet(faceSetName, faceSetCapacity,
createExternalFields);
```

3. Add faces.
```
String imageId = "image1";
File image = new File("data/face.jpg");
byte[] fileData = FileUtils.readFileToByteArray(image);
String imageBase64 = Base64.encodeBase64String(fileData);
AddExternalFields addExternalFields = new AddExternalFields();
addExternalFields.addField("testInteger", 123);
AddFaceResult addFaceResult = frsClient.getV2().getFaceService().addFaceByBase64(faceSetName,
imageId, imageBase64, addExternalFields);
```

4. Invoke the Face Detection service.
```
String obsUrl = "/bucket/face.jpg";
String attr = "1,2";
DetectFaceResult detectFaceResult = frsClient.getV2().getDetectService().detectFaceByObsUrl(obsUrl,
attr);
```

📖 **NOTE**

In the preceding example, **ak** and **sk** indicate the access keys. To obtain the AK/SK, log in to the **My Credentials** page, choose **Access Keys**, and click **Add Access Key**.

projectId: Project ID. For details about how to obtain a project ID, see **Obtaining a Project ID**.

2.3 Mappings Between Java SDK & APIs

API	Method	API
CompareService	compareFaceByBase64	POST /v2/{project_id}/face-compare
	compareFaceByFile	
	compareFaceByObsUrl	
DetectService	detectFaceByBase64	POST /v2/{project_id}/face-detect
	detectFaceByFile	
	detectFaceByObsUrl	
FaceService	addFaceByBase64	POST /v2/{project_id}/face-sets/{face_set_name}/faces
	addFaceByFile	
	addFaceByObsUrl	
	getFaces	GET /v2/{project_id}/face-sets/{face_set_name}/faces?offset=xxx&limit=xxx
	getFace	GET /v2/{project_id}/face-sets/{face_set_name}/faces?face_id={face_id}
	deleteFaceByFaceId	DELETE /v2/{project_id}/face-sets/{face_set_name}/faces ?external_image_id=imageID
	deleteFaceByExternalImageId	DELETE /v2/{project_id}/face-sets/{face_set_name}/faces ?face_id=faceID
	deleteFaceByFieldId	DELETE /v2/{project_id}/face-sets/{face_set_name}/faces ?id=home
FaceSetService	createFaceSet	POST /v2/{project_id}/face-sets
	getAllFaceSets	GET /v2/{project_id}/face-sets
	getFaceSet	GET /v2/{project_id}/face-sets/{face_set_name}
	deleteFaceSet	DELETE /v2/{project_id}/face-sets/{face_set_name}
LiveDetectService	liveDetectByBase64	POST /v2/{project_id}/live-detect
	liveDetectByFile	
	liveDetectByObsUrl	
SearchService	searchFaceByBase64	POST /v2/{project_id}/face-sets/{face_set_name}/search
	searchFaceByFile	
	searchFaceByObsUrl	
	searchFaceByFaceId	