ImageSearch

API Reference

Issue 01

Date 2023-12-11

HUAWEI TECHNOLOGIES CO., LTD.

Huawei Technologies Co., Ltd.

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to "Vul. Response Process". For details about the policy, see the following website:https://www.huawei.com/en/psirt/vul-response-process For enterprise customers who need to obtain vulnerability information, visit:https://securitybulletin.huawei.com/enterprise/en/security-advisory

Contents

1 Before You Start

1.1 Overview

ImageSearch leverages cutting-edge deep learning and image recognition technologies to deliver service- and industry-specific feature vectorization and search capabilities to help your search of exact/fuzzy match images from the specified gallery.

ImageSearch is offered through open application programming interfaces (APIs). You can obtain an image search result by accessing and calling an API in real time.

You can perform related operations based on the API description, syntax, parameter description, and examples provided in this document. For example, you can call the API for creating instances, search for images, or deleting images as instructed. For details about all supported operations, see **API Overview**.

Before calling an ImageSearch API, ensure that you are familiar with ImageSearch concepts. For details, see the *ImageSearch Overview*.

1.2 API Calling

ImageSearch provides Representational State Transfer (REST) APIs, allowing you to use HTTPS requests to call them. For details, see **Constructing a Request**.

.

1.3 Endpoints

An endpoint is the request address for calling an API. Endpoints vary depending on services and regions. For the endpoints of all services, see **Regions and Endpoints**.

Table 1-1 lists the endpoints of ImageSearch. Select a desired one based on service requirements.

Table 1-1 ImageSearch endpoints

Name of Covered Region	Region	Endpoint
CN-Hong Kong	ap-southeast-1	imagesearch.ap-southeast-1.myhuaweicloud.com

1.4 Limitations and Constraints

For details, see the API description and **Constraints** in the *ImageSearch Service Overview*.

1.5 Basic Concepts

- Account

  An account is generated after you register with HUAWEI CLOUD. An account has full access permissions for all the resources and cloud services under the account. It can reset user passwords and grant users permissions. The account is a payment entity, which should not be used directly to perform routine management. To ensure account security, create IAM users and grant them permissions for routine management.

- IAM user

  IAM users are created by an account to use cloud services. Each IAM user has their own password and access key to access HUAWEI CLOUD using the console or APIs.

  An IAM user can view the account ID and user ID on the **My Credentials** page of the management console. API authentication requires information such as the account name, username, and password.

- Region

  A region is a physical location where cloud resources are deployed. Availability zones (AZs) in the same region can communicate with each other over an intranet but AZs in different regions cannot communicate with each other. Deploying cloud resources in different regions can better suit certain user requirements or comply with local laws or regulations.

- AZ

  An availability zone (AZ) comprises of one or more physical data centers equipped with independent ventilation, fire, water, and electricity facilities. Computing, network, storage, and other resources in an AZ are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to allow users to build cross-AZ high-availability systems.

- Project

  A project corresponds to a HUAWEI CLOUD region. Default projects are defined to group and physically isolate resources (including computing, storage, and network resources) across regions. Users can be granted

permissions in a default project to access all resources in the region associated with the project. If you need more refined access control, create sub-projects under a default project and purchase resources in sub-projects. Then you can assign users the permissions required to access only the resources in the specific sub-projects.

Figure 1-1 Project isolation model
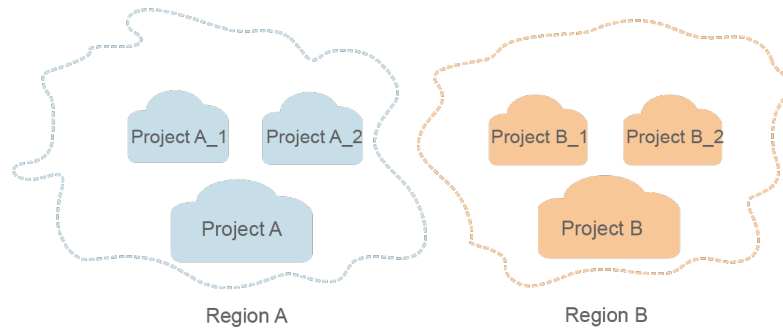
2 API Overview

Image Search provides Huawei-developed APIs. By using the Huawei-developed APIs of Image Search, you can experience all services in the following table.

The API chapter uses token-based request headers as examples. You can use either token-based or AK/SK-based request headers as required in actual use.

📖 NOTE

- The token value in the sample request is the one obtained during service application.
- The values of **endpoint** and **projectId** are the ones obtained in the region where the service is located.

API	Description
Creating an Instance	You can use the API to create an instance. An image library is created in the instance to store image features.
Adding an Image	You can use the API to import an image to the image library. The image can be stored in an OBS bucket in the same region or in the request body. Only the images in the image library can be searched for.
Searching for Images	You can use the API to search for similar images from the image library.
Querying an Image	You can use the API to check whether an image exists in the image library with the image path.
Deleting an Image	You can use the API to delete an image from the image library with the image path.
Modifying Image Information	You can use the API to modify the information about an existing image in the image library.
Deleting an Instance	You can use the API to delete an instance.

API	Description
Querying the Details About an Instance	You can use the API to query detailed information about a specified instance.

3 Calling APIs

3.1 Constructing a Request

This section describes the structure of a REST API, and uses the IAM API for **obtaining a user token** as an example to describe how to call an API. The obtained token is used to authenticate the calling of other APIs.

Request URI

A request URI is in the following format:

{URI-scheme} :// {Endpoint} / {resource-path} ? {query-string}

Although a request URI is included in a request header, most programming languages or frameworks require the request URI to be separately transmitted, rather than being conveyed in a request message.

Table 3-1 URI parameter description

Parameter	Description
URI-scheme	Protocol used to transmit requests. All APIs use HTTPS.
Endpoint	Domain name or IP address of the server bearing the REST service endpoint. Obtain the value from **Endpoints**. For example, the endpoint of IAM in region **CN-Hong Kong** is **iam.ap-southeast-1.myhuaweicloud.com**.
resource-path	Resource path, that is, API access path. Obtain the value from the URI of the API. For example, the **resource-path** of the API for obtaining a user token is **/v3/auth/tokens**.

Paramet er	Description
query-string	(Optional) Query parameter. Not all APIs have a query parameter. Ensure that a question mark (?) is included before a query parameter that is in the format of **?**_Parameter name_=_Parameter value_. For example, **?limit=10** indicates that a maximum of 10 pieces of data is to be viewed.

For example, to obtain the IAM token in region **CN-Hong Kong**, obtain the endpoint (**iam.ap-southeast-1.myhuaweicloud.com**) of this region and **resource-path** (**/v3/auth/tokens**) in the URI of the API for **obtaining a user token**. Then, assemble the fields as follows:

https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens

Figure 3-1 Example URI
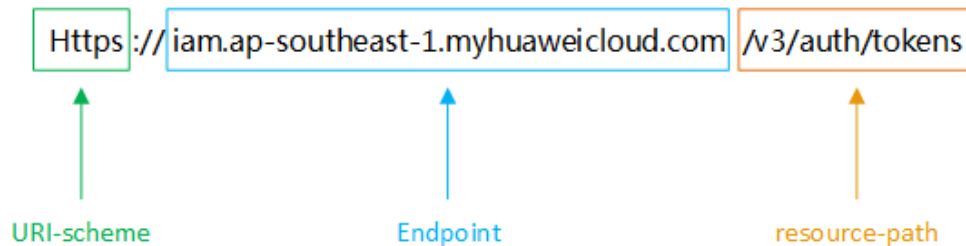
☐☐ **NOTE**

To simplify the URI display, each API provides only **resource-path** and the request method. This is because the **URI-scheme** value of all APIs is **HTTPS**, and the endpoints in a region are the same. Therefore, the two parts are omitted.

Request Method

HTTP-based request methods, which are also called operations or actions, specify the type of operations that you are requesting. **Table 3-2** describes the methods supported by the REST APIs of Image Search.

Table 3-2 HTTP methods

Method	Description
GET	Requests the server to return a specified resource.
PUT	Requests the server to update a specified resource.
POST	Requests the server to add a resource or perform a special operation.
DELETE	Requests the server to delete a specified resource, for example, an object.

Method	Description
HEAD	Request a server resource header.
PATCH	Requests the server to apply partial modifications to a resource. If the resource does not exist, the PATCH method may create a resource.

For example, in the URI for **obtaining a user token**, the request method is **POST**, and the request is as follows:

POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens

Request Header

You can add additional fields, for example, the fields required by a specified URI or HTTP method, to a request header. For example, add **Content-Type** that defines a request body type to request for the authentication information.

Common request headers are as follows:

Table 3-3 Common request headers

Param eter	Description	Mandatory	Example Value
Conten t-Type	Message body type (or format)	Yes	applicati on/json
X-Auth-Token	User token. It is a response to the API for **obtaining a user token** (only this API does not require authentication).	This parameter is mandatory for token authenticati on.	-
x-sdk-date	Time to send a request The time is in *YYYYMMDD'T'HHMMSS'Z'* format. The value is the Greenwich Mean Time (GMT) of the current system.	This parameter is mandatory for AK/SK authenticati on.	2016062 9T101459 Z
Authori zation	Authentication information. The value is obtained from the request signing result.	This parameter is mandatory for AK/SK authenticati on.	-

Param eter	Description	Mandatory	Example Value
Host	Server information, which is obtained from the URL of a service API. This value is *host name*[:*port number*]. If the port number is not specified, the default port is used. The default port number for **https** is **443**.	This parameter is mandatory for AK/SK authenticati on.	-

NOTE

In addition to supporting authentication using tokens, public cloud APIs support authentication using the AK/SK, which uses SDKs to sign a request. During the signature, the **Authorization** (signature authentication) and **X-Sdk-Date** (time when a request is sent) headers are automatically added to the request.

For more information about authentication using the AK/SK, see the **API Signing Guide**.

The API for **obtaining a user token** does not require authentication. Therefore, this API only requires adding the **Content-Type** field. The request with the added **Content-Type** header is as follows:

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

Request Body

A request body is generally sent in structured format. It corresponds to **Content-Type** in the request header and transfers data except the request header. If the request body contains Chinese characters, these characters must be encoded in UTF-8.

The request body varies according to APIs. Certain APIs do not require the request body, such as the GET and DELETE APIs.

For the API of **obtaining a user token**, obtain the request parameters and parameter description in the API request. The following provides an example request with a body included. Replace *username*, *domainname*, ******** (login password), and *xxxxxxxx* (project name) with the actual values. To learn how to obtain your username, account name, and project ID, see **Obtaining the Username, User ID, Project Name, and Project ID**.

NOTE

scope specifies where a token takes effect. In the following example, the token takes effect only on the resources specified by the project ID. You can set **scope** to an account or a project under an account. For details, see **Obtaining a User Token**..

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json

{
    "auth": {
        "identity": {
            "methods": [
                "password"
```

```
        ],
        "password": {
          "user": {
            "name": "username",        //Replace the value with the actual username.
            "password": "********",   //Replace the value with the actual password.
            "domain": {
              "name": "domainname"  //Replace the value with the actual domain name.
            }
          }
        }
      },
      "scope": {
        "project": {
          "name": "xxxxxxxxxx" //Replace the value with the actual project name.
        }
      }
    }
  }
}
```

If all data required by a request is available, you can send the request to call an API through **curl**, **Postman**, or coding. For the API of **obtaining a user token**, **x-subject-token** in the response header is the desired user token. Then, you can use the token to authenticate the calling of other APIs.

3.2 Authentication

You can use either of the following authentication methods when calling APIs:

- Authentication using tokens: General requests are authenticated using tokens.

  To call the Silvan interface of an API gateway, use token authentication. The token whose **roles** is **op_service** must be used. The administrator account of each service has the **op_service** permission.

- Authentication using the AK/SK: Requests are encrypted using the access key ID (AK)/secret access key (SK).

  To use an API gateway to send requests to underlying services, sign the requests using the AK and SK.

Token Authentication

◻ **NOTE**

The validity period of a token is 24 hours. When using a token for authentication, cache it to prevent frequently calling the IAM API.

A token specifies certain permissions in a computer system. Authentication using a token adds the token to a request as its header during API calling to obtain permissions to operate APIs through IAM.

When calling the API to **obtain a user token**, you must set **auth.scope** in the request body to **project**.

For details about how to obtain *username* and *domainname* , see **Obtaining the Account Name and Account ID**. *password* indicates the user password.

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
```

```
                              ],
                              "password": {
                                "user": {
                                  "name": "username",
                                  "password": "*******",
                                  "domain": {
                                    "name": "domainname"
                                  }
                                }
                              }
                          },
                          "scope": {
                            "project": {
                              "name": "xxxxxxxx"
                            }
                          }
                        }
                      }
```

To call the Silvan interface of an API gateway, you must obtain the user's token and add **X-Auth-Token** to the request header of a service API.

After obtaining the token, add the **X-Auth-Token** header to a request to specify the token when calling other APIs. For example, if the token is **ABCDEFJ....**, add **X-Auth-Token: ABCDEFJ....** to a request as follows:

```
GET https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

Procedure

This section describes how to perform token authentication for API calling.

Step 1 Send **POST https://IAM endpoint/v3/auth/tokens** to obtain the endpoint of the Identity and Access Management (IAM) service and the region name in the message body.For details, see **Regions and Endpoints**.

A sample request is as follows:

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",  //Username
          "password": "password",  //Password
          "domain": {
            "name": "domainname"  //Account name
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "ap-southeast-1"  //Region of Image Search
      }
    }
  }
}
```

□ **NOTE**

> Replace the items in italic in the preceding code snippet with actual ones. For details, see the *Identity and Access Management API Reference*.

Step 2 Obtain the token. For details about how to obtain the token, see **Obtaining a User Token** in the *Identity and Access Management API Reference*. The token value is the **X-Subject-Token** value in the response header.

Step 3 To call a service API, add the **X-Auth-Token** header to the request and set its value to the token obtained in **Step 2**.

----End

AK/SK Authentication

□ **NOTE**

> Authentication using the AK/SK supports API requests with a body not larger than 12 MB. For API requests with a larger body, authentication using tokens is recommended.
>
> When the API gateway sends a request to the underlying service, the AK/SK must be used to sign the request. For details, see the *API Request Signing Guide*.

In authentication using the AK/SK, the AK/SK is used to sign a request and add the signature to a request as its header for authentication.

- AK: indicates the ID of the access key. It is the unique identifier associated with the SK. The AK and SK are used together to sign requests cryptographically.

- SK: indicates the secret access key used together with the AK to sign requests cryptographically. AK and SK can be used together to identify a request sender to prevent the request from being modified.

In authentication using the AK/SK, you can use the AK/SK to sign requests based on the signature algorithm or use a dedicated signature SDK to sign the requests. For details about how to sign requests and use the signature SDK, see the *API Request Signing Guide*.

NOTICE

The signature SDK supports only signature, which is different from the SDKs provided by services.

When the AK/SK of a user under a domain is used, add **X-Domain-Id** to the header and set it to the user's domain ID.

Obtaining AK/SK

Step 1 Register with HUAWEI CLOUD and log in to the Image Search management console.

Step 2 Hover the cursor on the username in the upper right corner and select **My Credentials** from the drop-down list. On the **My Credentials** page, click the **Access Keys** tab.

Step 3 Click **Add Access Key**. The **Add Access Key** dialog box is displayed.

Step 4 Enter the login password of the current user and the verification code received in the email or on the mobile phone. Click **OK** to download the access key.

📖 **NOTE**

- For users created in IAM, if no email address or mobile phone number is specified during user creation, you only need to enter the login password for verification.
- To prevent the access key from being leaked, keep it secure.

----End

3.3 Return Result

Status Code

After sending a request, you will receive a response containing the status code, response header, and response body.

A status code is a group of digits ranging from 1*xx* to 5*xx*. It indicates the status of a response. For more information, see **Status Codes**.

If status code **201** is returned for the calling of the API for **obtaining a user token**, the request is successful.

Response Header

A response header corresponds to a request header, for example, **Content-type**.

Figure 3-2 shows the response header for the API of **Obtaining a User Token**, in which **x-subject-token** is the desired user token. Then, you can use the token to authenticate the calling of other APIs.

Figure 3-2 Header of the response to the request for obtaining a user token

Response Body

A response body is generally returned in a structured format, corresponding to **Content-Type** in the response header, and is used to transfer content other than the response header.

The following shows part of the response body for the API of **obtaining a user token**.

```
{
    "token": {
        "expires_at": "2019-02-13T06:52:13.855000Z",
        "methods": [
            "password"
        ],
        "catalog": [
            {
                "endpoints": [
                    {
                        "region_id": "ap-southeast-1",
......
```

If an error occurs during API calling, the system returns an error code and a message to you. The following shows the format of an error response body:

```
{
    "error_msg": "The format of message is error",
    "error_code": "AS.0001"
}
```

In the preceding code snippet, **error_code** is an error code, and **error_msg** describes the error.

4 APIs

4.1 Creating an Instance

Function

You can use the API to create an instance. An image library is created in the instance to store image features.

URI

POST /v1/{project_id}/service

Table 4-1 Path parameters

Name	Mandatory	Type	Description
project_id	Yes	String	Project ID. See **Obtaining a Project ID and Name**.

Request Message

Table 4-2 Request header parameters

Name	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token Used to obtain the permission to call APIs. For details about how to obtain the token, see **Authentication**. The value of X-Subject-Token in the response header is the token.

Table 4-3 Request body parameters

Name	Mandatory	Type	Description
name	Yes	String	Instance name ● Only lowercase letters, digits, hyphens (-), and underscores (_) are allowed. ● The value starts with a letter and contains a maximum of 16 characters.
model	Yes	String	Model name. The following model names are supported: -**common-search**: searches for images of similar content or categories in the image library.
description	No	String	Instance description
level	No	Integer	Image quantity of an instance. The default value is **30000000**. Currently, only the default specifications are supported.
tags	No	Array of strings	Custom image tag. Each instance supports a maximum of 10 tags. Custom tags cannot contain uppercase letters.

Response Parameters

Status code: 200

Table 4-4 Response body parameter

Name	Type	Description
domain	String	Model name or domain name
desc	String	Service instance description
registerDate	Long	Registration time
expiredDate	Long	Expiration time. The value **-1** indicates that the instance will never expire.
level	Integer	Specifications, that is, the image quantity of an instance. The default value is **30000000**.

Name	Type	Description
tags	Array of strings	Custom image tag.
status	String	Instance status. Available enumeration values are as follows: ● **NORMAL** ● **ARREARAGE** ● **CREATION** ● **CREATION_FAILD** ● **DELETING** ● **DELETING_FAILED** ● **ABNORMAL**
instanceName	String	Instance name

Status code: 400

Table 4-5 Response body parameter

Name	Type	Description
error_code	String	Error code of a failed API call. For details, see **Error Codes**. This parameter is not included when the API is successfully called.
error_msg	String	Error message when the API call fails. This parameter is not included when the API is successfully called.

Example Requests

Create an instance named **test_instance** and set the model name to "General image search".

```
POST https://{endpoint}/v1/{project_id}/service

{
  "name" : "test_instance",
  "model" : "common-search",
  "description" : "register service for user model test.",
  "level" : 30000000,
  "tags" : [ "tag1", "tag2" ]
}
```

Example Response

Status code: 200

Example of a successful response

```
{
  "instanceName" : "test_instance ",
  "level" : 30000000,
  "expiredDate" : -1,
  "domain" : "model description",
  "desc" : "register service for user model test.",
  "registerDate" : 1528203334203,
  "tags" : [ "tag1", "tag2" ],
  "status" : "NORMAL"
}
```

Status code: 400

Example of a failed response

```
{
  "error_code" : "IS.3000",
  "error_msg" : "TInstance name is already exists."
}
```

Status Code

Status Code	Description
200	Successful response
400	Failed response

Error Code

For details, see **Error Codes**.

4.2 Adding an Image

Function

You can use the API to import an image to the image library. The image can be stored in an OBS bucket in the same region or in the request body. Only the images in the image library can be searched for.

📖 **NOTE**

- To add or search for an image stored in an OBS bucket, authorize the OBS bucket first. The specific steps are as follows:

  On the **Instance Management** page, click **Offline Import** in the **Operation** column. The **Offline Import** page is displayed. Select the OBS bucket where data is stored and click **Authorize**. When **Authorize** becomes unavailable, OBS authorization is completed. For details about offline import, see **Getting Started > Offline Import**.

URI

POST /v1/{project_id}/{instance_name}/image

Table 4-6 Path parameters

Name	Mandatory	Type	Description
project_id	Yes	String	Project ID. See **Obtaining a Project ID and Name**.
instance_name	Yes	String	Instance name

Request Message

Table 4-7 Request header parameters

Name	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token Used to obtain the permission to call APIs. For details about how to obtain the token, see **Authentication**. The value of X-Subject-Token in the response header is the token.

Table 4-8 Request body parameters

Name	Mandatory	Type	Description
file	No	String	Base64 character string converted from the image. Only the JPEG, JPG, PNG, and BMP formats are supported. The image resolution of the narrow sides of an image must be greater than or equal to 100 pixels, and that of the wide sides cannot exceed 2048 pixels.

Name	Mandatory	Type	Description
path	Yes	String	Image URL. This parameter is mandatory and is used as the ID of an image in the image library. **NOTE** • If the **file** field is specified, the image is obtained from **file** and **path** is used as the image index ID. If the **file** field does not exist or is unspecified, the image needs to be downloaded. In this case, **path** is the address for downloading the image (currently, images can be downloaded only from OBS in the same region as Image Search). **path** is also used as the image index ID.
tags	No	Object	Custom image tag. The total number of keys of each image cannot exceed 10, but the number of values corresponding to each key is not limited. For example, if a key is **animal**, the corresponding values can be **cat**, **dog**, and **bird**. To add a tag name (key), do as follows: 1. Log in to the ImageSearch management console, click **Create Instance**, and configure **Customize Tag**. 2. On the instance management tab page, click an instance name. On the displayed basic information page, add a custom tag.

Response Parameters

Status code: 200

Table 4-9 Response body parameter

Name	Type	Description
result	String	Calling result when the API call succeeds. This parameter is not included when the API fails to be called.

Status code: 400

Table 4-10 Response body parameter

Name	Type	Description
error_code	String	Error code of a failed API call. For details, see **Error Codes**. This parameter is not included when the API is successfully called.
error_msg	String	Error message when the API call fails. This parameter is not included when the API is successfully called.

Example Requests

- Method 1: Use the Base64 code and URL of an image to import the image index library. The image URL is /../image.jpg.
  POST https://{endpoint}/v1/{project_id}/{instance_name}/image

  ```
  {
   "path" : "/../image.jpg",
   "file" : "/9j/4AAQSkZJRgABAgEASABIAAD/
  4RFZRXhpZgAATU0AKgAAAAgABwESAAMAAAABAAEAAAEaAAUAAAABAAAAYgEbAAUAAAABAAAA..."
  }
  ```

- Method 2: Use the URL to import an image to the image index library. The image URL is https://bucketName.obs.myhuaweicloud.com/image/test1.jpg.
  POST https://{endpoint}/v1/{project_id}/{instance_name}/image

  ```
  {
   "path" : "https://bucketName.obs.myhuaweicloud.com/image/test1.jpg"
  }
  ```

- Method 3: Use the image Base64 code and tags to import the image to the image index library. The URL of the image is https://bucketName.obs.myhuaweicloud.com/image/test1.jpg.
  POST https://{endpoint}/v1/{project_id}/{instance_name}/image

  ```
  {
   "path" : "https://bucketName.obs.myhuaweicloud.com/image/test1.jpg",
   "file" : "/9j/4AAQSkZJRgABAgEASABIAAD/
  4RFZRXhpZgAATU0AKgAAAAgABwESAAMAAAABAAEAAAEaAAUAAAABAAAAYgEbAAUAAAABAAAA...",
   "tags" : {
     "test-tags" : "test-image"
   }
  }
  ```

- Method 4: Use the image URL and tags to import the image to the image index library. The image URL is https://bucketName.obs.myhuaweicloud.com/image/test1.jpg.

```
POST https://{endpoint}/v1/{project_id}/{instance_name}/image

{
  "path" : "https://bucketName.obs.myhuaweicloud.com/image/test1.jpg",
  "tags" : {
    "test-tags" : "test-image"
  }
}
```

Example Response

Status code: 200

Example of a successful response

```
{
  "result" : "Success"
}
```

Status code: 400

Example of a failed response

```
{
  "error_code" : "IS.0003",
  "error_msg" : "Arguments of path can not be null."
}
```

Status Code

Status Code	Description
200	Successful response
400	Failed response

Error Code

For details, see **Error Codes**.

4.3 Searching for Images

Function

You can use the API to search for similar images from the image library.

URI

POST /v1/{project_id}/{instance_name}/image/search

Table 4-11 Path parameters

Name	Mandatory	Type	Description
instance_name	Yes	String	Instance name
project_id	Yes	String	Project ID. See **Obtaining a Project ID and Name**.

Request Message

Table 4-12 Request header parameters

Name	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token Used to obtain the permission to call APIs. For details about how to obtain the token, see **Authentication**. The value of X-Subject-Token in the response header is the token.

Table 4-13 Request body parameters

Name	Mandatory	Type	Description
file	No	String	Base64 character string converted from the image. Only the JPEG, JPG, PNG, and BMP formats are supported. The image resolution of the narrow sides of an image must be greater than or equal to 100 pixels, and that of the wide sides cannot exceed 2048 pixels. Configure either this parameter or **path**. If both parameters are set, **file** is used.
path	No	String	Image URL, which is the index ID of an image in the image library. Configure either this parameter or **file**. If both parameters are set, **file** is used.

Name	Mandatory	Type	Description
limit	No	Integer	Number of retrieved images. The value is an integer ranging from 1 to 100. Minimum value: **1** Maximum value: **100** Default value: **10**
offset	No	Integer	Offset used to specify the start position of the search result. The value is an integer greater than or equal to 0. The default value is **0**. Minimum value: **0**
tags	No	Object	Custom image tag. A maximum of 10 tags can be customized. The tags are key-value pairs. To add a tag name (key), do as follows: ● Log in to the ImageSearch management console, click **Create Instance**, and configure **Customize Tag**. ● Log in to the Image Search management console. Go to the **Instance Management** page, and click an instance name. On the **Instance Details** page that is displayed, configure **Customize Tag**. This parameter is mandatory when the image tag is used for search.
is_crop	No	Boolean	Indicates whether to specify an area (parameter **box**) in an image for search. This parameter is valid only for certain models. The value can be **true** or **false**. ● **true**: Specify an area (parameter **box**) in an image for search. ● **false**: Use the entire image to search. The default value is **false**.

Name	Mandatory	Type	Description
box	No	**SearchBoxDetail** object	Area specified in the image.

Table 4-14 SearchBoxDetail

Parameter	Mandatory	Type	Description
x	No	Integer	Minimum value of the horizontal (x) coordinate in an area, in pixels
y	No	Integer	Minimum value of the vertical (y) coordinate in an area, in pixels
width	No	Integer	Width of the specified area, in pixels
height	No	Integer	Height of the specified area, in pixels

Response Parameters

Status code: 200

Table 4-15 Response body parameter

Name	Type	Description
count	Integer	Total number of search records
result	Array of **SearchPictureItem** objects	Search result details

Table 4-16 SearchPictureItem

Name	Type	Description
path	String	Path of the searched image
sim	Float	Similarity between the original image and the searched image. The closer the value reaches **1**, the more similar the images are.
tags	Object	Custom tag name and content

Status code: 400

Table 4-17 Response body parameter

Name	Type	Description
error_code	String	Error code of a failed API call. For details, see **Error Codes**. This parameter is not included when the API is successfully called.
error_msg	String	Error message when the API call fails. This parameter is not included when the API is successfully called.

Example Requests

- Method 1: Use the Base64 code of an image to search for similar images in the image index library. The number of returned images is 10.
  POST https://{endpoint}/v1/{project_id}/{instance_name}/image/search

  ```
  {
    "limit" : 10,
    "offset" : 0,
    "file" : "/9j/4AAQSkZJRgABAgEASABIAAD/
  4RFZRXhpZgAATU0AKgAAAgABwESAAMAAAABAAEAAAEaAAUAAAABAAAAYgEbAAUAAAABAAAA..."
  }
  ```

- Method 2: Use the image URL to search for similar images in the image index library. The number of returned images is 10.
  POST https://{endpoint}/v1/{project_id}/{instance_name}/image/search

  ```
  {
    "limit" : 10,
    "offset" : 0,
    "path" : "https://BucketName.obs.myhuaweicloud.com/image/test1.jpg"
  }
  ```

- Method 3: Use tags to search for similar images in the image index library. The number of returned images is 10.
  POST https://{endpoint}/v1/{project_id}/{instance_name}/image/search

  ```
  {
    "limit" : 10,
    "offset" : 0,
    "tags" : {
      "test-tags" : "test-image"
    }
  }
  ```

- Method 4: Use the Base64 code and tags of an image to search for similar images in the image index library. The number of returned images is 10.
  POST https://{endpoint}/v1/{project_id}/{instance_name}/image/search

  ```
  {
    "limit" : 10,
    "offset" : 0,
    "file" : "/9j/4AAQSkZJRgABAgEASABIAAD/
  4RFZRXhpZgAATU0AKgAAAgABwESAAMAAAABAAEAAAEaAAUAAAABAAAAYgEbAAUAAAABAAAA...",
    "tags" : {
      "test-tags" : "test-image"
  ```

```
  }
 }
```

- Method 5: Use the image URL and tags to search for similar images in the image index library. The number of returned images is 10.

  POST https://{endpoint}/v1/{project_id}/{instance_name}/image/search

```
{
 "limit" : 10,
 "offset" : 0,
 "path" : "https://BucketName.obs.myhuaweicloud.com/image/test1.jpg",
 "tags" : {
   "test-tags" : "test-image"
 }
}
```

- Method 6: Use a specified area to search for similar images in the image index library. The number of returned images is 10.

  POST https://{endpoint}/v1/{project_id}/{instance_name}/image/search

```
{
 "limit" : 10,
 "offset" : 0,
 "path" : "https://BucketName.obs.myhuaweicloud.com/image/test1.jpg",
 "box" : {
   "x" : 100,
   "y" : 200,
   "height" : 50,
   "width" : 20
 }
}
```

Example Response

Status code: 200

Example of a successful response

```
{
 "count" : 2,
 "result" : [ {
   "path" : "https://BucketName.obs.myhuaweicloud.com/CBL025106G/CBL025106G_2.JPG",
   "sim" : 0.51263,
   "tags" : {
     "test-tags" : "test-image"
   }
 } ]
}
```

Status code: 400

Example of a failed response

```
{
 "error_code" : "IS.0003",
 "error_msg" : "Arguments of path can not be null."
}
```

Status Code

Status Code	Description
200	Successful response

Status Code	Description
400	Failed response

Error Code

For details, see **Error Codes**.

4.4 Querying an Image

Function

You can use the API to check whether an image exists in the image library with the image path.

URI

POST /v1/{project_id}/{instance_name}/image/check

Table 4-18 Path parameters

Name	Mandatory	Type	Description
instance_name	Yes	String	Instance name
project_id	Yes	String	Project ID. See **Obtaining a Project ID and Name**.

Request Message

Table 4-19 Request header parameters

Name	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token Used to obtain the permission to call APIs. For details about how to obtain the token, see **Authentication**. The value of X-Subject-Token in the response header is the token.

Table 4-20 Request body parameters

Name	Mandatory	Type	Description
path	No	String	Image URL, which is used as the ID of an image in the image library

Response Parameters

Status code: 200

Table 4-21 Response body parameter

Name	Type	Description
exist	String	Calling result when the API call succeeds. This parameter is not included when the API fails to be called. The value can be **true** or **false**. The default value is **false**. ● **true**: the queried image exists in the image library. ● **false**: the queried image does not exist in the image library.

Status code: 400

Table 4-22 Response body parameter

Name	Type	Description
error_code	String	Error code of a failed API call. For details, see **Error Codes**. This parameter is not included when the API is successfully called.
error_msg	String	Error message when the API call fails. This parameter is not included when the API is successfully called.

Example Requests

Query an image in the index library based on the image path (https://bucketName.obs.myhuaweicloud.com/image/test1.jpg).

```
POST https://{endpoint}/v1/{project_id}/{instance_name}/image/check

{
```

```
    "path" : "https://bucketName.obs.myhuaweicloud.com/image/test1.jpg"
}
```

Example Response

Status code: 200

Example of a successful response

```
{
  "exist" : "true"
}
```

Status code: 400

Example of a failed response

```
{
  "error_code" : "IS.0003",
  "error_msg" : "Arguments of path can not be null."
}
```

Status Code

Status Code	Description
200	Successful response
400	Failed response

Error Code

For details, see **Error Codes**.

4.5 Deleting an Image

Function

You can use the API to delete an image from the image library with the image path.

URI

DELETE /v1/{project_id}/{instance_name}/image

Table 4-23 Path parameters

Name	Mandatory	Type	Description
instance_name	Yes	String	Instance name

Name	Mandatory	Type	Description
project_id	Yes	String	Project ID. See **Obtaining a Project ID and Name**.

Request Message

Table 4-24 Request header parameters

Name	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token Used to obtain the permission to call APIs. For details about how to obtain the token, see **Authentication**. The value of X-Subject-Token in the response header is the token.

Table 4-25 Request body parameters

Name	Mandatory	Type	Description
path	No	String	Image URL, which is used as the ID of an image in the image library

Response Parameters

Status code: 200

Table 4-26 Response body parameter

Name	Type	Description
result	String	Calling result when the API call succeeds. This parameter is not included when the API fails to be called.

Status code: 400

Table 4-27 Response body parameter

Name	Type	Description
error_code	String	Error code of a failed API call. For details, see **Error Codes**. This parameter is not included when the API is successfully called.
error_msg	String	Error message when the API call fails. This parameter is not included when the API is successfully called.

Example Requests

Delete an image from the image library with the image path.

```
DELETE  https://{endpoint}/v1/{project_id}/{instance_name}/image

{
  "path" : "https://BucketName.obs.myhuaweicloud.com/image/test1.jpg"
}
```

Example Response

Status code: 200

Example of a successful response

```
{
  "result" : "Success"
}
```

Status code: 400

Example of a failed response

```
{
  "error_code" : "IS.0003",
  "error_msg" : "Arguments of path can not be null."
}
```

Status Code

Status Code	Description
200	Successful response
400	Failed response

Error Code

For details, see **Error Codes**.

4.6 Modifying Image Information

Function

You can use the API to modify the information about an existing image in the image library.

URI

PUT /v1/{project_id}/{instance_name}/image

Table 4-28 Path parameters

Name	Mandatory	Type	Description
instance_name	Yes	String	Instance name
project_id	Yes	String	Project ID. See **Obtaining a Project ID and Name**.

Request Message

Table 4-29 Request header parameters

Name	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token Used to obtain the permission to call APIs. For details about how to obtain the token, see **Authentication**. The value of X-Subject-Token in the response header is the token.

Table 4-30 Request body parameters

Name	Mandatory	Type	Description
path	Yes	String	Image URL, which is used as the ID of an image in the image library

Name	Mandatory	Type	Description
tags	Yes	Object	Custom image tag. The tags are key-value pairs. ● The tag name (key) must exist in the instance. ● The tag content (value) is a custom value.

Response Parameters

Status code: 200

Table 4-31 Response body parameter

Name	Type	Description
result	String	Calling result when the API call succeeds. This parameter is not included when the API fails to be called.

Status code: 400

Table 4-32 Response body parameter

Name	Type	Description
error_code	String	Error code of a failed API call. For details, see **Error Codes**. This parameter is not included when the API is successfully called.
error_msg	String	Error message when the API call fails. This parameter is not included when the API is successfully called.

Example Requests

Enter an image path, set a custom label, and modify information about an existing image in the image index library.

```
PUT https://{endpoint}/v1/{project_id}/{instance_name}/image

{
 "path" : "https://bucketName.obs.myhuaweicloud.com/image/test1.jpg",
 "tags" : {
   "test-tags" : "test-image"
 }
}
```

Example Response

Status code: 200

Example of a successful response

```
{
  "result" : "Success"
}
```

Status code: 400

Example of a failed response

```
{
  "error_code" : "IS.0003",
  "error_msg" : "Arguments of path can not be null."
}
```

Status Code

Status Code	Description
200	Successful response
400	Failed response

Error Code

For details, see **Error Codes**.

4.7 Deleting an Instance

Function

You can use the API to delete an instance.

URI

DELETE /v1/{project_id}/service/{instance_name}

Table 4-33 Path parameters

Name	Mandatory	Type	Description
instance_name	Yes	String	Instance name
project_id	Yes	String	Project ID. See **Obtaining a Project ID and Name**.

Request Message

Table 4-34 Request header parameters

Name	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token Used to obtain the permission to call APIs. For details about how to obtain the token, see **Authentication**. The value of X-Subject-Token in the response header is the token.

Response Parameters

Status code: 200

Table 4-35 Response body parameter

Name	Type	Description
result	String	Calling result when the API call succeeds. This parameter is not included when the API fails to be called.

Status code: 400

Table 4-36 Response body parameter

Name	Type	Description
error_code	String	Error code of a failed API call. For details, see **Error Codes**. This parameter is not included when the API is successfully called.
error_msg	String	Error message when the API call fails. This parameter is not included when the API is successfully called.

Example Requests

Delete an instance.

```
DELETE https://{endpoint}/v1/{project_id}/service/{instance_name}
```

Example Response

Status code: 200

Returned Result

```
{
  "result" : "success"
}
```

Status code: 400

Example of a failed response

```
{
  "error_code" : "IS.0004",
  "error_msg" : "The authentication token is abnormal."
}
```

Status Code

Status Code	Description
200	Returned result
400	Failed response

Error Code

For details, see **Error Codes**.

4.8 Querying the Details About an Instance

Function

You can use the API to query detailed information about a specified instance.

URI

GET /v1/{project_id}/service/{instance_name}

Table 4-37 Path parameters

Name	Mandatory	Type	Description
instance_name	Yes	String	Instance name
project_id	Yes	String	Project ID. See **Obtaining a Project ID and Name**.

Request Message

Table 4-38 Request header parameters

Name	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token Used to obtain the permission to call APIs. For details about how to obtain the token, see **Authentication**. The value of X-Subject-Token in the response header is the token.

Response Parameters

Status code: 200

Table 4-39 Response body parameter

Name	Type	Description
domain	String	Model name or domain name
desc	String	Description
registerDate	Long	Registration time
expiredDate	Long	Expiration time. The value **-1** indicates that the instance will never expire.
level	Integer	Specifications, that is, the image quantity of an instance. The default value is **30000000**.
tags	Array of strings	Custom image tag
status	String	Instance status. Available enumeration values are as follows: ● **NORMAL** ● **ARREARAGE** ● **CREATION** ● **CREATION_FAILD** ● **DELETING** ● **DELETING_FAILED** ● **ABNORMAL**
instanceName	String	Instance name

Status code: 400

Table 4-40 Response body parameter

Name	Type	Description
error_code	String	Error code of a failed API call. For details, see **Error Codes**. This parameter is not included when the API is successfully called.
error_msg	String	Error message when the API call fails. This parameter is not included when the API is successfully called.

Example Request

Query detailed information about a specified instance.

GET https://{endpoint}/v1/{project_id}/service/{instance_name}

Example Response

Status code: 200

Example of a successful response

```
{
 "instanceName" : "test-instance",
 "level" : 30000000,
 "expiredDate" : 1530848365207,
 "domain" : "model description",
 "desc" : "register service for instance test.",
 "registerDate" : 1529377136279,
 "tags" : [ "tag1", "tag2" ],
 "status" : "NORMAL"
}
```

Status code: 400

Example of a failed response

```
{
 "error_code" : "IS.0004",
 "error_msg" : "The authentication token is abnormal."
}
```

Status Code

Status Code	Description
200	Successful response
400	Failed response

Error Code

For details, see **Error Codes**.

5 Appendix

5.1 Status Codes

Table 5-1 Status codes

Status Code	Value	Description
100	Continue	The client should continue with its request. This provisional response informs the client that part of the request has been received and has not yet been rejected by the server.
101	Switching Protocols	The protocol should be switched. The protocol can only be switched to a newer protocol. For example, the current HTTPS protocol is switched to a later version of HTTP.
201	Created	The request for creating a resource has been fulfilled.
202	Accepted	The request has been accepted, but the processing has not been completed.
203	Non-Authoritative Information	Unauthorized information. The request is successful.

Status Code	Value	Description
204	NoContent	The server has successfully processed the request, but does not return any content. The status code is returned in response to an HTTP OPTIONS request.
205	Reset Content	The server has successfully processed the request, but does not return any content.
206	Partial Content	The server has successfully processed a part of the GET request.
300	Multiple Choices	There are multiple options for the location of the requested resource. The response contains a list of resource characteristics and addresses from which a user terminal (such as a browser) can choose the most appropriate one.
301	Moved Permanently	This and all future requests should be permanently directed to the given URI indicated in this response.
302	Found	The requested resource resides temporarily under a different URI.
303	See Other	Retrieve a location. The response to the request can be found under a different URI, and should be retrieved using a GET or POST method.
304	Not Modified	The requested resource has not been modified. In such case, there is no need to retransmit the resource since the client still has a previously-downloaded copy.
305	Use Proxy	The requested resource is available only through a proxy.
306	Unused	This HTTP status code is no longer used.

Status Code	Value	Description
400	BadRequest	The request is invalid. The client should not repeat the request without modifications.
401	Unauthorized	The status code is returned after the client provides the authentication information, indicating that the authentication information provided by the client is incorrect or invalid.
402	Payment Required	This status code is reserved for future use.
403	Forbidden	The server has received the request and understood it, but the server is refusing to respond to it. The client should not repeat the request without modifications.
404	NotFound	The requested resource could not be found. The client should not repeat the request without modifications.
405	MethodNotAllowed	The method specified in the request is not supported by the requested resource. The client should not repeat the request without modifications.
406	Not Acceptable	The server could not fulfill the request according to the content characteristics of the request.
407	Proxy Authentication Required	This code is similar to 401, but indicates that the client must first authenticate itself with the proxy.

Status Code	Value	Description
408	Request Time-out	The server timed out waiting for the request. The client may repeat the request without modifications at any time later.
409	Conflict	The request could not be processed due to a conflict with the current state of the resource. This status code indicates that the resource that the client attempts to create already exits, or the request fails to be processed because of the update of the conflict request.
410	Gone	The requested resource cannot be found. The status code indicates that the requested resource has been deleted permanently.
411	Length Required	The server refused to process the request without a defined **Content-Length**.
412	Precondition Failed	The server does not meet one of the preconditions that the requester puts on the request.
413	Request Entity Too Large	The request is larger than the server is willing or able to process. The server may disable the connection to prevent the client from sending requests consecutively. If the server cannot process the request temporarily, the response will contain a **Retry-After** header field.
414	Request-URI Too Large	The URI provided was too long for the server to process.
415	Unsupported Media Type	The server does not support the media type in the request.
416	Requested range not satisfiable	The requested range is invalid.

Status Code	Value	Description
417	Expectation Failed	The server fails to meet the requirements of the **Expect** request-header field.
422	UnprocessableEntity	The request was well-formed but was unable to be followed due to semantic errors.
429	TooManyRequests	The client has sent more requests than its rate limit is allowed within a given amount of time, or the server has received more requests than it is able to process within a given amount of time. In this case, the client should repeat requests after the time specified in the **Retry-After** header of the response expires.
500	InternalServerError	The server is able to receive but unable to understand the request.
501	Not Implemented	The server does not support the requested function.
502	Bad Gateway	The server acting as a gateway or proxy has received an invalid request from a remote server.
503	ServiceUnavailable	The requested service is invalid. The client should not repeat the request without modifications.
504	ServerTimeout	The request cannot be fulfilled within a given time. This status code is returned to the client only when the **Timeout** parameter is specified in the request.
505	HTTP Version not supported	The server does not support the HTTP protocol version used in the request.

5.2 Error Codes

If an error occurs during API calling, no result is returned. You can locate the cause of the error according to the error code of each API.

If an error occurs during API calling, the returned message body contains the specific error code and error information. If you fail to locate the cause of an error, contact HUAWEI CLOUD service support and provide the error code so that we can help you solve the problem as soon as possible.

If an error code starting with **APIGW** is returned after you call an API, rectify the fault by referring to the instructions provided in **API Gateway Error Codes**.

Table 5-2 Error codes

Status Code	Error Code	Error Message	Description	Solution
400	IS.0001	The request parameter is missing or empty.	The request parameter is missing or empty.	Check whether the request parameters are valid.
400	IS.0002	The type of the request parameter is not supported.	The type of the request parameter is not supported.	Check whether the request parameters are valid.
400	IS.0003	The request parameter cannot be empty.	The request parameter cannot be empty.	Check whether the request parameters are valid.
400	IS.0004	Exception verifying token.	Exception verifying token.	Check whether the token is up to date.
400	IS.0005	The header is missing or empty.	The header is missing or empty.	Check whether the header is correct.
400	IS.0006	The request body is too big.	The request body is too big.	Shrink the request body to the size allow.
400	IS.0007	Invalid instance name.	Invalid instance name.	Check whether the instance name is correct.
400	IS.0008	The JSON format of the input data is incorrect.	The JSON format of the input data is incorrect.	Check whether the JSON format is valid.

Status Code	Error Code	Error Message	Description	Solution
400	IS.0009	The request failed due to an internal error.	The request failed due to an internal error.	Locate the fault based on the error information returned by the service.
400	IS.0010	The request method is not allowed.	The request method is not allowed.	Use allowed request method instead.
400	IS.0011	The user is not allowed to use the request method.	The user is not allowed to use the request method.	Grant permission to current user, or send the request as another user.
400	IS.0012	Operation timed out.	Operation timed out.	Check whether the service is running properly.
400	IS.0013	The token is empty.	The token is empty.	Use a valid token.
400	IS.0014	Instance status is abnormal.	Instance status is abnormal.	Check whether the instance status is normal.
400	IS.0015	The model file does not exist.	The model file does not exist.	Check whether the model file exists.
400	IS.0016	The path is invalid.	The path is invalid.	Check whether the path is valid.
400	IS.0017	The request is not authorized.	The request is not authorized.	Apply for authorization.
400	IS.0018	The OBT is not applied for.	The OBT is not applied for.	Apply for OBT.
400	IS.0019	The user service ID does not match the project ID.	The user service ID does not match the project ID.	Enter the matching ID.
400	IS.0020	The value cannot be obtained.	The value cannot be obtained.	Check whether the user is blacklisted. If yes, remove the user from the blacklist.
400	IS.0021	The token has expired.	The token has expired.	Obtain the latest token.

Status Code	Error Code	Error Message	Description	Solution
400	IS.0022	The current user has restricted rights.	The current user has restricted rights.	Check the account balance or contact the administrator.
400	IS.1001	The service does not exist.	The service does not exist.	Check whether the service name is correct.
400	IS.1002	No tag value.	No tag value.	Check whether the input image contains tag information.
400	IS.1003	Too many tags.	Too many tags.	Use appropriate number of tags.
400	IS.1004	The tag does not match.	The tag does not match.	Enter a correct tag key.
400	IS.1005	Failed to add the image.	Failed to add the image.	Check whether the request parameters are valid.
400	IS.1006	Failed to delete the image.	Failed to delete the image.	Check whether the request parameters are valid.
400	IS.1007	Failed to update the image.	Failed to update the image.	Check whether the request parameters are valid.
400	IS.1008	Failed to search for the image.	Failed to search for the image.	Check whether the request parameters are valid.
400	IS.1009	Failed to query the image.	Failed to query the image.	Check whether the request parameters are valid.
400	IS.1010	The value of limit is incorrect.	The value of **limit** is incorrect.	The value of **limit** must be a positive integer.
400	IS.1011	The value of offset is incorrect.	The value of **offset** is incorrect.	The value of **offset** must be a positive integer or 0.
400	IS.1015	Too many images in the instance.	Too many images in the instance.	Contact HUAWEI CLOUD customer service to scale up the instance.

Stat us Code	Error Code	Error Message	Description	Solution
400	IS.1016	Failed to detect the image.	Failed to detect the image.	Check whether the industry type is supported.
400	IS.2000	Failed to download the image.	Failed to download the image.	Check whether the image path exists.
400	IS.2001	The image format is not supported.	The image format is not supported.	Change the image format to JPEG, JPG, PNG, or BMP.
400	IS.2002	The image file has been damaged.	The image file has been damaged.	Check whether the image file is complete.
400	IS.2003	The OBS object cannot be obtained.	The OBS object cannot be obtained.	Check whether OBS is accessible. If it is, check whether the object exists.
400	IS.2004	The OBS bucket does not exist.	The OBS bucket does not exist.	Check whether the OBS bucket exists.
400	IS.2005	Failed to save the temporary file.	Failed to save the temporary file.	Check whether the image search service is available.
400	IS.2006	The image is not indexed.	The image is not indexed.	Check whether the image search service is available.
400	IS.2007	Base64 decoding of the input data is incorrect.	Base64 decoding of the input data is incorrect.	Check whether the entered data is correct.
400	IS.2008	The request parameters must contain path information.	The request parameters must contain path information.	Add path information to the request parameter.
400	IS.2009	The request parameters must contain tag information.	The request parameters must contain tag information.	Add tag information to the request parameter.
400	IS.2010	The request parameter does not exist.	The request parameter does not exist.	Check whether the request parameter is complete.

Status Code	Error Code	Error Message	Description	Solution
400	IS.2011	OBS is not authorized to obtain images.	OBS is not authorized to obtain images.	Check whether OBS is authorized.
400	IS.2012	The image path cannot be empty.	The image path cannot be empty.	Include the **path** field in the request parameter.
400	IS.2013	The path and file in the request parameter cannot be empty at the same time.	The **path** and **file** in the request parameter cannot be empty at the same time.	The **path** and **file** in the request parameter cannot be empty at the same time.
400	IS.2014	Failed to create the cache file path.	Failed to create the cache file path.	Resend the request.
400	IS.2015	The request parameter must contain path or file.	The request parameter must contain **path** or **file**.	The request parameter must contain **path** or **file**.
400	IS.2016	Failed to create the OBS client.	Failed to create the OBS client.	Check whether the OBS bucket is authorized.
400	IS.2019	The instance does not exist.	The instance does not exist.	Check whether the instance exists.
400	IS.2100	Failed to add the image.	Failed to add the image.	Check whether the image search service is available.
400	IS.2101	Failed to delete the image.	Failed to delete the image.	Check whether the image search service is available.
400	IS.2102	Failed to search for the image.	Failed to search for the image.	Check whether the image search service is available.
400	IS.2103	Failed to check the image.	Failed to check the image.	Check whether the image search service is available.
400	IS.2104	Failed to update the image information.	Failed to update the image information.	Check whether the update request is correct.

Status Code	Error Code	Error Message	Description	Solution
400	IS.2105	The service is not running properly.	The service is not running properly.	Restart the service or contact the administrator.
400	IS.2106	The image is too big.	The image is too big.	The image size cannot exceed 2048*2048.
400	IS.2107	The image is too small.	The image is too small.	The image cannot be smaller than 100*100.
400	IS.2108	An error occurred while the service is running.	An error occurred while the service is running.	Check whether a correct method is used to call the API.
400	IS.2109	Failed to connect to the gallery.	Failed to connect to the gallery.	Check whether the instance status is normal.
400	IS.2111	Failed to perform image detection.	Failed to perform image detection.	Check whether the industry parameters are supported.
400	IS.2112	The request parameter must contain image.	The request parameter must contain **image**.	Check whether the **image** parameter exists in the request parameters.
400	IS.3000	The service name already exists.	The service name already exists.	The service already exists.
400	IS.3001	Failed to save the service information.	Failed to save the service information.	Check whether the image search service is available.
400	IS.3002	Failed to delete services from some nodes.	Failed to delete services from some nodes.	Check whether the image search service is available.
400	IS.3003	Failed to delete the instance from the node.	Failed to delete the instance from the node.	Check whether the image search service is available.
400	IS.3004	The model cannot be used.	The model cannot be used.	Check whether the model name and path are correct.
400	IS.3005	The service name does not exist.	The service name does not exist.	The service name is incorrect or the service is not created.

Status Code	Error Code	Error Message	Description	Solution
400	IS.3006	The model is in use.	The model is in use.	The model is in use and does not need to be added again.
400	IS.3007	The user does not exist.	The user does not exist.	Check whether the username is correct.
400	IS.3008	Failed to start the instance.	Failed to start the instance.	Check whether the image search service is available.
400	IS.3009	Decompressing failed.	Decompressing failed.	Check whether the image search service is available.
400	IS.3010	Failed to start the service.	Failed to start the service.	Check whether the image search service is available.
400	IS.3011	Failed to initialize the configuration file.	Failed to initialize the configuration file.	Check whether the image search service is available.
400	IS.3012	No management instance is found.	No management instance is found.	Check whether the image search service is available.
400	IS.3013	Failed to obtain the instance.	Failed to obtain the instance.	Check whether the image search service is available.
400	IS.3014	The model name does not exist.	The model name does not exist.	Check whether the model name is correct.
400	IS.3015	Failed to obtain the OBS bucket.	Failed to obtain the OBS bucket.	Check whether the OBS bucket name is correct.
400	IS.3016	Failed to authorize the OBS bucket.	Failed to authorize the OBS bucket.	Check whether OBS is running properly.
400	IS.3017	The OBS bucket is empty.	The OBS bucket is empty.	There is nothing in the OBS bucket.
400	IS.3018	Failed to obtain the OBS bucket list.	Failed to obtain the OBS bucket list.	Check whether the OBS operation is correct.

Status Code	Error Code	Error Message	Description	Solution
400	IS.3019	Failed to add the OBS bucket.	Failed to add the OBS bucket.	Check whether the OBS operation is correct.
400	IS.3020	Failed to obtain the bucket listening event.	Failed to obtain the bucket listening event.	Check whether the OBS operation is correct.
400	IS.3021	No instance is found.	No instance is found.	Check whether the instance exists.
400	IS.3022	Failed to change the instance name.	Failed to change the instance name.	Check whether the request is valid.
400	IS.3023	The instance tag does not exist.	The instance tag does not exist.	Check whether the instance tag is created.
400	IS.3024	Too many instance tags.	Too many instance tags.	The number of instance tags cannot exceed 10.
400	IS.3025	The instance tag cannot be deleted.	The instance tag cannot be deleted.	Currently, deleting instance tags is not supported.
400	IS.3026	Failed to add the instance tag.	Failed to add the instance tag.	Check whether the adding request is correct.
400	IS.3027	Failed to upload the model to OBS.	Failed to upload the model to OBS.	Check whether OBS is running properly.
400	IS.3028	Failed to create the instance.	Failed to create the instance.	Check whether the creation request is correct.
400	IS.3029	Failed to obtain the instance model.	Failed to obtain the instance model.	Check whether the instance model exists.
400	IS.3030	The instance cannot be deleted because it is being created.	The instance cannot be deleted because it is being created.	It is impossible to delete an instance that is being created.
400	IS.3031	Failed to obtain the statistics.	Failed to obtain the statistics.	Check whether the request is valid and whether the service is available.

Status Code	Error Code	Error Message	Description	Solution
400	IS.3032	The instance is being deleted.	The instance is being deleted.	The instance is already being deleted.
400	IS.3033	Failed to obtain the batch import task.	Failed to obtain the batch import task.	Check whether the batch import data exists.
400	IS.3034	The blacklist or whitelist is invalid.	The blacklist or whitelist is invalid.	Check the blacklist and whitelist.
400	IS.3035	The instance scale does not exist.	The instance scale does not exist.	Check whether the instance scale is correct.
400	IS.3036	The user instance description is too long.	The user instance description is too long.	The description can contain a maximum of 512 characters.
400	IS.3037	A bucket event exists in the instance.	A bucket event exists in the instance.	The instance cannot have multiple bucket events.
400	IS.3038	An offline import task is running on the instance. The instance cannot be deleted.	An offline import task is running on the instance. The instance cannot be deleted.	Wait until the offline import task is complete.
400	IS.3039	Failed to reset the instance scale.	Failed to reset the instance scale.	Check the service status and try again.
400	IS.3040	Failed to reset the instance to a smaller scale.	Failed to reset the instance to a smaller scale.	The instance can be reset to a larger scale only.
400	IS.3041	Failed to obtain the instance scale.	Failed to obtain the instance scale.	Check the service status and try again.
400	IS.3050	Failed to obtain the OBS bucket folder list.	Failed to obtain the OBS bucket folder list.	The OBS bucket does not have a folder list.

Status Code	Error Code	Error Message	Description	Solution
400	IS.3051	The bucket name in the path does not match the name of the selected OBS bucket.	The bucket name in the path does not match the name of the selected OBS bucket.	Check whether the bucket name in the path is the same as the selected bucket name.
400	IS.3052	The path does not contain the file directory or list in the bucket.	The path does not contain the file directory or list in the bucket.	Check whether the entered path is correct.
400	IS.3053	The file list or directory in the path does not exist.	The file list or directory in the path does not exist.	Check whether the entered path is correct.
400	IS.4000	Failed to download the system package.	Failed to download the system package.	Check whether the image search service is available.
400	IS.4001	The OBS bucket is empty.	The OBS bucket is empty.	The OBS bucket is empty.
400	IS.4002	The service cannot be initialized.	The service cannot be initialized.	Check whether the image search service is available.
400	IS.4003	Failed to obtain the search result.	Failed to obtain the search result.	Check whether the image search service is available.
400	IS.4004	The instance name does not exist.	The instance name does not exist.	Check whether the instance name exists.
400	IS.4005	Failed to create the offline import task.	Failed to create the offline import task.	Restart the offline import task.
400	IS.4006	The account is frozen.	The account is frozen.	Check whether the account is in arrears. If yes, it will unfreeze after recharge.

Stat us Code	Error Code	Error Message	Description	Solution
400	APIG.01 01	The API does not exist or has not been published.	The API does not exist or has not been published.	Check whether the entered API information is correct and whether the service is available in the corresponding region. Check whether the URL of the API is spelled correctly and whether the HTTP request method (such as POST and GET) is correct. Check whether the domain name and URI configured for service call are correct. For details about the domain name, see **Regions and Endpoints**.
400	APIG.03 01	1. Incorrect IAM authentication information: Failed to decrypt the token. Check whether the entered token is complete. 2. The validity period of a token is 24 hours. If the token expires, obtain a new token and pass it. 3. The AK/SK authentication fails. Check whether the AK/SK is correct and whether the account is restricted due to arrears.	1. Incorrect IAM authentication information: Failed to decrypt the token. Check whether the entered token is complete. 2. The validity period of a token is 24 hours. If the token expires, obtain a new token and pass it. 3. The AK/SK authentication fails. Check whether the AK/SK is correct and whether the account is restricted due to arrears.	**decrypt token fail**: The token request authentication information of **x-auth-token** in the HTTP request header is incorrect. Check the sent request and token. **token expires**: The token expires. Obtain a new token and pass it. **verify aksk signature fail**: Check whether the AK and SK are correct and whether the account is in arrears.

Status Code	Error Code	Error Message	Description	Solution
400	APIG.0201	The request body is oversized.	The request body is oversized.	Check whether the request body is oversized. For details, see **Constraints**.
400	APIG.0308	The request is sent too fast and exceeds the default rate limit of the service.	The request is sent too fast and exceeds the default rate limit of the service.	The request is sent too fast and reaches the rate limit of the API. Reduce the request speed. If you have higher requirements on the request speed, submit a **service ticket**.
400	ModelArts.0203	Invalid token.	Invalid token.	Check whether the token is correct.
400	ModelArts.4101	The token is empty.	The token is empty.	The HTTP request header does not contain the token request authentication information of **x-auth-token**. Check the request.
400	ModelArts.4102	Failed to parse the token.	Failed to parse the token.	The token request authentication information of **x-auth-token** in the HTTP request header is incorrect. Check the sent request and token.
400	ModelArts.4103	The token is invalid.	The token is invalid.	The token request authentication information of **x-auth-token** in the HTTP request header is incorrect. Check the sent request and token.
400	ModelArts.4104	The length of the request body is invalid.	The length of the request body is invalid.	Check the request body length.
400	ModelArts.4105	The JSON format of the request body is incorrect.	The JSON format of the request body is incorrect.	Check the JSON format of the request body.

Status Code	Error Code	Error Message	Description	Solution
400	ModelArts.4106	The account is restricted.	The account is restricted.	Check the user resource status. For details about the account restriction reason, see the description of the account center.
400	ModelArts.4107	An exception occurred when obtaining the temporary AK/SK.	An exception occurred when obtaining the temporary AK/SK.	Contact HUAWEI CLOUD technical support and submit a **service ticket**.
400	ModelArts.4201	The request URL does not contain the service ID.	The request URL does not contain the service ID.	Check the service ID in the request URL.
400	ModelArts.4202	The request URL format is invalid.	The request URL format is invalid.	Check the request URL format.
400	ModelArts.4203	No access permission.	No access permission.	Check the access permission.
400	ModelArts.4204	The API is not subscribed to.	The API is not subscribed to.	Subscribe to this API. For details, see **Subscribing to a Service**. If the service has been subscribed to, check whether the region where the service is subscribed to is the same as the region where the service is called. If the region is correct, check whether the URL of the API is spelled correctly and whether the HTTP request method (such as POST and GET) is correct.
400	ModelArts.4601	The external URL is invalid.	The external URL is invalid.	Check whether the entered URL is correct.

Status Code	Error Code	Error Message	Description	Solution
400	ModelArts.4603	The file failed to be downloaded from the external URL.	The file failed to be downloaded from the external URL.	Check whether the entered URL is correct.
400	ModelArts.4702	The OBS agency failed to be queried.	The OBS agency failed to be queried.	Check whether the OBS agency has been enabled for the service.
400	ModelArts.4703	The OBS URL is invalid.	The OBS URL is invalid.	Check the entered OBS URL.
400	ModelArts.4704	Failed to obtain the OBS file.	Failed to obtain the OBS file.	Failed to download the OBS file. Check whether the file exists.
400	ModelArts.4705	The OBS file is oversized.	The OBS file is oversized.	Use a file that meets the service size limit as the input. For details, see **Constraints**.
400	ModelArts.4706	The OBS file does not exist.	The OBS file does not exist.	Failed to download the OBS file. Check whether the file exists.
400	Others	If other error codes are displayed, submit a **service ticket**.		

5.3 Obtaining a Project ID and Name

Scenarios

A project ID or name is required for some requests when an API is called. Therefore, obtain the project ID and name before calling the API. Use either of the following methods:

- **Obtaining a Project ID from the Console**
- **Obtaining a Project ID by Calling an API**

Obtaining a Project ID from the Console

1. Log in to the management console.

2. Hover the cursor over your username in the upper right corner and click **My Credentials** from the drop-down list.

3. On the **My Credentials** page, view the username and account name and view projects in the project list.

Figure 5-1 Viewing project ID
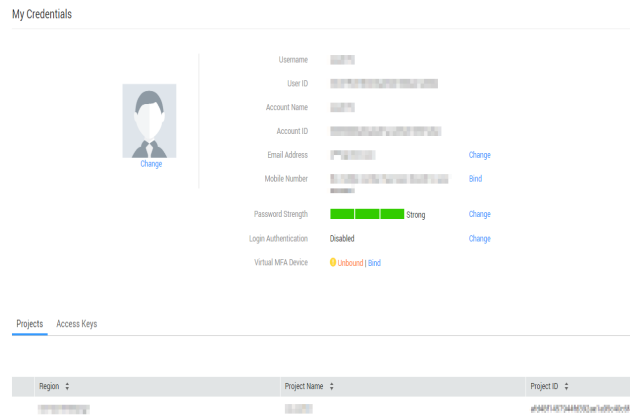
If there are multiple projects in one region, expand **Region** and view subproject IDs in the **Project ID** column.

Obtaining a Project ID by Calling an API

A project ID can be obtained by calling a specific API. For details, see **Querying Project Information Based on the Specified Criteria**.

The API for obtaining a project ID is **GET https://**{*iam-endpoint*}**/v3/projects/**. You can obtain the value from **Regions and Endpoints**.

The following is an example response. For example, if Image Search is deployed in the **ap-southeast-1** region, the value of **name** in the response body is **ap-southeast-1**. The value of **id** in **projects** is the project ID.

```
{
    "projects": [{
        "domain_id": "65382450e8f64ac0870cd180d14e684b",
        "is_domain": false,
        "parent_id": "65382450e8f64ac0870cd180d14e684b",
        "name": "ap-southeast-1",
        "description": "",
        "links": {
            "next": null,
            "previous": null,
            "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
        },
        "id": "a4a5d4098fb4474fa22cd05f897d6b99",
        "enabled": true
    }],
    "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects"
    }
}
```

5.4 Obtaining an Account Name and ID

When you call APIs, certain requests require the account name and ID. To obtain an account name and ID, do as follows:

1. Sign up and log in to the console.

2. Hover the cursor on the username and choose **My Credentials** from the drop-down list.

On the **API Credentials** page, view the account name and ID.