



人脸识别服务 API 操作任务指南

1 任务介绍

本次任务将主要介绍如何使用人脸识别服务API接口，进行人脸检测、人脸比对、人脸搜索。

2 任务步骤

2.1 API 使用方法

参考《人脸识别服务申请及调用API接口查询服务》2.2节，或查看人脸识别服务帮助文档[《API简介》](#)

2.2 API 说明

2.2.1 人脸检测

功能介绍

对输入图片进行人脸检测和分析。

URI

- URI 格式

```
POST /v1/{project_id}/face-detect
```

- 参数说明

名称	是否必选	参数类型	说明
project_id	是	String	项目编号

请求

- 请求样例（方式一：使用图片的 BASE64 编码）

```
POST https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-detect
Request Header:
Content-Type: application/json
X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDt...
Request Body:
```

```
{
  "image base64": "/9j/4AAQSkZJRgABAgEASABIAAD",
  "attributes": "0,1,2,3,4,5"
}
```

- 请求样例（方式二：使用图片文件）

```
POST https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-detect
Request Header:
X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQEExDT...
Request Body:
  image_file: File(图片文件)
  attributes: 0,1,2,3,4,5
```

- 请求样例（方式三：使用图片 url 使用图片 url）

```
POST https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-detect
Request Header:
Content-Type: application/json
X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQEExDT...
Request Body:
{
  "image url": "/BucketName/ObjectName",
  "attributes": "0,1,2,3,4,5"
}
```

- 参数说明

名称	是否必选	参数类型	说明
image_url	与 image_file、image_base64 三选一	String	图片的 URL 路径，目前仅支持华为云上 OBS 提供的临时授权。
image_file	与 image_url、image_base64 三选一	File	本地图片文件。上传文件时，请求格式为 multipart。图片不超过 12M。
image_base64	与 image_file、image_url 三选一	String	图像数据，base64 编码，要求： <ul style="list-style-type: none">base64 编码后大小不超过 12M。支持 JPG/JPEG/BMP/PNG 格式。

名称	是否必选	参数类型	说明
attributes	否	String	<p>是否返回人脸属性，希望获取的属性列表，多个属性间使用逗号(,) 隔开。目前支持的属性有：</p> <ul style="list-style-type: none"> • 0: 人脸姿态 • 1: 性别 • 2: 年龄 • 3: 人脸关键点 • 4: 装束（帽子、眼镜） • 5: 笑脸

响应

- 响应样例

```
{
  "faces": [
    {
      "bounding box": {
        "width": 174,
        "top left y": 37,
        "top left x": 22,
        "height": 174
      },
      "attributes": {
        "age": 35,
        "smile": "yes",
        "gender": "male",
        "dress": {
          "glass": "none",
          "hat": "none"
        },
        "headpose": {
          "pitch angle": -3.8639126,
          "roll angle": -3.988193,
          "yaw angle": -1.0292832
        }
      },
      "landmark": {
        "eyebrow contour": {
          "point": [
            {
              "x": 158.78517,
              "y": 90.02418
            },
            {
              "x": 150.86162,
              "y": 82.432076
            }
          ]
        }
      }
    }
  ]
}
```

```
    },  
    {  
      "x": 131.3591,  
      "y": 80.78191  
    },  
    {  
      "x": 121.39423,  
      "y": 87.62841  
    },  
    {  
      "x": 132.25839,  
      "y": 88.44499  
    },  
    {  
      "x": 149.63844,  
      "y": 89.3718  
    },  
    {  
      "x": 59.65225,  
      "y": 99.64757  
    },  
    {  
      "x": 65.96038,  
      "y": 88.92635  
    },  
    {  
      "x": 85.72037,  
      "y": 85.07116  
    },  
    {  
      "x": 97.09234,  
      "y": 89.34221  
    },  
    {  
      "x": 86.07851,  
      "y": 92.297516  
    },  
    {  
      "x": 67.51544,  
      "y": 95.58905  
    }  
  ]  
},  
"eyes contour": {  
  "point": [  
    {  
      "x": 69.22003,  
      "y": 105.52183  
    },  
    {  
      "x": 81.599174,  
      "y": 98.70017  
    },  
    {  
      "x": 94.22879,  
      "y": 103.18641  
    }  
  ]  
}
```

```
    },  
    {  
      "x": 82.32345,  
      "y": 107.02255  
    },  
    {  
      "x": 150.97876,  
      "y": 99.29232  
    },  
    {  
      "x": 137.91818,  
      "y": 93.49849  
    },  
    {  
      "x": 125.78866,  
      "y": 100.8499  
    },  
    {  
      "x": 138.81555,  
      "y": 102.29028  
    },  
    {  
      "x": 75.16096,  
      "y": 100.01521  
    },  
    {  
      "x": 75.6541,  
      "y": 107.08471  
    },  
    {  
      "x": 89.18221,  
      "y": 106.11414  
    },  
    {  
      "x": 88.87559,  
      "y": 98.87617  
    },  
    {  
      "x": 130.7761,  
      "y": 95.49036  
    },  
    {  
      "x": 131.64914,  
      "y": 102.47154  
    },  
    {  
      "x": 145.64833,  
      "y": 101.04822  
    },  
    {  
      "x": 145.52222,  
      "y": 94.39164  
    }  
  ]  
},  
"face_contour": {
```

```
"point": [  
  {  
    "x": 50.049454,  
    "y": 104.4499  
  },  
  {  
    "x": 52.502007,  
    "y": 123.68245  
  },  
  {  
    "x": 55.028637,  
    "y": 143.12273  
  },  
  {  
    "x": 59.972454,  
    "y": 163.16571  
  },  
  {  
    "x": 69.73235,  
    "y": 180.53296  
  },  
  {  
    "x": 81.76993,  
    "y": 195.74933  
  },  
  {  
    "x": 96.15602,  
    "y": 210.08699  
  },  
  {  
    "x": 115.32445,  
    "y": 215.14973  
  },  
  {  
    "x": 134.04883,  
    "y": 209.08778  
  },  
  {  
    "x": 147.96034,  
    "y": 194.2003  
  },  
  {  
    "x": 158.3112,  
    "y": 178.27126  
  },  
  {  
    "x": 165.62479,  
    "y": 159.71454  
  },  
  {  
    "x": 168.95554,  
    "y": 139.12373  
  },  
  {  
    "x": 169.59795,  
    "y": 119.938416  
  },  
]
```

```
    },  
    {  
      "x": 169.9221,  
      "y": 99.658844  
    }  
  ]  
},  
"mouth contour": {  
  "point": [  
    {  
      "x": 82.56091,  
      "y": 160.6078  
    },  
    {  
      "x": 91.546906,  
      "y": 154.97893  
    },  
    {  
      "x": 101.45366,  
      "y": 152.51962  
    },  
    {  
      "x": 111.45223,  
      "y": 153.75433  
    },  
    {  
      "x": 122.52581,  
      "y": 152.07089  
    },  
    {  
      "x": 133.3687,  
      "y": 152.31891  
    },  
    {  
      "x": 142.48122,  
      "y": 154.72552  
    },  
    {  
      "x": 136.62144,  
      "y": 167.81747  
    },  
    {  
      "x": 126.79148,  
      "y": 175.79927  
    },  
    {  
      "x": 113.959526,  
      "y": 179.32193  
    },  
    {  
      "x": 99.90567,  
      "y": 178.14986  
    },  
    {  
      "x": 88.95302,  
      "y": 172.11215  
    }  
  ]  
}
```

```
    },  
    {  
      "x": 95.30528,  
      "y": 168.83522  
    },  
    {  
      "x": 112.094025,  
      "y": 170.73698  
    },  
    {  
      "x": 129.00156,  
      "y": 166.50134  
    },  
    {  
      "x": 128.46986,  
      "y": 156.3436  
    },  
    {  
      "x": 111.70824,  
      "y": 158.2964  
    },  
    {  
      "x": 96.44703,  
      "y": 158.49106  
    },  
    {  
      "x": 111.8587,  
      "y": 164.61029  
    }  
  ]  
},  
"nose contour": {  
  "point": [  
    {  
      "x": 100.12734,  
      "y": 103.06376  
    },  
    {  
      "x": 99.64418,  
      "y": 121.0322  
    },  
    {  
      "x": 92.39153,  
      "y": 134.87288  
    },  
    {  
      "x": 95.70406,  
      "y": 140.70796  
    },  
    {  
      "x": 113.04284,  
      "y": 143.04053  
    },  
    {  
      "x": 129.63658,  
      "y": 138.64207  
    }  
  ]  
}
```



```
    },
    {
      "x": 131.6913,
      "y": 131.54257
    },
    {
      "x": 122.73644,
      "y": 119.520386
    },
    {
      "x": 119.106,
      "y": 102.450005
    },
    {
      "x": 102.1847,
      "y": 138.2676
    },
    {
      "x": 122.21693,
      "y": 137.64107
    },
    {
      "x": 112.65759,
      "y": 131.8112
    }
  ]
}
}
}
]
```

- 失败响应样例

```
{
  "error code": "FRS.0019",
  "error msg": "The service has not been subscribed."
}
```

- 参数说明

名称	参数类型	说明
faces	List [DetectFace]	检测到的人脸，DetectFace 结构见 DetectFace 。 调用失败无此字段。
error_code	String	调用失败时的错误码，具体请参考 错误码说明 。 调用成功时无此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

返回值

- 正常
200
- 失败

返回值	说明
400	<ul style="list-style-type: none">• 语义有误，当前请求无法被服务器解析。除非进行修改，否则客户端不应重复提交这个请求。• 请求参数有误。
401	当前请求需要用户验证。
403	没有操作权限。
404	请求失败，在服务器上未找到请求所希望得到的资源。
500	服务器遇到了一个未曾预料的状态，导致无法完成对请求的处理。

2.2.2 人脸比对

功能介绍

将两个人脸进行比对，来判断是否为同一个人，返回比对置信度。

如果传入的图片中包含多个人脸，选取最大的人脸进行比对。

URI

- URI 格式

```
POST /v1/{project_id}/face-compare
```

- 参数说明

名称	是否必选	参数类型	说明
project_id	是	String	项目编号

请求

- 请求样例（方式一：使用图片的 BASE64 编码）

```
POST https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-compare
```

Request Header:

Content-Type: application/json

X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQEzDT...

Request Body:

```
{
  "image1_base64": "/9j/4AAQSkZJRgABAgEASABIAAD",
  "image2_base64": "/9j/4AAQSkZJRgABAgEASABIAAD"
}
```

- 请求样例（方式二：使用图片文件）

```
POST https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-compare
```

Request Header:

X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQEzDT...

Request Body:

image1_file: File(图片文件)

image2_file: File(图片文件)

- 请求样例（方式三：使用图片 url）

```
POST https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-compare
```

Request Header:

Content-Type: application/json

X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQEzDT...

Request Body:

```
{
  "image1_url": "/BucketName/ObjectName",
  "image2_url": "/BucketName/ObjectName"
}
```

- 参数说明

名称	是否必选	参数类型	说明
image1_url	与 image1_file、image1_base64 三选一	String	图片的 URL 路径，目前仅支持华为云上 OBS 提供的临时授权。
image1_file	与 image1_url、image1_base64 三选一	File	本地图片文件。上传文件时，请求格式为 multipart。图片不超过 12M。
image1_base64	与 image1_file、image1_url 三选一	String	图像数据，base64 编码，要求： <ul style="list-style-type: none"> base64 编码后大小不超过 12M。 支持 JPG/JPEG/BMP/PNG 格式。

名称	是否必选	参数类型	说明
image2_url	与 image2_file、 image2_base64 三选一	String	图片的 URL 路径，目前仅支持华为云上 OBS 提供的临时授权。
image2_file	与 image2_url、 image2_base64 三选一	File	本地图片文件。上传文件时，请求格式为 multipart。图片不超过 12M。
image2_base64	与 image2_file、 image2_url 三 选一	String	图像数据，base64 编码，要求： <ul style="list-style-type: none"> base64 编码后大小不超过 12M。 支持 JPG/JPEG/BMP/PNG 格式。

响应

● 响应样例

```
{
  "image1 face": {
    "bounding box": {
      "width": 174,
      "top left y": 37,
      "top left x": 22,
      "height": 174
    }
  },
  "similarity": 0.4078676104545593,
  "image2 face": {
    "bounding box": {
      "width": 118,
      "top left y": 28,
      "top left x": 94,
      "height": 118
    }
  }
}
```

● 失败响应样例

```
{
  "error code": "FRS.0501",
  "error msg": "Detect no face, check out your picture."
}
```

● 参数说明

参数类型	说明	
image1_face	DetectFace	第 1 幅图像中检测到的人脸。 DetectFace 结构见 DetectFace 。 调用失败时无此字段。
image2_face	DetectFace	第 2 幅图像中检测到的人脸。 调用失败时无此字段。
similarity	Double	人脸相似度，1 表示最大，0 表示最小，值越大表示越相似。一般情况下超过 0.93 即可认为是同一个人。 调用失败时无此字段。
error_code	String	调用失败时的错误码，具体请参考 错误码说明 。 调用成功时无此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

返回值

- 正常
200
- 失败

返回值	说明
400	<ul style="list-style-type: none">• 语义有误，当前请求无法被服务器解析。除非进行修改，否则客户端不应重复提交这个请求。• 请求参数有误。
401	当前请求需要用户验证。
403	没有操作权限。
404	请求失败，在服务器上未找到请求所希望得到的资源。
500	服务器遇到了一个未曾预料的状态，导致无法完成对请求的处理。



2.2.3 人脸库资源管理 API

创建人脸库

功能介绍

创建用于存储人脸特征的人脸库。

您最多可以创建10个人脸库，每个人脸库最大容量为100000。如有更大规格的需求请联系客服。

URI

- URI 格式

```
POST /v1/{project_id}/face-sets
```

- 参数说明

名称	是否必选	参数类型	说明
project_id	是	String	项目编号

请求

- 请求样例

```
POST https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-sets
```

Request Header:

Content-Type: application/json

X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQEExDT...

Request Body:

```
{
  "face set name": "test",
  "face set capacity": 100000,
  "external fields" : {
    "timestamp" : {
      "type" : "long"
    },
    "id" : {
      "type" : "string"
    },
    "number" : {
      "type" : "integer"
    }
  }
}
```

- 参数说明

名称	是否必选	参数类型	说明
face_set_name	是	String	人脸库名称。
face_set_capacity	否	Integer	人脸库最大的容量，填写 1 万整数倍的数字，例如：30000。默认为 100000，最大值为 100000。
external_fields	否	Json	用户自定义数据，自定义字段不能以系统保留字段 vector、bounding_box、external_image_id、face_id、create_time、_id、_all、_source 等字段命名，json 字符串不校验重复性，自定义字段的 key 值长度范围为[1,36]，string 类型的 value 长度范围为 [1,256]，参考 自定义字段 。

响应

- 响应样例

```
{
  "face set info": {
    "face number": 0,
    "face set id": "WYYOFIGb",
    "face set name": "test10",
    "create date": "2018-05-28 02:19:00",
    "face set capacity": 10000,
    "external fields": {
      "number": {
        "type": "integer"
      },
      "id": {
        "type": "string"
      },
      "timestamp": {
        "type": "long"
      }
    }
  }
}
```

- 失败响应样例

```
{
  "error code": "FRS.0002",
  "error msg": "The authentication token is abnormal."
}
```

- 参数说明

名称	参数类型	说明
face_set_info	FaceSetInfo	人脸库信息，详见 FaceSetInfo 。 调用失败时无此字段。
error_code	String	调用失败时的错误码，具体请 错误码说明 。 调用成功时无此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

返回值

- 正常
200
- 失败

返回值	说明
400	<ul style="list-style-type: none"> 语义有误，当前请求无法被服务器解析。除非进行修改，否则客户端不应重复提交这个请求。 请求参数有误。
401	当前请求需要用户验证。
403	没有操作权限。
404	请求失败，在服务器上未找到请求所希望得到的资源。
500	服务器遇到了一个未曾预料的状态，导致无法完成对请求的处理。

查询所有人脸库

功能介绍

查询当前用户所有人脸库的状态信息。

URI

- URI 格式

```
GET /v1/{project_id}/face-sets
```

- 参数说明

名称	是否必选	参数类型	说明
project_id	是	String	项目编号



请求

请求样例

```
Get https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-sets

Request Header:
Content-Type: application/json
X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQEExDT...
```

响应

- 响应样例

```
{
  "face sets info": [
    {
      "face number": 0,
      "face set id": "ylXMMZTO",
      "face set name": "test",
      "create date": "2018-05-11 07:49:40",
      "face set capacity": 10000,
      "external fields": {
        "number": {
          "type": "integer"
        },
        "id": {
          "type": "string"
        },
        "timestamp": {
          "type": "long"
        }
      }
    }
  ]
}
```

- 失败响应样例

```
{
  "error code": "FRS.0002",
  "error msg": "The authentication token is abnormal."
}
```

- 参数说明

名称	参数类型	说明
face_sets_info	List [FaceSetInfo]	人脸库信息集合，详见 FaceSetInfo 。 调用失败时无此字段。
error_code	String	调用失败时的错误码，具体请参考 错误码说明 。 调用成功时无此字段。

名称	参数类型	说明
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

返回值

- 正常
200
- 失败

返回值	说明
400	<ul style="list-style-type: none"> 语义有误，当前请求无法被服务器解析。除非进行修改，否则客户端不应重复提交这个请求。 请求参数有误。
401	当前请求需要用户验证。
403	没有操作权限。
404	请求失败，在服务器上未找到请求所希望得到的资源。
500	服务器遇到了一个未曾预料的状态，导致无法完成对请求的处理。

删除人脸库

功能介绍

删除人脸库以及其中所有的人脸。

URI

- URI 格式

```
DELETE /v1/{project_id}/face-sets/{face_set_name}
```

- 参数说明

名称	是否必选	参数类型	说明
project_id	是	String	项目编号
face_set_name	是	String	人脸库名称

请求

请求样例

```
DELETE https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-sets/showFaceSet

Request Header:
Content-Type: application/json
X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQEzDT...
```

响应

- 响应样例

```
{
  "face set name": "test1"
}
```

- 失败响应样例

```
{
  "error_code": "FRS.0002",
  "error_msg": "The authentication token is abnormal."
}
```

- 参数说明

名称	参数类型	说明
face_set_name	String	人脸库名称。 调用失败时无此字段
error_code	String	调用失败时的错误码，具体请参考 错误码说明 。 调用成功时无此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

返回值

- 正常
200
- 失败

返回值	说明
400	<ul style="list-style-type: none">● 语义有误，当前请求无法被服务器解析。除非进行修改，否则客户端不应重复提交这个请求。● 请求参数有误。
401	当前请求需要用户验证。
403	没有操作权限。

返回值	说明
404	请求失败，在服务器上未找到请求所希望得到的资源。
500	服务器遇到了一个未曾预料的状态，导致无法完成对请求的处理。

2.2.4 人脸资源管理

添加人脸

功能介绍

添加图片到人脸库中，检测到多少张人脸，则增加多少张人脸到人脸库当中。

URI

- URI 格式

```
POST /v1/{project_id}/face-sets/{face_set_name}/faces
```

- 参数说明

名称	是否必选	参数类型	说明
project_id	是	String	项目编号
face_set_name	是	String	人脸库名称

请求

- 请求样例（方式一：使用图片的 BASE64 编码）

```
POST https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-sets/showFaceSet/faces
```

Request Header:

Content-Type: application/json

X-Auth-Token: MIINRWYJKoZIhvcNAQcCoIINODCCDTQCAQEExDT...

Request Body:

```
{
  "image base64": "/9j/4AAQSkZJRgABAgEASABIAAD",
  "external image id": "imageID",
  "external fields": {
    "timestamp": 12,
    "id": "home"
  }
}
```

- 请求样例（方式二：使用图片文件）

```
POST https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-sets/showFaceSet/faces
```

Request Header:

X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQEExDT...

Request Body:

image file: File(图片文件)

external image id: imageID

external_fields: {"timestamp" : 12, "id" : "home"}

- 请求样例（方式三：使用图片 url）

```
POST https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-sets/showFaceSet/faces
```

Request Header:

Content-Type: application/json

X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQEExDT...

Request Body:

```
{
  "image url": "/BucketName/ObjectName",
  "external image id": "imageID",
  "external fields" : {
    "timestamp": 12,
    "id": "home"
  }
}
```

- 参数说明

名称	是否必选	参数类型	说明
image_url	与 image_file、image_base64 三选一	String	图片的 URL 路径，目前仅支持华为云上 OBS 提供的临时授权。
image_file	与 image_url、image_base64 三选一	File	本地图片文件。上传文件时，请求格式为 multipart。图片不超过 12M。
image_base64	与 image_file、image_url 三选一	String	图像数据，base64 编码，要求： <ul style="list-style-type: none">base64 编码后大小不超过 12M。支持 JPG/JPEG/BMP/PNG 格式。

名称	是否必选	参数类型	说明
external_image_id	否	String	用户指定的图片外部ID，与当前图像绑定。用户没提供，系统会生成一个。 该ID长度范围为1~36位，可以包含字母、数字、中划线或者下划线，不包含其他的特殊字符。
external_fields	否	Json	根据用户自定义数据类型，填入相应的数值。 创建 faceset 时定义该字段，Json 字符串不校验重复性，参考 自定义字段 。

响应

- 响应样例

```
{
  "face set id": "T785tx1N",
  "face set name": "test1",
  "faces": [
    {
      "bounding box": {
        "width": 63,
        "top left y": 100,
        "top left x": 221,
        "height": 63
      },
      "external image id": "Xr0phyap",
      "external fields" : {
        "timestamp": 12,
        "id": "home"
      },
      "face id": "JLa9hYL1"
    }
  ]
}
```

- 失败响应样例

```
{
  "error code": "FRS.0404",
  "error msg": "Detect no face, can not add it to face set."
}
```

- 参数说明

名称	参数类型	说明
face_set_name	String	人脸库名称。 调用失败时无此字段。
face_set_id	String	人脸库 ID。 调用失败时无此字段。
faces	List [FaceSetFace]	人脸库当中的人脸结构，详见 FaceSetFace 。 调用失败时无此字段。
error_code	String	调用失败时的错误码，具体请参考 错误码说明 。 调用成功时无此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

返回值

- 正常
200
- 失败

返回值	说明
400	<ul style="list-style-type: none"> 语义有误，当前请求无法被服务器解析。除非进行修改，否则客户端不应重复提交这个请求。 请求参数有误。
401	当前请求需要用户验证。
403	没有操作权限。
404	请求失败，在服务器上未找到请求所希望得到的资源。
500	服务器遇到了一个未曾预料的状态，导致无法完成对请求的处理。

2.2.5 人脸搜索

功能介绍

人脸搜索是指在已有的人脸库中，查询与目标人脸相似的一张或者多张人脸，并返回相应的置信度。

支持传入图片或者faceID进行人脸搜索，如果传入的是多张人脸图片，选取图片中检测到的最大尺寸人脸作为检索的输入。

URI

- URI 格式

```
POST /v1/{project_id}/face-sets/{face_set_name}/search
```

- 参数说明

名称	是否必选	参数类型	说明
project_id	是	String	项目编号
face_set_name	是	String	人脸库名称

请求

- 请求样例（方式一：使用图片的 BASE64 编码）

```
POST https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-sets/showFaceSet/search
```

Request Header:

Content-Type: application/json

X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQEExDT...

Request Body:

```
{
  "image base64": "/9j/4AAQSkZJRgABAgEASABIAAD",
  "sort" : [
    {
      "timestamp" : "asc"
    }
  ],
  "return fields" : ["timestamp", "id"],
  "filter" : "timestamp:12"
}
```

- 请求样例（方式二：使用图片文件）

```
POST https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-sets/showFaceSet/search
```

Request Header:

X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQEExDT...

Request Body:

```
image_file: File(图片文件)
return fields: ["timestamp", "id"]
filter: timestamp:12
```

- 请求样例（方式三：使用图片 url）


```
POST https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-sets/showFaceSet/search

Request Header:
Content-Type: application/json
X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQEExDT...

Request Body:
{
  "image_url": "/BucketName/ObjectName",
  "sort" : [
    {
      "timestamp" : "asc"
    }
  ],
  "return_fields" : ["timestamp", "id"],
  "filter" : "timestamp:12"
}
```

- 请求样例（方式四：使用人脸编号）

```
POST https://face.cn-north-1.myhuaweicloud.com/v1/d3f8e6476fbf416689f209e168ef4d31/face-sets/showFaceSet/search

Request Header:
Content-Type: application/json
X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQEExDT...

Request Body:
{
  "face_id": "6KLB1Ktu",
  "sort" : [
    {
      "timestamp" : "asc"
    }
  ],
  "return_fields" : ["timestamp", "id"],
  "filter" : "timestamp:12"
}
```

- 参数说明

名称	是否必选	参数类型	说明
image_url	与 image_file、image_base64、face_id 四选一	String	图片的 URL 路径，目前仅支持华为云上 OBS 提供的临时授权。
image_file	与 image_url、image_base64、face_id 四选一	File	本地图片文件。上传文件时，请求格式为 multipart。图片不超过 12M。

名称	是否必选	参数类型	说明
image_base64	与 image_file、image_url、face_id 四选一	String	图像数据，base64 编码，要求： <ul style="list-style-type: none"> base64 编码后大小不超过 12M。 支持 JPG/JPEG/BMP/PNG 格式。
face_id	与 image_file、image_base64、image_url 四选一	String	导入人脸时，系统返回的人脸编号。
top_n	否	Integer	返回查询到的最相似的 N 张人脸，N 默认为 10。
threshold	否	Double	人脸相似度阈值，低于这个阈值则不返回，取值范围 0~1，一般情况下建议取值 0.93，默认为 0。
sort	否	JSONArray	支持字段排序，参考 sort 语法 。
filter	否	String	过滤条件，参考 filter 语法 。
return_fields	否	String	指定返回的自定义字段。

响应

- 响应样例

```
{
  "faces": [
    {
      "bounding_box": {
        "width": 170,
        "top_left_y": 37,
        "top_left_x": 20,
        "height": 170
      },
      "similarity": 0.996146,
      "external_image_id": "123",
      "external_fields": {
```

```

        "id": "home",
        "timestamp": 12
      },
      "face_id": "6KLB1Ktu"
    },
    {
      "bounding box": {
        "width": 170,
        "top left y": 37,
        "top left x": 20,
        "height": 170
      },
      "similarity": 0.996146,
      "external_image_id": "12",
      "external_fields": {
        "id": "home1",
        "timestamp": 12
      },
      "face_id": "PexOpqRj"
    }
  ]
}

```

- 失败响应样例

```

{
  "error code": "FRS.0018",
  "error msg": "The service inner error."
}

```

- 参数说明

名称	参数类型	说明
faces	List [SearchFace]	查找的人脸集合，详见 SearchFace 错误!未找到引用源。 调用失败时无此字段。
error_code	String	调用失败时的错误码，具体请参考 错误码说明 。 调用成功时无此字段。
error_msg	String	调用失败时的错误信息。 调用成功时无此字段。

返回值

- 正常
200
- 失败

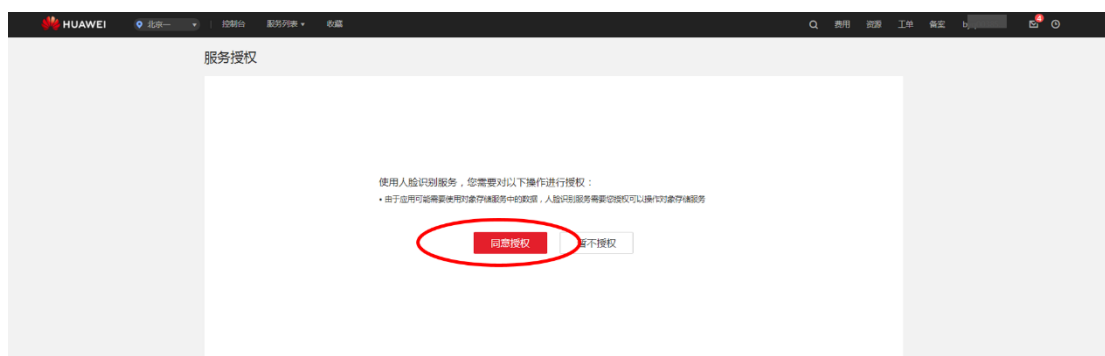
返回值	说明
400	<ul style="list-style-type: none">语义有误，当前请求无法被服务器解析。除非进行修改，否则客户端不应重复提交这个请求。请求参数有误。
401	当前请求需要用户验证。
403	没有操作权限。
404	请求失败，在服务器上未找到请求所希望得到的资源。
500	服务器遇到了一个未曾预料的状态，导致无法完成对请求的处理。

2.2.6 其他服务接口，如查询所有人脸库、查询人脸、删除人脸的 API 接口，详情参见 [《API 文档》](#) 对应章节。

2.2.7 使用图片 url 操作步骤

1、首先需要对人脸识别服务进行授权，可以操作对象存储服务（OBS），点击人脸服务控制台右上角的“服务授权”，“同意授权”即可；





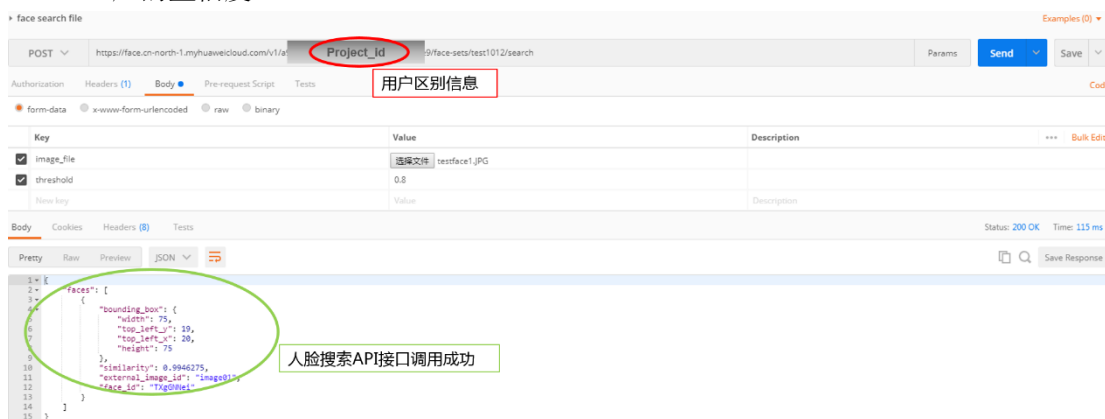
- 2、将待处理图片，上传到对象存储服务（OBS）中，具体操作步骤参考《[快速通过OBS控制台上传文件](#)》；
- 3、确认图片文件上传成功后，拼接图片url（image_url），格式为：/桶名/文件夹目录（可选）/文件名；
- 4、注意，人脸服务中只能使用用户自己的OBS中上传的图片。

3 打卡任务

3.1 人脸搜索

使用人脸搜索API，需要先创建人脸库，在已有的人脸库中查询与目标人脸相似的一张或多张人脸，并得到相应的置信度。可参考如下步骤完成：

- 1、创建自己的人脸集，具体操作参看2.2.3节；
- 2、向人脸库中添加人脸图片1、图片2，具体操作参看2.2.4节；
- 3、上传图片3，在第一步创建的人脸库中查询与目标人脸相似的一张人脸，并得到响应的置信度。



3.2 删除人脸库并查询所有人脸库

- 1、使用删除人脸库API，删除人脸库及其中所有的人脸，得到删除成功的响应信息；
- 2、成功删除当前用户所有人脸库后，调用查询所有人脸库API接口，查看当前用户所有人脸库的状态信息，最终显示人脸库信息为空。



show all face set

GET https://face.cn-north-1.myhuaweicloud.com/v1/a/Project_id/face-sets

Authorization Headers (2) Body Pre-request Script Tests

KEY	VALUE	DESCRIPTION
Content-Type	application/json	
X-Auth-Token	MiITCAVjKzIhvdNAQcColIS+TCCeVUcAQExDTALBgghg8ZQMEAgEwghFWBgghkiG9w...	

Key Value Description

Body Cookies Headers (7) Test Results

Status: 200 OK Time: 231 ms Size: 279 B

Pretty Raw Preview JSON

```
1 {
2   "face_sets_info": []
3 }
```

当前用户中人脸已全部删除