Optical Character Recognition

API Reference

Issue	01
Date	2020-12-25

Huawei Technologies Co., Ltd.

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

Contents

1 Before You Start

1.1 Overview

OCR provides services through open Application Programming Interfaces (APIs). You can obtain the inference result by accessing and calling APIs in real time. It helps you collect key data automatically and build an intelligent business system, thereby improving service efficiency.

Before calling an OCR API, ensure that you are familiar with OCR concepts.

1.2 API Calling

OCR provides Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS. For details about API calling, see **Calling APIs**.

1.3 Endpoints

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

Table 1-1 OCR endpoints

| Region Name	Endpoint Region	Endpoint	Service
UAE - Abu Dhabi	ae-ad-1	ocr.ae-ad-1.g42cloud.com	Passport OCR, UAE ID Card OCR, and UAE Bank Card OCR

1.4 Constraints

For details, see the API description and Constraints.

1.5 Concepts

- Account

  The account has full access permissions for all of its cloud services and resources. It can be used to reset user passwords and grant user permissions. For security purposes, do not directly use the account to perform routine management. Instead, create IAM users and grant them permissions for routine management.

- User

  A user is created using an account, for access to cloud services. Each user has their own identity credentials (password and access keys).

  An IAM user can view the account ID and user ID on the My Credentials page of the console. The account name, username, and password will be required for API authentication.

- Region

  Regions are divided based on geographical location and network latency. Public services, such as Elastic Cloud Server (ECS), Elastic Volume Service (EVS), Object Storage Service (OBS), Virtual Private Cloud (VPC), Elastic IP (EIP), and Image Management Service (IMS), are accessible within the same region. Regions are classified as universal regions and dedicated regions. A universal region provides universal cloud services for common tenants. A dedicated region provides specific services for specific tenants.

- AZ

  An AZ contains one or more physical data centers. It has independent cooling, fire extinguishing, moisture-proof, and electricity facilities. Within an AZ, computing, network, storage, and other resources are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to allow you to build cross-AZ high-availability systems.

- Project

  Projects isolate resources (including compute, storage, and network resources) across physical regions. A default project is provided for each cloud service region, and subprojects can be created under each default project. Users can be granted permissions to access all resources in a specific project. For more refined access control, create subprojects under a project and create resources in the subprojects. Users can then be assigned permissions to access specific resources in the subprojects.

Figure 1-1 Project isolation model
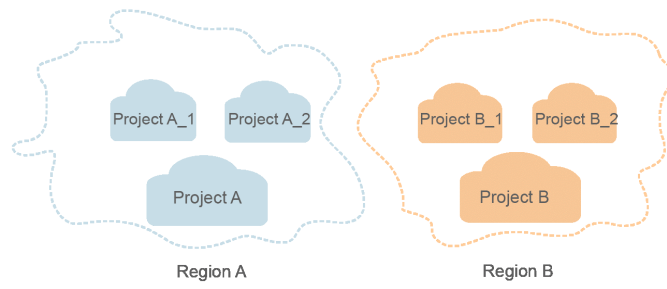
2 Calling APIs

2.1 Subscribing to an OCR Service

1. On the **Overview** page, locate a service you want to use and click **Subscribe** in the **Operation** column.

2. After the service is successfully subscribed to, check that the status is **Subscribed**.

2.2 Making an API Request

Make a RESTful API request and call the API.

Request URI

A request URI is in the following format:

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

Although a request URI is included in the request header, most programming languages and frameworks require the request URI to be transmitted separately.

Table 2-1 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. The endpoint varies depending on services in different regions. For more details, see **Endpoints**.
resource-path	Path in which the requested resource is located, that is, the API access path.

Parameter	Description
query-string	Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of *Parameter name=Parameter value*. For example, **limit=10** indicates that up to 10 data records will be displayed.

```
# UAE ID Card OCR
https://ocr.ae-ad-1.g42cloud.com/v2/{project_id}/ocr/emirates-id-card
```

📖 NOTE

To simplify the URI display in this document, each API is provided only with a **resource-path** and a request method. The **URI-scheme** of all APIs is **HTTPS**, and the endpoints of all APIs in the same region are identical.

Request Methods

The HTTP protocol defines the following request methods that can be used to send a request to the server:

Table 2-2 HTTP-defined request methods

Method	Description
GET	Requests the server to return specified resources.
PUT	Requests the server to update specified resources.
POST	Requests the server to add resources or perform special operations.
DELETE	Requests the server to delete specified resources, for example, an object.
HEAD	Same as GET except that the server must return only the response header.
PATCH	Requests the server to update partial content of a specified resource. If the resource does not exist, a new resource will be created.

```
# UAE ID Card OCR
POST https://ocr.ae-ad-1.g42cloud.com/v2/{project_id}/ocr/emirates-id-card
```

Request Header

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

- **Content-Type**: request body type or format. This field is mandatory and its default value is **application/json**. Other values of this field will be provided for specific APIs if any.

- **X-Auth-Token**: user token. This field is required only for token-based API authentication. For details, see **Authentication**.

The following provides an example request with a request header included.

```
# UAE ID Card OCR
POST https://ocr.ae-ad-1.g42cloud.com/v2/{project_id}/ocr/emirates-id-card
Content-Type: application/json
x-auth-token: MIIaBgYJKoZIhvcNAQcC...
```

Request Body

The body of a request is often sent in a structured format as specified in the **Content-Type** header field. The request body transfers all content except the request header.

The request body varies depending on APIs. Some APIs do not require the request body, for example, the APIs requested using the GET and DELETE methods.

The request parameters and parameter description can be obtained from the API request. The following provides an example request with a body included.

```
# UAE ID Card OCR
POST https://ocr.ae-ad-1.g42cloud.com/v2/{project_id}/ocr/emirates-id-card
Request Header:
Content-Type: application/json
X-Auth-Token: MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAgABwESAAMAA..."
}
```

If all data required for the API request is available, you can send the request to call the API through curl, Postman, or coding.

2.3 Authentication

Requests for calling an API can be authenticated using either of the following methods:

- Token-based authentication: Requests are authenticated using a token.

  Call APIs using **endpoints** and tokens. Object Storage Service (OBS), Data Ingestion Service (DIS), Natural Language Processing (NLP), Optical Character Recognition (OCR), and Image Recognition in the **UAE-Abu Dhabi -OP5** region cannot be called.

- AK/SK-based authentication: Requests are authenticated by encrypting the request body using an AK/SK pair.

> **⚠ CAUTION**
>
> To call Object Storage Service (OBS), Data Ingestion Service (DIS), Natural Language Processing (NLP), Optical Character Recognition (OCR), or Image Recognition in the **UAE-Abu Dhabi** region, you need to create a user and generate an AK/SK. Then, call the corresponding API through AK/SK authentication and endpoint. For details about endpoints, visit **https://docs.g42cloud.com/en-us/endpoint/index.html**. For details about how to obtain the AK/SK for the **UAE-Abu Dhabi** region, see **How Do I Obtain an Access Key (AK/SK) in the ME-Abu Dhabi-OP5 Region?**.

Token-based Authentication

A token specifies temporary permissions in a computer system. During API authentication using a token, the token is added to a request to get permissions for calling the API.

> **📖 NOTE**
>
> - A token is valid for 24 hours. When using a token for authentication, cache it to avoid frequent calling.

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

username indicates the IAM username, **domainname** indicates the account name, ************ indicates the login password, and **project name** indicates the project name.

```
POST https://iam.ae-ad-1.g42cloud.com/v3/auth/tokens
Content-Type: application/json
{
    "auth": {
        "identity": {
            "methods": [
                "password"
            ],
            "password": {
                "user": {
                    "name": "username", //IAM username
                    "password": " ********", //Password
                    "domain": {
                        "name": "domainname" //Account name
                    }
                }
            }
        },
        "scope": {
            "project": {
                "name": "project name" //Replace project name with the actual project name.
            }
        }
    }
}
```

As shown in the following figure, **x-subject-token** in the response header is the desired user token. This token can then be used to authenticate the calling of OCR APIs.

Figure 2-1 Response header for obtaining a user token

AK/SK-based Authentication

📖 **NOTE**

AK/SK-based authentication supports API requests with a body less than or equal to 12 MB. For API requests with a larger body, perform token-based authentication.

In AK/SK-based authentication, AK/SK is used to sign requests and the signature is then added to the requests for authentication.

- AK: access key ID, which is a unique identifier used in conjunction with a secret access key to sign requests cryptographically.
- SK: secret access key used in conjunction with an AK to sign requests cryptographically. It identifies a request sender and prevents the request from being modified.

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

2.4 Response

Status Code

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

A status code consists of multiple digits, which indicates the response status of a request. If the $2xx$ status code is returned, the API is called successfully. For details about the status code list, see **Status Codes**.

Response Header

Similar to a request, a response also has a header, for example, **Content-Type**. **Table 2-3** lists the fields in the response header.

Table 2-3 Response Header

Field	Description
Content-Length	Length of the response body. The unit is byte.
Date	Time when a request response is returned
Content-Type	MIME type of the response body

Response Body

The body of a response is returned in structured format as specified in the **Content-Type** header field. The response body transfers all content except the response header.

For UAE ID Card OCR, the response body is as follows:

```
{
    "result": {
        "side": "front",
        "id_number": "784-XXXX-XXXXXXX-X",
        "name": "XXX",
        "name_arabic": "XXX",
        "nationality": "Egypt",
        "nationality_arabic": "مصر",
        "confidence": {
            "id_number": 0.9712,
            "name": 0.9731,
            "name_arabic": 0.9234,
            "nationality": 0.9543,
            "nationality_arabic": 0.9543,
        }
    }
}
```

If an error occurs during API calling, an error code and a message will be displayed. The following shows an error response body.

```
{
    "error_code": "AIS.0103",
    "error_msg": "The image size does not meet the requirements."
}
```

In the response body, **error_code** indicates the error code, and **error_msg** provides information about the error.

3 APIs

3.1 Passport

Function

This API identifies and extracts text from the first page of images of passports and converts the text into a structured JSON format.

Prerequisites

Before using this API, you need to apply for the service and complete authentication. For details, see **Subscribing to an OCR Service** and **Authentication**.

📖 **NOTE**

Before using the service for the first time, you need to enable the service by clicking . You only need to subscribe to the service once. If the service is not enabled, an error message with error code "ModelArts.4204" will be displayed when you call the service. Before calling the service, log in to the OCR console and enable the service. Ensure that the region where the service is enabled is the same as that where the service is called.

URI

POST https://{endpoint}/v2/{project_id}/ocr/passport

Table 3-1 Path parameters

Parameter	Mandatory	Description
endpoint	Yes	Domain name or IP address of the server bearing the REST service endpoint. The endpoint varies depending on services in different regions. For more details, see **Endpoints**.

Parameter	Mandatory	Description
project_id	Yes	Project ID, which can be obtained from **Obtaining an Account ID and Project ID**.

Request Parameters

Table 3-2 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token Used to obtain the permission to operate APIs. The token is the value of **X-Subject-Token** in the response header in **Authentication**.
Content-Type	Yes	String	MIME type of the request body. The value is **application/json**.

Table 3-3 Request body parameters

Parameter	Mandatory	Type	Description
image	No	String	Configure either this parameter or **url**. Base64-encoded image file. The image file has a size limit of 10 MB. No side of the image can be smaller than 15 or larger than 4,096 pixels. Only images in JPEG, JPG, PNG, BMP, or TIFF format can be recognized. An example is **/9j/ 4AAQSkZJRgABAg....** If the image data contains an unnecessary prefix, the error "The image format is not supported" is reported.

Parameter	Mandatory	Type	Description
url	No	String	Configure either this parameter or **image**. Image URL. Currently, the following URLs are supported: • URL of an image. **NOTE** • The API response time depends on the image download time. If the image download takes a long time, the API call will fail. • Ensure that the storage service where the images to be detected reside is stable and reliable. OBS is recommended for storing image data.

Response Parameters

📖 NOTE

The status code may vary depending on the recognition results. For example, **200** indicates that the API is successfully called, and **400** indicates that the API fails to be called. The following describes the status codes and corresponding response parameters.

Status code: 200

Table 3-4 Response body parameter

Parameter	Type	Description
result	**PassportResult** object	Calling result This parameter is not returned when the API fails to be called. This parameter consists of the following three parts: 13 key fields, expressed in English; **extra_info**, expressed in local official language; and confidence of key fields. A higher confidence indicates a more accurate result.

Table 3-5 PassportResult

Parameter	Type	Description
country_code	String	Country code
passport_number	String	Passport ID

Parameter	Type	Description
surname	String	Last name
given_name	String	Given game
sex	String	Gender
date_of_birth	String	Date of birth, for example, **1990-12-12**
date_of_expiry	String	Passport date of expiry, for example, **2020-07-08**
confidence	Object	Confidence of a field. The value ranges from 0 to 1. A higher confidence indicates a higher accuracy of the field identified. The confidence is calculated using algorithms and is not equal to the accuracy.
extra_info	Object	By default, this parameter is left blank.

Status code: 400

Table 3-6 Response body parameters

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

Request Example

- Request example (Method 1: Use the image Base64 string.)

```
POST https://ocr.ae-ad-1.g42cloud.com/v2/{project_id}/ocr/passport
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAA...",
}
```

- Request example (Method 2: Use the image URL.)

```
POST https://ocr.ae-ad-1.g42cloud.com/v2/{project_id}/ocr/passport
```

```
Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
Request Body:
{
    "url":"https://BucketName.obs.xxxx.com/ObjectName",
}
```

Example Response

Status code: 200

```
{
  "result": {
    "country_code": "ETF",
    "surname": "HUZHAO",
    "given_name": "ZHAOMIN DESALEGN ",
    "passport_number": "EP435XXXX",
    "date_of_birth": "1985-09-18",
    "sex": "M",
    "date_of_expiry": "2022-01-15",
    "machine_code": "P<ETFHUZHAO<< ZHAOMIN <DESALEGN<<<<<<<<<<<<<<<",
    "machine_code2": "EP435XXXX7ETF8509185M2201155<<<<<<<<<<<<<<08",
    "extra_info": {},
    "confidence": {
      "country_code": 0.9727,
      "surname": 0.9727,
      "given_name": 0.9727,
      "passport_number": 0.9558,
      "date_of_birth": 0.9558,
      "sex": 0.9558,
      "date_of_expiry": 0.9558
    }
  }
}
```

Status code: 400

Failure response example

```
{
  "error_code": "AIS.0103",
  "error_msg": "The image size does not meet the requirements."
}
```

Status Codes

Status Code	Description
200	Success response
400	Failure response

For details about status codes, see **Status Codes**.

Error Codes

For details about error codes, see **Error Codes**.

3.2 UAE ID Card OCR

Function

This API identifies and extracts text from images of UAE ID cards and converts the text into JSON format.

Figure 3-1 UAE ID Card example

Prerequisites

Before using UAE ID Card OCR, you need to apply for the service and complete authentication. For details, see **Subscribing to an OCR Service** and **Authentication**.

☐ **NOTE**

Before using the service for the first time, you need to enable the service by clicking **Subscribe**. You only need to subscribe to the service once. If the service is not enabled, an error message with error code "ModelArts.4204" will be displayed when you call the service. Before calling the service, log in to the OCR console and enable the service. Ensure that the region where the service is enabled is the same as that where the service is called.

URI

POST https://{endpoint}/v2/{project_id}/ocr/emirates-id-card

Table 3-7 Path parameters

Parameter	Mandatory	Description
endpoint	Yes	Domain name or IP address of the server bearing the REST service endpoint. The endpoint varies depending on services in different regions. For more details, see **Endpoints**.
project_id	Yes	Project ID, which can be obtained from **Obtaining an Account ID and Project ID**.

Request Parameters

Table 3-8 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token Used to obtain the permission to operate APIs. The token is the value of **X-Subject-Token** in the response header in **Authentication**.
Content-Type	Yes	String	MIME type of the request body. The value is **application/json**.

Table 3-9 Request body parameters

Parameter	Mandatory	Type	Description
image	No. Set either this parameter or url.	String	Base64-encoded image file. The image file has a size limit of 10 MB. No side of the image can be smaller than 15 or larger than 8,192 pixels. Only images in JPEG, JPG, PNG, BMP, or TIFF format can be recognized. An example is **/9j/4AAQSkZJRgABAg....** If the image data contains an unnecessary prefix, the error "The image format is not supported" is reported.
url	No. Set either this parameter or image.	String	URL of the image file NOTE ● The API response time depends on the image download time. If the image download takes a long time, the API call will fail. ● Ensure that the storage service where the image to be detected resides is stable and reliable. OBS is recommended for storing image data.
side	No	String	● **front**: front of an ID card ● **back**: back of an ID card If this parameter is left empty or not included, the system automatically recognizes whether the image is the front or back of an ID card. To ensure accuracy, set this parameter.

Response Parameters

Response parameters and status codes vary in different recognition results. They are described as below.

Status code: 200

Table 3-10 Response parameters

Parameter	Type	Description
result	Object	Calling result This parameter is not returned when the API fails to be called.

Table 3-11 Response parameters

Parameter	Type	Description
side	String	**front**: front of an ID card **back**: back of an ID card
id_number	String	ID number
name_arabic	String	Name in Arabic
name	String	Name
nationality_arabic	String	Arabic nationality
nationality	String	Nationality
sex	String	Gender
date_of_birth	String	Date of birth
expiry_date	String	Date of expiry of the ID card
card_number	String	ID number
country_code	String	Code of the country where an ID card is issued
machinecode1	String	Machine code in the first line on the back of an ID card
machinecode2	String	Machine code in the second line on the back of an ID card
machinecode3	String	Machine code in the third line on the back of an ID card

Parameter	Type	Description
confidence	Object	Confidence of a field. The value ranges from 0 to 1. A higher confidence indicates a higher accuracy of the field identified. The confidence is calculated using algorithms and is not equal to the accuracy.

Status code: 400

Table 3-12 Response body parameters

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

Request Example

- Request example (Method 1: Use the image Base64 string.)
  ```
  POST  https://ocr.ae-ad-1.g42cloud.com/v2/{project_id}/ocr/emirates-id-card

  Request Header:
  Content-Type: application/json
  X-Auth-Token:
  MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
  Request Body:
  {
      "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAgABwESAAMAA..."
  }
  ```

- Request example (Method 2: Use the image URL.)
  ```
  POST  https://ocr.ae-ad-1.g42cloud.com/v2/{project_id}/ocr/emirates-id-card

  Request Header:
  Content-Type: application/json
  X-Auth-Token:
  MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
  Request Body:
  {
      "url":"https://BucketName.obs.xxxx.com/ObjectName"
  }
  ```

Example Response

Status code: 200

Successful response example (the front)

```
{
    "result": {
        "side": "front",
        "id_number": "784-XXXX-XXXXXXX-X",
        "name": "XXX",
        "name_arabic": "XXX",
        "nationality": "Egypt",
        "nationality_arabic": "مصر",
        "confidence": {
            "id_number": 0.9712,
            "name": 0.9731,
            "name_arabic": 0.9234,
            "nationality": 0.9543
            "nationality_arabic": 0.9543,
        }
    }
}
```

Successful response example (the back)

```
{
    "result": {
        "side": "back",
        "card_number": "XXXX-XXX-XX",
        "laser_number": "XXXX-XXXXXXX-XX",
        "confidence": {
            "id_number": 0.9999,
            "laser_number": 0.9994
        }
    }
}
```

Status code: 400

Failure response example

```
{
    "error_code": "AIS.0103",
    "error_msg": "The image size does not meet the requirements."
}
```

Status Codes

Status Code	Description
200	Success response
400	Failure response

For details about status codes, see **Status Codes**.

Error Codes

For details about error codes, see **Error Codes**.

3.3 UAE Bank Card OCR

Function

This API identifies and extracts text from images of UAE bank cards and converts the text into a structured format.

Prerequisites

Before using UAE Bank Card OCR, you need to apply for the service and complete authentication. For details, see **Subscribing to an OCR Service** and **Authentication**.

📖 **NOTE**

Before using the service for the first time, you need to enable the service by clicking **Subscribe**. You only need to subscribe to the service once. If the service is not enabled, an error message with error code "ModelArts.4204" will be displayed when you call the service. Before calling the service, log in to the OCR console and enable the service. Ensure that the region where the service is enabled is the same as that where the service is called.

URI

POST https://{endpoint}/v2/{project_id}/ocr/bankcard

Table 3-13 Path parameters

Parameter	Mandatory	Description
endpoint	Yes	Domain name or IP address of the server bearing the REST service endpoint. The endpoint varies depending on services in different regions. For more details, see **Endpoints**.
project_id	Yes	Project ID, which can be obtained from **Obtaining an Account ID and Project ID**.

Request Parameters

Table 3-14 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token Used to obtain the permission to operate APIs. The token is the value of **X-Subject-Token** in the response header in **Authentication**.
Content-Type	Yes	String	MIME type of the request body. The value is **application/json**.

Table 3-15 Request body parameters

Parameter	Mandatory	Type	Description
image	No. Set either this parameter or url.	String	Base64-encoded image file. The image file has a size limit of 10 MB. No side of the image can be smaller than 15 or larger than 8,192 pixels. Only images in JPG, PNG, BMP, or TIFF format can be recognized. An example is **/9j/4AAQSkZJRgABAg....** If the image data contains an unnecessary prefix, the error "The image format is not supported" is reported.
url	No. Set either this parameter or image.	String	URL of the image file **NOTE** ● The API response time depends on the image download time. If the image download takes a long time, the API call will fail. ● Ensure that the storage service where the image to be detected resides is stable and reliable. OBS is recommended for storing image data.

Response Parameters

Response parameters and status codes vary in different recognition results. They are described as below.

Status code: 200

Table 3-16 Response parameters

Parameter	Type	Description
result	Object	Calling result This parameter is not returned when the API fails to be called.

Table 3-17 Response parameters

Parameter	Type	Description
type	String	Bank card type ● Visa ● MasterCard ● Unrecognized
card_number	String	Bank card number
issue_date	String	Date of issue
expiry_date	String	Date of expiry
card_holder	String	Cardholder name
confidence	Object	Confidence of a field. The value ranges from 0 to 1. A higher confidence indicates a higher accuracy of the field identified. The confidence is calculated using algorithms and is not equal to the accuracy.

Status code: 400

Table 3-18 Response body parameters

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

Request Example

- Request example (Method 1: Use the image Base64 string.)

  POST https://ocr.ae-ad-1.g42cloud.com/v2/{project_id}/ocr/bankcard

  Request Header:
  Content-Type: application/json
  X-Auth-Token:
  MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
  Request Body:
  {
      "image":"/9j/4AAQSkZJRgABAgEASABIAAD/4RFZRXhpZgAATU0AKgAAAAgABwESAAMA..."
  }

- Request example (Method 2: Use the image URL.)

  POST https://ocr.ae-ad-1.g42cloud.com/v2/{project_id}/ocr/ bankcard

  Request Header:
  Content-Type: application/json
  X-Auth-Token:
  MIINRwYJKoZIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
  Request Body:
  {
      "url":"https://BucketName.obs.xxxx.com/ObjectName"
  }

Example Response

Status code: 200

Successful response example

```
{
    "result": {
        "bank_name": "",
        "card_number": "6225 7687 0000 8243",
        "issue_date": "12/10",
        "expiry_date": "12/20",
        "type": "Visa",
        "card_holder": "SALIM RASHID",
        "confidence": {
            "bank_name": 0.0,
            "card_number": 0.8562,
            "issue_date": 0.9962,
            "expiry_date": 0.9256,
            "type": 0.9012,
            "card_holder": 0.8452,
        }
    }
}
```

Status code: 400

Failure response example

```
{
    "error_code": "AIS.0103",
    "error_msg": "The image size does not meet the requirements"
}
```

Status Codes

Status Code	Description
200	Success response

Status Code	Description
400	Failure response

For details about status codes, see **Status Codes**.

Error Codes

For details about error codes, see **Error Codes**.

4 Appendix

4.1 Status Codes

An HTTP status code consists of three digits, which is classified into five categories: 1xx: related information; 2xx: operation successful; 3xx: redirection; 4xx: client error; 5xx: server error.

The following table lists the common status codes.

Status Code	Coding	Description
100	Continue	The server has received the initial part of the request and the client should continue to send the remaining part. It is issued on a provisional basis while request processing continues. It alerts the client to wait for a final response.
101	Switching Protocols	The requester has asked the server to switch protocols and the server has agreed to do so. The target protocol must be more advanced than the source protocol. For example, the current HTTPS protocol is switched to a later version.
200	OK	The server has successfully processed the request.
201	Created	The request has been fulfilled, resulting in the creation of a new resource.
202	Accepted	The request has been accepted, but the processing has not been completed.

Status Code	Coding	Description
203	Non-Authoritative Information	The server has successfully processed the request, but is returning information that may be from another source.
204	No Content	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 the user or user agent (such as a browser) can choose the most appropriate one.
301	Moved Permanently	The requested resource has been assigned a new permanent URI, and the new URI is contained in the response.
302	Found	The requested resource resides temporarily under a different URI.
303	See Other	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 a case, there is no need to retransmit the resource since the client has a previously-downloaded copy.
305	Use Proxy	The requested resource must be accessed through a proxy.
306	Unused	The HTTP status code is no longer used.
400	Bad Request	The request is invalid. The client should not repeat the request without modifications.
401	Unauthorized	The authorization information provided by the client is incorrect or invalid.
402	Payment Required	This status code is reserved for future use.

Status Code	Coding	Description
403	Forbidden	The server has received the request and understood it, but refuse to respond to it. The client should modify the request instead of re-initiating it.
404	Not Found	The requested resource cannot be found. The client should not repeat the request without modifications.
405	Method Not Allowed	The method specified in the request is not supported for the requested resource. The client should not repeat the request without modifications.
406	Not Acceptable	The server cannot fulfill the request according to the content characteristics of the request.
407	Proxy Authentication Required	This status code is similar to 401, but indicates that the client must first authenticate itself with the proxy.
408	Request Timeout	The server has timed out waiting for the request. The client may repeat the request without modifications at a later time.
409	Conflict	The request could not be processed due to a conflict. For example, an edit conflict between multiple simultaneous updates or the resource that the client attempts to create already exists.
410	Gone	The requested resource has been deleted permanently and is no longer available. The status code indicates that the requested resource has been deleted permanently.
411	Length Required	The server refuses to accept the request without a defined Content-Length.
412	Precondition Failed	The server did not meet one of the preconditions contained in the request.

Status Code	Coding	Description
413	Request Entity Too Large	The request is larger than the server is willing or able to process. The server may close the connection to prevent the client from continuing the request. If the server is only temporarily unable to process the request, the response will contain a Retry-After header field.
414	Request URI Too Long	The Request-URI is too long for the server to process.
415	Unsupported Media Type	The server is unable to process the media format in the request.
416	Requested Range Not Satisfiable	The requested range is invalid.
417	Expectation Failed	The server has failed to meet the requirements of the Expect request-header field.
422	Unprocessable Entity	The request is well-formed but cannot be processed due to semantic errors.
429	Too Many Requests	The client has sent an excessive number of requests to the server within a given time (exceeding the limit on the access frequency of the client), or the server has received an excessive number of requests within a given time (beyond its processing capability). In this case, the client should resend the requests at any point after the time specified in the Retry-After header of the response.
500	Internal Server Error	The server is able to receive the request but unable to understand it.
501	Not Implemented	The server does not support the function required to fulfill the request.
502	Bad Gateway	The server was acting as a gateway or proxy and received an invalid response from the upstream server.
503	Service Unavailable	The requested service is invalid. The client should not repeat the request without modifications.
504	Gateway Timeout	The request cannot be fulfilled within a given time. This status code is returned to the client only if the **Timeout** parameter is specified in the request.

Status Code	Coding	Description
505	HTTP Version Not Supported	The server does not support the HTTPS protocol version used in the request.

4.2 Error Codes

If an error occurs during API calling, no result will be returned. You can locate the cause of an error using the error code of each API. When an API call fails, HTTPS status code 4xx or 5xx is returned. The returned message body contains a specific error code and error message. If you fail to identify the cause of an error, contact technical support and provide the error code for quick troubleshooting.

Format of an Error Response Body

If an error occurs during API calling, an error code and a message will be displayed. The following shows an error response body.

```
{
    "error_msg": "The input parameter is invalid.",
    "error_code": "AIS.0101"
}
```

In the response body, **error_code** indicates the error code, and **error_msg** provides information about the error.

Error Codes

OCR is deployed on ModelArts and uses API Gateway (APIG). Therefore, OCR error codes may start with ModelArts, APIGW, or APIG.

If an error code starting with **APIGW** is returned after you call an API, resolve the problem by referring to the instructions provided in . If an error code starts with APIG, rectify the fault by referring to this table.

Module	Error Code	Error Message.	Description	Solution
OCR	AIS.0101	The input parameter is invalid.	The input parameter does not meet the requirements.	Check the entered parameter. For details about the input parameter format of each API, see the request parameters in section "APIs" in *Optical Character Recognition API Reference*.

Mod ule	Error Code	Error Message.	Description	Solution
	AIS.0102	The image format is not supported.	The image format is not supported.	Check the image format.
	AIS.0103	The image size does not meet the requirements.	The image size does not meet the requirements.	Check the image size.
	AIS.0104	The image is not supported or the image quality is poor.	The image is not supported or is of poor quality.	Check the image type and quality. Check whether the called API is correct.
	AIS.0105	Recognition failed.	Algorithm calculation failed.	Contact technical support.
Platf orm	ModelArt s.0203	Invalid token.	The token is invalid.	Check whether the token is correct.
	ModelArt s.4101	Token header cannot be empty.	The token is empty.	The HTTP request header does not contain the token request authentication information of **x-auth-token**. Check the request.
	ModelArt s.4102	Parse Token error.	The token failed to be parsed.	The token request authentication information of **x-auth-token** in the HTTP request header is incorrect. Check the sent request and token.
	ModelArt s.4103	Invalid Token header.	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.
	ModelArt s.4104	Invalid Request Content Length.	The length of the request body is invalid.	Check the request body length. The size cannot exceed 10 MB.

Mod ule	Error Code	Error Message.	Description	Solution
	ModelArt s.4105	The JSON format of the input data is incorrect.	The JSON format of the request body is incorrect.	Check the JSON format of the request body.
	ModelArt s.4106	Invalid authorization request.	The account is restricted.	Check the user's resources.
	ModelArt s.4107	Get user temp ak sk error.	An exception occurred when obtaining the temporary AK/SK.	Contact technical support.
	ModelArt s.4201	Request url need service id.	The request URL does not contain the service ID.	Check the service ID in the request URL.
	ModelArt s.4202	Request url format invalid.	The request URL format is invalid.	Check the request URL format.
	ModelArt s.4203	Access denied! You do not have permission.	Access permission is unavailable.	Check the access permission.
	ModelArt s.4204	Request api error! The API XXX is not subscribed.	Failed to request the service because the service has not been subscribed.	● Go to the console, select an appropriate region, and subscribe to the API you need. ● If the service has been subscribed to, check whether the region (or account) where the service is subscribed to is the same as the region (or account) where the service is called. If they are the same, check whether the URL of the API is spelled correctly.

Mod ule	Error Code	Error Message.	Description	Solution
	ModelArt s.4301	Have error when get ai-service admin token.	Failed to obtain the admin token of the service.	Contact technical support.
	ModelArt s.4302	Gateway forwarding error.	The serviced failed to be accessed.	Contact technical support.
	ModelArt s.4401	Max concurrency error.	The number of concurrent requests exceeds the upper limit.	Contact technical support.
	ModelArt s.4402	Backend service timeout error.	Service processing timed out.	Contact technical support.
	ModelArt s.4403	Backend service unavailable error.	The service is unavailable because it failed to respond or its failure rate is high.	Contact technical support.
	ModelArt s.4502	Backend service response error.	The service failed to respond.	Contact technical support.
	ModelArt s.4503	Backend service found error.	The service does not exist.	Contact technical support.
	ModelArt s.4504	Backend service api not found.	The API to be accessed does not exist.	Contact technical support.
	ModelArt s.4505	Backend service internal error.	The internal service error occurs.	Contact technical support.
	ModelArt s.4506	SubService service found error.	The subservice corresponding to the API does not exist.	Contact technical support.
	ModelArt s.4508	Service not start.	The service does not exist or is stopped.	Contact technical support.

Mod ule	Error Code	Error Message.	Description	Solution
	ModelArt s.4601	The URL is not allowed.	The external URL is invalid.	Check the format of the entered download address.
	ModelArt s.4603	Obtaining the file from the URL failed.	The file failed to be downloaded from the external URL.	Check the network and URL.
	ModelArt s.4702	Query Obs agency failed.	The OBS agency failed to be queried.	Check whether the OBS agency has been enabled for the service.
	ModelArt s.4703	The Obs URL is invalid.	The OBS URL is invalid.	Check the OBS URL.
	ModelArt s.4704	Obtaining the file from the OBS failed.	The OBS file failed to be obtained.	Check the OBS file.
	ModelArt s.4705	The file stored on the OBS is oversized.	The OBS file is oversized.	Check the size of the OBS file and ensure that the file does not exceed the size limit.
	ModelArt s.4706	The Obs file is not exist.	The OBS file does not exist.	Check whether the corresponding file exists.

Mod ule	Error Code	Error Message.	Description	Solution
APIG	APIG. 0101	The API does not exist or has not been published in the environment.	The API does not exist or has not been published.	● Check whether the API URL is correct. For example, check whether the project ID is included in the URL. ● Check whether the region in the URI is the same as that configured for calling the service by referring to **Endpoints**. ● Check whether the HTTP request method (such as POST and GET) is correct. ● For details about the URI, see the corresponding API page.
	APIG. 0201	Backend timeout.	Request timed out.	Contact technical support.

Mod ule	Error Code	Error Message.	Description	Solution
	APIG. 0301	Incorrect IAM authentication information.	The IAM authentication information is incorrect. • **decrypt token fail**: The token fails to be parsed. • **token expires**: The token expires. • **verify aksk signature fail**: The AK/SK authenticati on fails. • **x-auth-token not found**: The **x-auth-token** parameter is not found.	• If the token fails to be parsed, check the method for obtaining the token, whether the request body is correct, whether the token is correct, and whether the environment for obtaining the token is the same as the environment for calling the token. • If the token expires, obtain a new token that is valid permanently. • Check whether the AK/SK pair is correct. For example, the SK corresponding to the AK is incorrect; an extra space is entered in the AK/SK pair. • AK/SK-based authentication errors occur frequently. If an AK/SK pair fails to be authenticated for more than five consecutive times, the AK/SK pair is locked for 5 minutes (the AK/SK-based authentication is considered as an abnormal

Mod ule	Error Code	Error Message.	Description	Solution
				authentication request within 5 minutes). After 5 minutes, the AK/SK pair is unlocked and re-authenticated. ● Check whether the spelling of **X-Auth-Token** in the request header is correct.
	APIG. 0308	The throttling threshold has been reached: policy user over ratelimit,limit:XX,ti me:1 minute.	The request exceeds the default rate limit of the service.	Contact technical support.

4.3 Obtaining an Account ID and Project ID

When calling APIs, you need to enter the username, user ID, project name, and project ID in some URIs. You can obtain the information on the **My Credentials** page.

1. Log in to the console.

2. Click the username in the upper right corner, and choose **My Credentials** from the drop-down list.

3. On the **My Credentials** page, you can view the username, project ID, project name, and project ID.

A Change History

Release Date	Description
2020-12-25	This issue is the first official release.