Модуль видеораспознавания товаров VISION-AI. Описание REST API

version 1.0.1

Mertech

April 10, 2025

Содержание

← Вернуться к списку всех документаций

Описание REST API

Интерфейс REST API

Connection SETUP

VISION-AI module USB-RNDIS configuration

The new generation of intelligent VISION-AI module is different from the previous USB-HID mode, and uses the USB-RNDIS mode to communicate with the host computer. In addition, the HTTP method is used to make the communication content richer and more diverse, decoupled from the specific implementation language, and more convenient to integrate.

Implementation principle

USB-RNDIS mode means that when the VISION-AI module is inserted into the weighing terminal, it will be recognized as a USB network card (in most cases, no drive is required). After the configuration is correct, the weighing terminal and the VISION-AI module form a local area network, so that the two can use the network protocol to access each other. This device uses the HTTP protocol for interaction.

Configure on Linux

IP configuration

VISION-AI module:

IP: 172.22.0.6

Mask: 255.255.255.0 (already configured, no need to configure)

Linux-pos end:

IP: 172.22.0.7

Mask: 255.255.255.0

After the VISION-AI module is inserted into the linux terminal, after the module is turned on normally (usually within 30 seconds), it will generally be correctly recognized as a USB network card by Linux. Since the VISION-AI module is configured with DHCP service, some systems will automatically configure the IP and subnet mask for the network card, as shown above. Generally, the network card is usb0, and if the ip has been configured through `ifconfig usb0` command, skip this section.

On some Linux, the configuration of the network card cannot be loaded automatically. At this time, manual configuration is required. First, confirm whether the network card is correctly recognized by the system. Use the `ifconfig -a` command to check whether there is a device with a name similar to usb0. If so, configure the ip and enable the network card through `ifconfig usb0 172.22.0.7 netmask 255.255.255.0 up`, if the configuration is successful, you can use `ping 172.22.0.6` to confirm whether the connection with the module is normal.

If the usb network card is not found through the `ifconfig -a` command, first use the `lsusb | grep 2207:372d` command to confirm whether the usb is enumerated successfully. 2207:372d is the VID and PID of this module in USB-RNDIS mode. If the enumeration is unsuccessful, please confirm the connection cable and firmware version of the module. If the enumeration is successful and there is no such network card node, please contact us.

Communication

The software and VISION-AI modules on the scale-end Linux use `http://172.22.0.6:80/{api}` to access. For details, see the relevant interface documentation.

Notice

* Please ensure that the above IP is configured and the VISION-AI module can be pinged normally on the Linux side. If the IP is incorrectly configured, the communication will fail.

* The IP of the weighing terminal must be `172.22.0.7`, because this IP is the gateway to identify the module. Please do not set other IP.

* Please make sure that there are no other devices on the 172.22.0.0/24 network segment.

* After the IP configuration is restarted on the scale end Linux, or after the VISION-AI module is restarted, the corresponding configuration may be lost and may need to be reconfigured. When the firmware of the VISION-AI module is upgraded, it needs to be restarted. In other cases, the VISION-AI module generally does not restart.

* Network sharing and proxy configurations may be lost after the end of Linux restarts, and may need to be reconfigured.

* The VID and PID of the old version of the USB-HID device are 2207:3729, and the VID and PID of the new version of the USB-RNDIS device are 2207:372d. If you need to be compatible with both new and old devices. Please make the distinction.

Device management

Query recognition module info

This interface is used to query the current recognition modules SDK version, firmware version, algorithm model and synchronization package version, SN serial number, customer number, customer name information,etc.

Interface URL

http://172.22.0.6/api/get_device_info

Query way

POST

Content-Type

none

The exsample of a successful response


```
{
    "code": 0,
    "data": {
        "clientID": "11095",
        "clientName": "Fruit Store Name",
        "engine": "9.5.0",
        "firmware": "9.9.4.2",
        "learnMode": 1,
        "masterIP": "",
        "masterSN": "",
        "sdk": "9.1.1",
        "sn": "12345678",
        "soldOutLimit": 3,
        "sync": "9.5.17.0",
        "syncMode": "normal"
    }
}
```


Parameter name	Example value	Parameter type	Parameter description
code	0	Integer	Request return value: 0 means success; Not 0 means err. See error code table

data	—	Object	Returns the data segment, or null if none.
data.clientID	11095	String	Customer store number, factory default 0000
data.clientName	Lan	String	Customer store name, factory is not activated
data.engine	9.5.0	String	Recognize engine version number
data.firmware	9.9.4.2	String	Firmware version no.
data.learnMode	1	Integer	Learning mode. There are two: 1 recognition while learning; 0 only recognition. Default 1
data.masterIP	—	String	Main scales ip, only when sync_mode = «slave», the field will have the valid data, otherwise it is null character.
data.masterSN	—		Main scales sn, this field is valid only when sync_mode = «slave», there is valid data, otherwise it is a null character.
data.sdk	9.1.1	String	Sdk version
data.sn	12345678	String	Recognition serial number
data .soldOutLimit	3	Integer	Automatic removal threshold.
data.sync	9.5.17.0	String	Synchronization package version number.
data.syncMode	Normal	String	There are three mode of synchronization: «normal», «master», «slave». Default is «normal»

Factory reset

This interface restores the VISION-AI module to factory settings, cancels the association between the store and the device, and clears all caches, identified products, and registered products.

Interface URL

http://172.22.0.6/api/reset_device

Query way

POST

Content-Type

none

Restart recognition module

Call this interface to restart the recognition module

Interface URL

http://172.22.0.6/api/reboot

Query way

POST

Content-Type

none

Example of a successful response

```
{
    "code": 0,
    "data": null
}
```

Parameter name	Example value	Parameter type	Parameter description
code	—	Number	Request return value: 0 means success, no 0 means error see detail in error code list
data	—	Object	Returns the data segment, or null if none.

Set the recognition area

The recognition module needs to calibrate the recognition area to improve the recognition accuracy

Before calibration, you need to call the get_image interface to display a picture, and returned picture with resolution 1280x720, and then manually frame the coordinates of the recognition area on the current picture, and pass the coordinates into the set_calib interface

When the location of the device changes, the recognition area needs to be reset.

Interface URL

http://172.22.0.6/api/set_calib

Query way

POST

Content-Type

json

Request Body parameter

```
{
    "height": 482,
    "left": 444,
    "top": 39,
    "width": 348
}
```

Example of a successful response

```
{
    "code": 0,
    "data": null
}
```

Parameter name	Example value	Parameter type	Parameter description

code	—	Number	Request return value: 0 means success, no 0 means error see detail in error code list
data	—	Object	Returns the data segment, or null if none.

Query recognition area

Get the coordinate parameters of the current recognition area, the coordinates are based on the 1280x720 resolution picture.

Interface URL

http://172.22.0.6/api/get_calib

Query way

POST

Content-Type

none

Example of a successful response

Parameter name	Example value	Parameter type	Parameter description
code	—	Number	Request return value: 0 means success, no 0 means error see detail in error code list
data	—	Object	Returns the data segment, or null if none.
data.height	482	Number	Height of the recognition area
data.left	444	Number	The abscissa of up left recognition area
data.top	39	Number	The ordinate of the left recognition area
data.width	348	Number	Width of the recognition area

Query learning mode

Query the status of the current learning mode.

Interface URL

http://172.22.0.6/api/get_learnmode

Query way

POST

Content-Type

none

Example of a successful response

```
{
    "code": 0,
    "data": {
        "mode": 0
    }
}
```

Parameter name	Example value	Parameter type	Parameter description
code	—	Number	Request return value: 0 means success, no 0 means error see detail in error code list
data	—	Object	Returns the data segment, or null if none.
data.mode	—	Number	Learning mode: 0: learning mode off 1: learning mode on

Set learning mode

The setting switch of the learning mode enables the recognition module to freely turn off and turn on the learning function.

When it is turned on, returns the clicked result, interface has a learning function.

Interface URL

http://172.22.0.6/api/set_learnmode

Query way

POST

Content-Type

json

Request Body parameter

```
{
    "mode": 0
}
```

Parameter name	Example value	Parameter type	Is it necessary?	Parameter description
mode	—	Number	Yes	Learning mode: 0: learning mode off 1: learning mode on

Example of a successful response

```
{
    "code": 0,
    "data": null
}
```

Parameter name	Example value	Parameter type	Parameter description
code	—	Number	Request return value: 0 means success, no 0 means error see detail in error code list

data	—	Object	Returns the data segment, or null if none.

Category management

Query the recognizable list

Get the list of recognizable product IDs.

Interface URL

http://172.22.0.6/api/get_item_list

Query way

POST

Content-Type

none

Example of a successful response

Parameter name	Example value	Parameter type	Parameter description
code	—	Number	Request return value: 0 means success, no 0 means error see detail in error code list
data	—	String	Returns the data segment, or null if none.
data.id	001	String	Recognizable category id
data.name	Test1	Object	The name corresponding to the identifiable category id.

Register pictures to cache

Register the template image to the cache area, and the template in the registered cache area needs to be submitted to take effect.

Interface URL

http://172.22.0.6/api/register_item

Query way

POST

Content-Type

json

Request Body parameter

```
{
    "id": "001",
    "name": "Test"
}
```

Parameter name	Example value	Parameter type	Is it necessary?	Parameter description

item_id	001	String	Yes	Product ID that needs to be registered
item_name	Test	String	Yes	The product ID that needs to be registered corresponds to the Chinese name (utf-8code)

Example of a successful response

```
{
    "code": 0,
    "data": {
        "image": "example_base64",
        "image_id": "2022032217214958388443 2"
    }
}
```

Parameter name	Example value	Parameter type	Parameter description
code	—	Number	Request return value: 0 means success, no 0 means error see detail in error code list
data	—	Object	Returns the data segment, or null if none.
data.image	example_base64	String	The base64 encoding of the registration template image and image format is jpeg
data.image_id	202203221721 49583884432	String	The image ID of the product ID that needs to be registered.

Delete the picure in the registration cache area

This command is to delete the specified picture that has been registered in the cache

Interface URL

http://172.22.0.6/api/delete_registered_image

Query way

POST

Content-Type

json

Request Body parameter

```
{
    "id": "001",
    "image_id": "20220316161042636773341"
}
```

Parameter name	Example value	Parameter type	Is it necessary?	Parameter description
item_id	1	String	Yes	The product ID to be deleted
image_id	202203161606 39483889308	String	Yes	Image ID of product ID need to be deleted

Example of a successful response

```
{
    "code": 0,
    "data": null
}
```

Parameter name	Example value	Parameter type	Parameter description
code	—	Number	Request return value: 0 means success, no 0 means error see detail in error code list
data	—	Object	Returns the data segment, or null if none.

Delete all picture in the category cache

This command is to delete all pictures registered in the cache under the specified category.

Interface URL

http://172.22.0.6/api/delete_registered_item

Query way

POST

Content-Type

json

Request Body parameter

```
{
    "id": "001",
}
```

Example of a successful response

```
{
    "code": 0,
    "data": null
}
```

Parameter name	Example value	Parameter type	Parameter description

code	—	Number	Request return value: 0 means success, no 0 means error see detail in error code list
data	—	Object	Returns the data segment, or null if none.

Submit the cache picture

Submit all the pictures in the temporary storage area of a category, and they will be registered in the VISION-AI list after submission.

Interface URL

http://172.22.0.6/api/commit_registered_item

Query way

POST

Content-Type

json

Request Body parameter

```
{
    "id": "001",
}
```

Parameter name	Example value	Parameter type	Is it necessary?	Parameter description
item_id	001	String	Yes	The product ID that needs to be submitted

Example of a successful response

```
{
    "code": 0,
    "data": null
}
```

Parameter name	Example value	Parameter type	Parameter description
code	—	Number	Request return value: 0 means success, no 0 means error see detail in error code list
data	—	Object	Returns the data segment, or null if none.

Get the product ID and picture ID that have not been submitted in the current cache

Obtain the product ID and image ID that have not been submitted in the current cache, and you can query image_id for submission or deletion.

Interface URL

http://172.22.0.6/api/get_registered_item_list

Query way

POST

Content-Type

none

Example of a successful response

Parameter name	Example value	Parameter type	Parameter description
code	—	Number	Request return value: 0 means success, no 0 means error see detail in error code list
data	—	Object	Returns the data segment, or null if none.
data.id	001	String	The registered category id.
data.image_id	20220322175440932968580	String	The registered image id.
data.name	Test	String	The name corresponding to the recognizable category id.

Local backup

Function description

Start to backup the local algorithm data, and you need to call the 3.10 interface to query the backup process condition.

Interface URL

http://172.22.0.6/api/start_backup

Query way

POST

Content-Type

none

Example of a successful response

```
{
    "code": 0,
}
```

Parameter name	Example value	Parameter type	Parameter description

code	0	Number	Request return value: 0 means success, no 0 means error see detail in error code list

Query backup status

Function description

Query the status of backup

Status == 0 || status >= 3 mean the and of backup, other status need to continue call this interface. Suggest one call by one second, see the command return table.

After backup competed, the interface will return a filename and download the backup file according to the name(can choose the download platform), the download link format: http://172.22.0.6/download/{filename}

The following request example returns the actual download link, for example: http://172.22.0.6/download/test.evo

Interface URL

http://172.22.0.6/api/get_backup_status

Query way

POST

Content-Type

none

Example of a successful response

```
{
    "code": 0,
    "filename": "test.evo",
    "msg": "backup complete",
    "status": 3
}
```

Parameter name	Example value	Parameter type	Parameter description
code	0	Int	Request return value: 0 means success, no 0 means error see detail in error code list
status status status status	3	Int Int Int Int	Backup data status code: 0 no backup task 1 data generating 3 data backup completed 6 data fail to generate
filename	test.evo	String	The file name generated by the backup data is valid only when status = 3, otherwise it is empty
msg	back up complete	String	The Chinese description of the backup data status, which can be directly used for display.

Call example

Local recovery

Function Description:

Send the backed up file to the recognition module to restore the algorithm data through this interface, and the body is in multipart/form-data format. The return of the interface indicates that the file transfer is completed, and the subsequent recovery progress needs to call 4.4 interface query.

Interface URL

http://172.22.0.6/api/start_restore

Query way

POST

Content-Type

Form-data

Request Body parameter

files to restore

Example of a successful response

Parameter name	Example value	Parameter type	Parameter description
code	—	Number	Request return value: 0 means success, no 0 means error see detail in error code list
data	—	Object	Returns the data segment, or null if none.

Call example

```
curl --location --request POST 'http://172.22.0.6/api/restore' \\
--form '=@"/X:/data/test.evo"'
```

Query recovery status

Function description

Query the status of restored data.

Status == 0 || status >= 4 means that the recovery has ended, if continue use this interface to call other status, suggest to call once a second, see the command return table for details.

Interface URL

http://172.22.0.6/api/get_restore_status

Query way

POST

Content-Type

none

Example of a successful response

Parameter name	Example value	Parameter type	Parameter description

code	0	int	Request return value: 0 means success, no 0 means error see detail in error code list
msg	Data recovery complete	String	The Chinese description of the restored data status, which can be directly used for display.
status	0	int	Backup data status code: 0 no recovery task 2 data recovering 3 data on verification 4 data recovery completed 5 data fail to recovery 7 data fail to verify

Category VISION-AI

Get the recognition result

Obtain a recognition result, and suggest up to 10 results at a time, sorted by confidence from high to low

Interface URL

http://172.22.0.6/api/get_result

Query way

POST

Content-Type

none

Example of a successful response


```
{
    "code": 0,
    "data": {
        "items": [
            {
                "id": "002",
                "score": 0.55000000000000004
            },
            {
                "id": "001",
                "score": 0.45000000000000001
            }
        ],
        "sid": "20220414150011198237897"
    }
}
```


Parameter name	Example value	Parameter type	Parameter description
code	—	Number	Request return value: 0 means success, no 0 means error see detail in error code list

data	—	String	Returns the data segment, or null if none.
data.items	—	Object	Recognizes the result set.
data.items.id	002	String	Product id
data .items.score	0.55	Number	According to the confidence level, the higher of it, the product will more similar to recommended id. A floating point number in the range [0,1]
data.sid	202204141500 11198237897	String	The unique id identified this time is used for clicking return.

Return the recognition result

The recognition module needs to learn continuously according to the real results to enhance the recognition ability.

If the obtained recognition result is correct, the prediction result is returned to the recognition module.

If the recognition result is incorrect, the real category needs to be returned to the recognition module.

Interface URL

http://172.22.0.6/api/result_feedback

Query way

POST

Content-Type

json

Request Body parameter

```
{
    "sid": "2022041415001198237897",
    "id": "002",
    "name": "test",
    "intop": true
}
```

Parameter name	Example value	Parameter type	Is it necessary?	Parameter description
sid	2022041 4150011 198237897	String	Yes	The unique id of the image, this parameter is required for submitting registration or deleting the cache template interface.

item_id	001	String	Yes	Click on the product ID, which is the ID of the product selected by the clerk.
item_name	test	String	Yes	Click the product name
intop	true	Boolean	Yes	Whether the click result is in the recommended list (true/false)

Example of a successful response

```
{
    "code": 0,
    "data": null
}
```

Parameter name	Example value	Parameter type	Parameter description
code	—	Number	Request return value: 0 means success, no 0 means error see detail in error code list
data	—	Object	Returns the data segment, or null if none.

General interface

Get the current camera screen

Get a frame of current module, return data type Content-Type: image/jpeg, body is an unsigned char array, picture format:jpeg , resolution:1280*720. It can be use for calibration of the weighing pan.

Interface URL

http://172.22.0.6/api/get_image

Query way

POST

Content-Type

none

Example code

```
response = http.post("http://172.22.0.6/api/get_image")
unsigned char \* p = response.body_binary;
int len = response.body_binary_length;
FILE\* fp = fopen("test.jpg", "wb");
fwrite(p, sizeof(unsigned char), len, fp);
fclose(fp)
```

Error code table

Code	Meaning
-6103	Camera lost

-6117	Device communication unkown command
-6124	Device is activated
-6126	Store id format error
-6127	The server failed to verify the store
-6128	Device not activated
-6130	Download not started
-6131	The device is already the latest version
-6132	Download verification failed
-6133	Download failed
-6138	Device firmware mismatch, function limited
-6140	File format mismatch
-6141	Store mismatch
-6142	Device mismatch
-6143	Engine version mismatch
-6200	Json parsing failed
-6202	File does not exist
-6203	Failed to write file
-6204	Failed to write file
-6207	Network Error
-6209	Algorithm api timeout
-6210	Unknown mistake
-6211	File verification failed
-6212	Login failed
-6213	Parameter error
-6214	Sync request timed out
-65xx	Algorithm api error

Документация PDF