

Goodview Dynamic Detection Display LAN Interface V1.4

Copyright Notice

The copyright of this manual belongs to Shanghai Goodview Electronic Technology Co., Ltd. and reserves all rights.

The communication technology company agrees (in writing) that no unit or individual may extract part or all of this manual without authorization.

Its legal responsibility will be pursued.

About this document

This document is used as a guide. The photos, graphics, charts and illustrations provided in the document are for explanation and explanation only.

The purpose may differ from the specific product, please refer to the actual product

Release notes

Version	Description	Date
V1.0	Created	2019-07-26
V1.1	Added temperature interface	2020-02-28
V1.2	Add to get the setting parameters, set the card number, get the device parameters, set the time, set the specific interface	2020-03-22
V1.3	Add stranger record switch parameter setting; Added fan switch; Added access to temperature parameters, open the door, and restart the interface;	2020-03-28

Content

EZ-Pass Access LAN Interface V1.3	1
Copyright notice	2
Release Notes	3
Interface Operation Guide	5
Interface description	5
Interface List	6
1. Set device password	6
2. Parameter configuration (face recognition, opening mode, etc.).....	7
3. Obtain parameter configuration (face recognition, open mode, etc.)	9
4. Personnel registration	10
5. Personnel information editing	11
6. Delete personnel	12
7. Acquisition of identification records	12
8. Modify LOGO	14
9. Get device mac address	14
10. Parameter configuration (body temperature, mask, fan, etc.)	15
11. Access to photo	16
12. Device parameter information acquisition	16
13. Taking pictures of the device	17
14. Device initialization	17
15. System time setting	18
16. Recognition record interface callback settings	18
17. Device opening	19
18. Device restart	19
19. Get body temperature and mask parameters	20

Interface specification

- Interface root address: http: // device ip address: 8080 /
 - Interface form: provide external services through HTTP requests.
 - Interface security: The device password (pass) needs to be set first when calling
 - the interface for the first time, and the device password (pass) needs to be passed in as the interface security verification key for any subsequent calls to the interface.

Interface returns

Interface general return instructions:

```
public class ResultInfo <T> {  
  
private Int result; // Indicates whether the interface is tuned up, 1: succeeds, 0: fails,  
usually as long as the device server can respond, the value is 1  
  
private Boolean success; // Whether the operation is successful, the success is true, the  
failure is false private T data; // The business data returned by the interface, the type can  
be numeric, string or collection, etc
```

Interface list

1. Set device password

Request method: POST request

Request address: http://device IP:8080/setPassWord

DATE:

Filed	Date	request	description
oldPass	String	Y	Old pass word
newPass	String	Y	New pass word,

Request description:

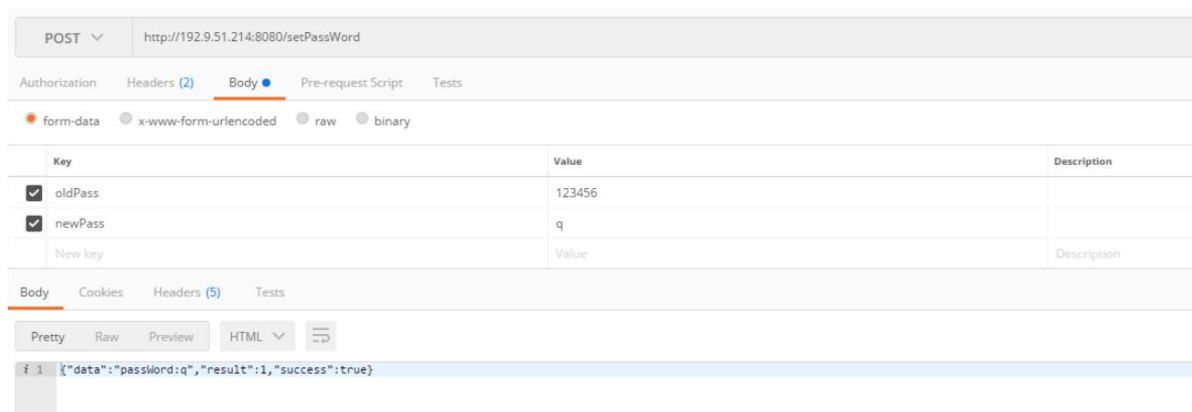
For new devices or devices after resetting (restoring initialization), you need to set the initial password before calling other interfaces. OldPass and newPass can be passed the same value.

When changing the password, pass in the new and old password separately

Eg:

```
{
  "data": "passWord:q", // The device password, also called the interface calling
  password, please keep it properly. If you forget your password,
  You need to reset the device, the device will erase all data
  "result": 1, // Interface tuning
  "success": true // Device password set successfully
}
```

Postman eg:



2. Parameter configuration (face recognition, opening mode, etc.)

Request method: POST request

Request address: http:// device IP: 8080 / setConfig

Date:

field	mode	request	description
pass	String	Y	Pass word setting
config	Json	Y	Parameter configuration Json The following are the parameters of json

Request description

POST request parameters are placed in the body

The config request parameter is in json format, the example is as follows (please adjust according to the actual situation

```
{
  "companyName": "smdt",
  "deviceId": 86,
  "diplayCustom": "{name}",
  "displayMode": 1,
  "id": 1,
  "liveIdentiLevel": 1,
  "passType": false,
  "password": "123456",
  "recoDistance": "1.5",
  "recoInterval": "2000",
  "relayDelay": 5,
  "relayMode": 0,
  "serialCustomize": "#{idcardNum}#",
  "serialMode": 2,
  "similarity": 80,
  "strangerVoiceCustom": "moshengren",
  "strangerVoiceMode": 2,
  "voiceCustom": "name",
  "voiceMode": 1,
  "wg": 0
}
```

field

field	mode	description
companyName	String	companyName
deviceId	int	deviceId
diplayCustom	String	diplayCustom, default as {name}
displayMode	int	displayMode, 1: name, 100: custom
id	int	

liveIdentiLevel	int	Living body recognition level, default fast mode, 0 no living body, 1 fast mode cannot Reject photos, 2 can reject some photos, 3 can reject photos and adapt
passType	boolean	Access type false into true out
password	String	passcode
recoDistance	String	Recognition distance (0.5~3)
recoInterval	String	Recognition interval (2000~10000)
relayDelay	Int	Relay automatically closes after a delay of x seconds (5~63)
relayMode	Int	Relay mode, 0: indicates the automatic closing mode, that is, X seconds will be delayed after the relay is opened. Dynamic closure 1: It means that it does not close automatically, that is, it will not close automatically after opening the relay. Fit
serialCustomize	String	Serial port definition, default is "#{idcardNum}#", Unlimited length
similarity	Int	Similarity, default 80,30 ~ 100 adjustment
strangerVoiceCustom	String	Stranger voice customization, default for stranger recognition, within 32 characters
strangerVoiceMode	Int	Stranger voice mode, the default stranger alarm, 1 does not require voice broadcast, 2 Stranger alarm, 100 custom
voiceCustom	String	Voice customization, default is "name", within 32 characters
voiceMode	Int	Voice mode, 1 no broadcast, 2 broadcast name, 100 custom
wg	Int	Wiegand 26 or 34 output mode, 0: output card number WG26, 1: output person Staff ID WG26,2: output card number WG34, 3: output staff ID WG34
icNO	int	IC Card, 1: Card Swipe
idCardNo	int	ID Card 1: Close
tricolorLamp	int	Red light, 1:Off, 2: On

Return data

```
{
  "data": { // Set parameters
    "companyName": "smdt",
    "deviceId": 86,
    "diplayCustom": "{name}"
  }
}
```



```
"displayMode": 1,  
"id": 1,  
"liveIdentiLevel": 1,  
"passType": false,  
"password": "123456",  
"recoDistance": "1.5",  
"recoInterval": "2000",  
"relayDelay": 5,  
"relayMode": 0,  
"serialCustomize": "#{idcardNum}#",  
"serialMode": 2,  
"similarity": 80,  
"strangerVoiceCustom": "moshengren",  
"strangerVoiceMode": 2,  
"voiceCustom": "name",  
"voiceMode": 1,  
"wg": 0  
},  
  
"result": 1, // Interface access  
"success": true // Setting successful  
}
```

3. Get parameter configuration (face recognition, switching mode, etc.)

Request method: POST request

Request address: http://device IP:8080/getConfig

Request data:

field	mode	request	description
pass	String	Y	Device pass word

Return eg:

```
{
  "data": { // Set parameters
    "companyName": "smdt",
    "deviceId": 86,
    "diplayCustom": "{name}",
    "displayMode": 1,
    "id": 1,
    "liveIdentiLevel": 1,
    "passType": false,
    "password": "123456",
    "recoDistance": "1.5",
    "recoInterval": "2000",
    "relayDelay": 5,
    "relayMode": 0,
    "serialCustomize": "#{idcardNum}#",
    "serialMode": 2,
    "similarity": 80,
    "strangerVoiceCustom": "moshengren",
    "strangerVoiceMode": 2,
    "voiceCustom": "name",
    "voiceMode": 1,
    "wg": 0
  },
  "result": 1, // Interface access
  "success": true // Setting successful
}
```

4. Personnel registration

Request method: POST request

Request address: http://device ip:8080/person/create

Request data:

field	mode	request	description
pass	String	Y	Device pass word
person	Json	Y	Personnel information Json The following are the parameters of json

Request description:

POST request parameters are placed in the body

The base64 string of the photo, without the header, such as: data: image / jpg; base64.

Staff photos only support JPG and PNG

Person filed

field	mode	description
age	int	age
imgBase64	String	imgBase64 String
name	String	name
prescription	String	yyyy-MM-dd HH:mm,yyyy-MM-dd HH:mm
sex	int	sex
type	Int	Type The default is 1, 1 for visitors, 2 blacklists, and 3 for employees
vipID	Int	Create a person's id, find the person's editor based on the id
welCome	String	Reserved, can be left blank
icCard	String	ic card information
card	String	ID CARD nO
wn	String	Reserved, can be left blank

RETURN eg:

```
{
  "data": "msg: Successful registration ",
  "result": 1,
  "success": true
}
```

Registration failed, return failure description, the corresponding error code is as follows:

5. Personnel information editing

Request method: POST request

Request address: http://device ip:8080/person/update

REQUEST DATE:

Filed	mode	request	description
pass	String	Y	Device pass word
person	Json	Y	Personnel information Json The following are the parameters of json

request:

POST request parameters are placed in the body

The base64 string of the photo, without the header, such as

Staff photos only support JPG and PNG

Person Parameter Description

filed	mode	Description
age	int	age
imgBase64	String	imgBase64 String
name	String	Name
prescription	String	, yyyy-MM-dd HH:mm,yyyy-MM-dd HH:mm
sex	int	sex
type	Int	Type The default is 1, 1 for visitors, 2 blacklists, and 3 for employees
vipID	Int	Create a person's id, find the person's editor based on the id
welCome	String	Reserved, can be left blank
icCard	String	ic card information
card	String	ID number
wn	String	Reserved, can be left blank

Example of returned data

```
{
  "data": "msg: Staff editing success ",
  "result": 1,
  "success": true
}
```

6. Delete person

Request method: POST request

Request address: http://device ip:8080/person/delete

Request date:

filed	mode	request	Description
pass	String	Y	Device pass word
deleteId	String	Y	Delete person id

Return eg:

```
{
  "data": "msg: Person deleted successfully",
  "result": 1,
  "success": true
}
```

7. Identification record acquisition

Request method: POST request

Request address: [http:// device ip:8080/newFindRecords](http://device ip:8080/newFindRecords)

Request date:

field	mode	request	Description
pass	String	Y	Device pass word
startTime	String	Y	start: 2019-07-11 12:00
endTime	String	Y	finish: 2019-07-11 17:00

Eg:

```
{
  "data": [
    {
      "birthDate": "",//reserved
      "currentTime": 1562819094334, // Transit time, milliseconds
      "idCardNum": "",// staff ID
      "imageFlag": 0,// The name of the photo taken by the public, if you want to get the photo, please
      call getRecordImg interface to get it.
      "name": "pdf", //passer name
      "type": 1, //mode
      "temprature": "36.5", // Body temperature when passing, transmitted when body temperature
      detection is off -1
      "mask": 1 // Whether to wear a mask, 1: wear, 0: no wear mask, □ ,-1
      when mask detection is closed},
    {
      "birthDate": "",
      "currentTime": 1562824109687,
      "idCardNum": "",
      "imageFlag": 0,
      "imageName": "1_86_1562824109687.jpg",
      "name": "pdf",
      "type": 1
      "temprature": "36.5",
      "mask": 1 // Whether to wear a mask, 1: worn, 0: not worn
    },
    {
      "birthDate": "",
      "currentTime": 1562825856546,
      "idCardNum": "null",
      "imageFlag": 1,
      "imageName": "100_86_1562825856546.jpg",
      "name": "tts",
      "type": 1,
      "temprature": "36.5",
      "mask": 1 // Whether to wear a mask, 1: wearing, 0: not wearing
      "birthDate": "",
```

```
"currentTime": 1562825856546,
"idCardNum": "null",
"imageFlag": 1,
"imageName": "100-86-1562825856546.jpg",
"name": "tts",
"type": 1,
"temprature": "36.5",
"mask": 1 //Whether to wear a mask, 1: wearing, 0: not wearing
}
],
"result": 1, // Interface transfer through
"success": true // success
```

8. Modify LOGO

Request method: POST request

Request address: http://device ip:8080/changeLogo

Request data:

field	mode	request	Description
pass	String	Y	device
imgBase64	String	Y	Logo pic Base64, only support png、jpg

request:

POST request parameters are placed in the body

The base64 string of the photo, without the header, such as: data:image/jpg;base65

Staff photos only support JPG and PNG

eg

```
{
  "data": "msg: logo modify successfully",
  "result": 1,
  "success": true
}
```

9. Acquisition of equipment mac address

Request method: POST request

Request address: http://device ip:8080/getDeviceMac

Request data:

field	mode	request	Description
pass	String	Y	Device pass word

eg

```
{
  "data": "8CFCA0000023",
}
```



```
"result": 1,  
"success": true  
}
```

10. Parameter configuration (body temperature, mask, fan, etc

Request method: POST request

Request address: http://device ip:8080/tempAndMaskSetting

Request date:

field	mode	request	Description
pass	String	Y	Device pass word
config	Json	Y	Personnel information Json. The following are the parameters of json

Request description

POST request parameters are placed in the body

The config request parameter is in json format, the example is as follows (please adjust according to the actual situation)

```
{  
  "isBodyTempAlarm": 1,  
  "isBodyTempStart": 1,  
  "isHighFeverAdopt": 0,  
  "isLowFeverAdopt": 0,  
  "isLowTempAdopt": 0,  
  "isStandardTempAdopt": 1,  
  "isWearingMask": 0,  
  "standardBodyTemp": "37.3",  
  "isStrangerRecord": 0,  
  "isFan": 0  
}
```

config description

filed	mode	Description
isBodyTempAlarm	int	1 Turn on body temperature alarm 0 Turn off body temperature alarm
isBodyTempStart	Int	1 Turn on body temperature detection 0 Turn off body temperature detection
standardBodyTemp	String	Body temperature threshold 37.3 (accurate to one decimal place), open body temperature detection and body temperature alarm After detecting that the body temperature exceeds the threshold, a sound alarm is played
isHighFeverAdopt	int	Whether high fever passed (0: not passed 1: passed) is used to open the door (range 38.5-43.0)
isLowFeverAdopt	int	Whether high fever passed (0: not passed 1: passed) is used to open the door (range 38.5-43.0)
isLowTempAdopt	int	Whether the low temperature is passed (0: not passed 1: passed) for door opening (range 30.0-below)
isStandardTempAdopt	int	Whether the normal body temperature is passed (0: not passed 1: passed) for opening the door (range 36.1-37.2)
isWearingMask	int	Wear mask detection switch (0: No 1: Yes)
isStrangerRecord	int	Stranger identification record storage switch (0: No, 1: Yes)
isFan	int	Fan switch (0: No, 1: Yes)
tempCompensation	float	Temperature compensation, range (-1~1), positive number means positive compensation, negative number means negative compensation

E eg

```
{
  "data": {
    "deviceId": 88,
    "id": 10,
    "isBodyTempAlarm": 1,
    "isBodyTempStart": 1,
    "isHighFeverAdopt": 0,
    "isLowFeverAdopt": 0,
    "isLowTempAdopt": 0,
    "isStandardTempAdopt": 1,
    "isWearingMask": 0,
    "standardBodyTemp": "37.3",
    "isStrangerRecord":0,
    "isFan":0
  },
  "result": 1,
  "success": true
}
```

11. Access to photos

Request method: POST request

Request address: http://device ip: 8080 / getRecordImg

Request data:

filed	mode	request	Description
pass	String	Y	device
imgName	String	Y	Photo name (obtained by the identification record interface)

EG

```
{
  "data":"/9j/4AAQSkZJRgABAQAAQABAAD/2wB", // Image's base64 string
  "result": 1, //1 Interface call 0 Interface abnormal
  "success": true // true Get success false Get failure;
}
```

12. Device parameter information acquisition

Request method: POST request

Request address: http://device ip: 8080 / getDeviceInfo Request data:

/ getDeviceInfo Request data:

field	mode	request	Description
pass	String	Y	Device password

eg

```
{
  "data": {"freeSpace": "3.61GB", // freeSpace: Remaining storage space
    "ip": "192.9.51.45", // ip: ip address
    "mac": "8CFCA0036225", // mac address
    "time": 1584518232928, // System time (ms) long type
    "version": "1.5.0.22.0.01" // app Current version number
  },
  "result": 1,
  "success": true
}
```

13. Taking pictures

Request method: POST request

Request address: http://equipment ip: 8080 / photograph

Request data:

field	mode	request	Description
pass	String	Y	Device password

eg

```
{
  "data": "/9j/4AAQSkZJRgABAQAA ...", //base64 String, need to be converted to suffix png pictures
  "result": 1,
  "success": true
}
```

14. Device initialization

Request method: POST request

Request address: http://device ip: 8080 / initialization Request data:

field	mode	request	Description
pass	String	Y	Device password

- Request description:
- Delete all identification records, personnel data, characteristics and other data on the device, and clear all databases
 - Delete the attributes set through the device configuration interface

eg

```
{
  "data": " Initialization successful ",
  "result": 1,
  "success": true
}
```

15. System time setting

Request method: POST request

Request address: http:// device ip: 8080 / setDeviceTime

Request data:

field	mode	request	Description
pass	String	Y	Device pass word
year	String	Y	year
month	String	Y	month
day	String	Y	day
hours	String	Y	hours
minute	String	Y	minute

Request description:

After successful configuration, the device time is changed to the currently set time

If the device is connected to the public network, the device itself has a network time calibration mechanism, and the system adjusts the device time to the public network time

For the device to display the time manually set, the device must be in the local area network. If connected to the public network, the device uses the public network time by default when refreshing its time.

Eg:

```
{
  "data": "2020-03-15 22:10:50", // Device current time
  "result": 1,
  "success": true
}
```

16. Identify the recording interface callback settings

Request method: POST request

Request address: http:// device ip: 8080 / setIdentifyCallback

Request data:

field	mode	request	Description
pass	String	Y	Device pass word
callbackUrl	String	Y	Identify the url interface for record upload

Request description:

The callback can be set in other settings of the APP terminal settings

Please refer to the MIPS face recognition callback interface document V1.0

EG:
EG:

```
{
  "data": "http://192.9.51.45:8080/setIdentifyCallback",
  "result": 1, //1 Set successfully -1 No callbackUrl field -2 callbackUrl Field is empty
  "success": true //true Set successfully, false set failed;
}
```

17. Device-door control

Request method: POST request

Request address: http://device ip: 8080 / device / openDoorControl

Request data:

field	mode	request	Description
pass	String	Y	Device pass word

Eg

```
{
  "data": "open the door success",
  "result": 1,
  "success": true
}
```

Device restart

Request method: POST request

Request address: http://device ip: 8080 / restartDevice Request data:

field	mode	request	Description
pass	String	Y	设 Device pass word

Eg

```
{
  "data": " Restart the success ",
  "result": 1,
  "success": true
}
```

18. Get body temperature and mask parameters

Request method: POST request

Request address: http://device ip: 8080 / getTempAndMaskSetting

Request data:

field	mode	request	Description
pass	String	Y	Device pass word
config	Json	Y	Parameter configuration Json. The following are the parameters of json

Return data (for example, please refer to the description of temperature and mask parameter configuration for details):

```
{
  "data": {
    "isBodyTempAlarm ":1,"
    isBodyTempStart ":1,"
    isHighFeverAdopt ":0,"
    isLowFeverAdopt ":0,"
    isLowTempAdopt ":0,"
    isStandardTempAdopt ":1,"
    isStrangerRecord ":0,"
    isWearingMask ":0,"
    standardBodyTemp ":"37.3 ","
    tempCompensation ":0.3,"
    tempCompensationParam ":1
  },
  "result": 1,
  "success": true
}
```


19. Personnel Check-up

Request method: POST request

Request address: `http://device ip:8080/person/find`

field	mode	request	Description
pass	String	Y	Device pass word
config	Json	Y	Person ID specifies that multiple persons to be queried are separated by commas. For example, "100,101,102", the maximum number is 50;

Return Data:

Except for the faceID field, please refer to interface 4 for the detailed field description

```
{
  "data": "[
    {
      "prescription": "2020-03-21 00:00,2030-12-13 00:00",
      "age": 0,
      "card": "",
      "faceID": 57358,
      "icCard": "",
      "name": "Lee",
      "sex": 1,
      "type": 3,
      "vipID": 57358 //Less than 0 means that the picture verification
failed, greater than or equal to 0 means success, the reason for failure
Please refer to the interface 4, Instructions for returning personnel
registration results
    }
  ],
  "result": 1,
  "success": true
}
```

20. Personnel page query

Request method: POST request

Request address: `http://device ip:8080/person/findByPage`

Field	Type	Required	Description
pass	String	Y	Device Password
index	int	Y	Page number, starting from 0
length	Int	Y	Maximum number per page, <= 50, if > 50, only 50 can be found

Return Data:

Except for the faceID field, please refer to interface 4 for the detailed field description

```
{
  "data": "{
    "pageInfo": {
      "index": 0,
      "length": 1,
      "size": 1,
      "total": 2
    },
    "records": [
      {
        "age": 0,
        "card": "",
        "faceID": 66614,
        "icCard": "",
        "name": "fff",
        "sex": 0,
        "type": 3,
        "prescription": "2020-03-21 00:00,2030-12-13 00:00",
        "vipID": 66614 //Less than 0 means that the picture verification
        failed, greater than or equal to 0 means success, the reason
        for failure
        Please refer to the interface 4, Instructions for returning
        personnel registration results
      }
    ]
  }",
  "result": 1,
  "success": true
}
```