

Body temperature detection and face recognition intelligent terminal HTTP docking protocol V1.16

1. Sample of web settings:



Example:

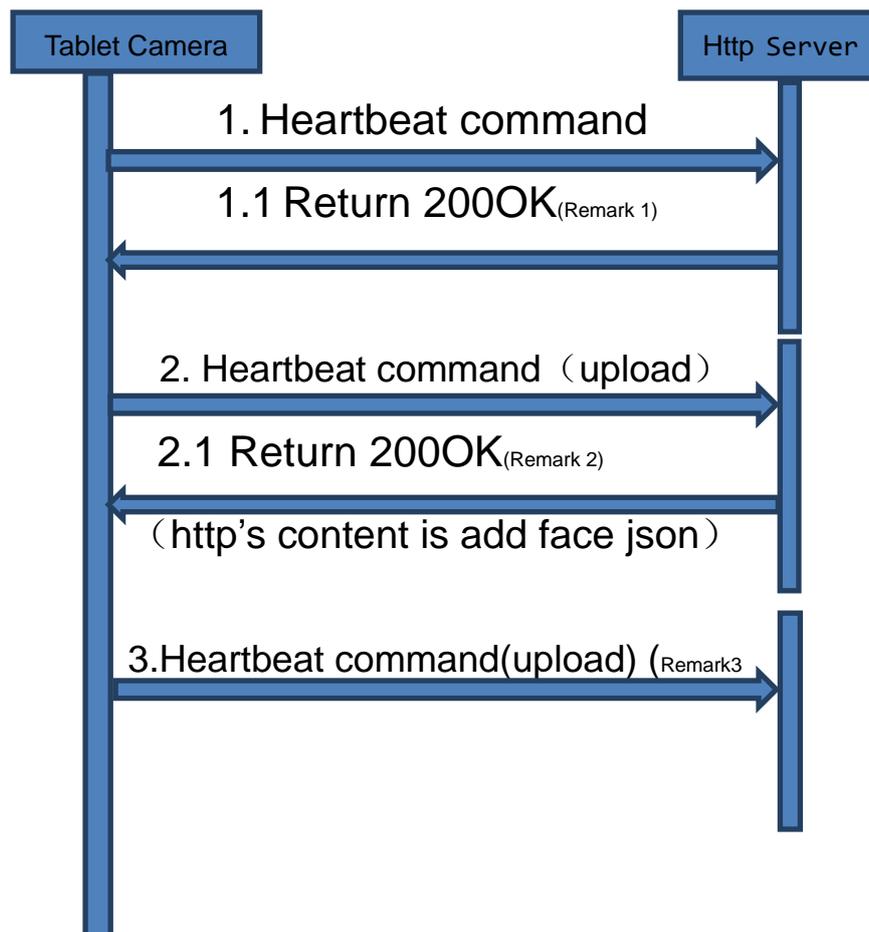
server: Set the domain name of the http server such as ccdcam.com, And access paths such as /api/face, according to the specific conditions of the http server to set.

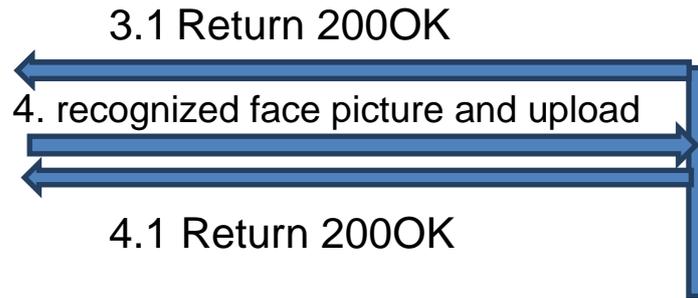
Port: Set the port number of http server access, such as 80, according to the specific conditions of the server to set.

Heartbeat time: Set the default heartbeat time, It can be modified according to the protocol in protocol communication.

Protocol communication process:

Take adding face library as an example





200 OK refers to the first line of the HTTP response header, not the data body, such as: HTTP/1.1 200 OK The tablet camera resolves to a value of 200 to indicate normal, and the rest of the values are abnormal. When abnormal, it will not continue to read the data body.

Remark 1: The http server docking uses HTTP protocol to communicate, the content is in json format, the http server needs to reply to the tablet camera with a heartbeat package, and it needs to reply with 200OK. If there is no service to be processed, the http content is empty

Note 2: The tablet camera sends a heartbeat command to the server, and the server returns the http 200OK content through the heartbeat connection to the json content that needs to be set, such as adding a face, deleting a face, etc.

Note 3: The result of the face database add and delete operation will be informed to the http server through the seq of the heartbeat protocol in the next heartbeat. If seq is equal to seq in the face database operation protocol, the operation is successful, if not equal, the operation is unsuccessful .

The protocol uses http post to send content to the http server, using http protocol header + json format content. This document only explains the json content:

2. Heartbeat command

The tablet camera sends a heartbeat command to the http server, and the server needs to reply with 200OK Description of content:

Target	Type	Description
app_version	string	Tablet camera version
cid	string	Tablet camera identification number
seq	int	Confirm the sending time of heartbeat or the success of the last face bank operation
uptime	int	UTC time for this boot
type	int	Heartbeat Protocol type=3

Example:

```
{
  "app_version": "1.0.1.1",
  "cid": "0012b516a202",
  "seq": 1569497378,
  "uptime": 1569497378,
```

```
"type": 3
}
```

Attachment: sorting summary of the type field of the reported json

Target	Type	Description
type	int	1:Snapshot and upload 3:Heartbeat 4: Face database query results 6: Upgrade push results 7: Strategy query result / configuration strategy result 8: Retransmission time query result 9:Results of batch import of personnel 10: Configuration query results / configuration delivery confirmation 11: Report the log operation result 12: Remote control

3. Tablet Camera Operating

The http server receives a heartbeat command and replies with 200OK. If there is a processing service, the content will carry the json content of the corresponding protocol.

1. Config http server address and port

The http server responds with 200OK + the following json content, the tablet camera will upload the content to the new url address

Content:

Target	Type	Description
http_server_url	string	http server url
http_server_port	int	http server port

Example:

```
{
  "http_server_url": "http://ccdcam.com/api/face",
  "http_server_port": 8080
}
```

2. Config heartbeat period

The server replies with 200OK+ the following json content, the heartbeat interval time sent by the tablet camera is changed to the set length of time.

Content:

Target	Type	Description
--------	------	-------------

time_interval	int	The interval at which the tablet camera sends heartbeat commands
---------------	-----	--

Example:

```
{
  "time_interval": 3
}
```

3.Config tablet camera time

The http server responds with 200OK+ the following json content, and the tablet camera modifies the local time of the tablet camera according to the settings.

Content:

Target	Type	Description
time_stamp	int	The server set UTC time, s
time_zone	int	The server set time zone, s

Example:

```
{
  "time_set": {
    "time_stamp": 1566459660,
    "time_zone": 28800
  }
}
```

4.Face database

Face database operations, including querying face database, adding and modifying face database, deleting face database, etc. The heartbeat cycle setting function can be used in conjunction with the face library operation, which is convenient to postpone or advance the next heartbeat time. The following examples will bring the heartbeat cycle setting, and you can remove this item if you don't need it.

Search face database

Content:

Target	Type	Description
facelib_manage	object	Mark face database
type	int	type = 5 : search face database
seq	int	Server operation serial number

Example:

```
{
  "facelib_manage": {
    "type": 5,
    "seq": 1566358435
  }
}
```

```

    },
    "time_interval": 3
}

```

Note: After the tablet camera receives the query request, the query result will be attached to the next heartbeat.

Contact:

Target	Type	Description
cid	string	Tablet caera mark number
type	int	type=4 Return query results
time	int	UTC time sent by this command
total_face	int	Number of query results
person_id	string	Person ID, maximum length 35 bytes
person_name	string	Person's name, maximum length 63 bytes
age	int	Person's age
sex	string	Person's gender, female/male
strategyId	int	Personnel strategy id
cardNo	String	Card ID, maximum length 35 bytes

Return example:

```

{
  "cid": "0012b516a202",
  "type": 4,
  "time": 1542430377,
  "total_face": 2,
  "face": [
    {
      "person_id": "20190629153538m3CH6Jdtc2DHnANaab",
      "person_name": "Zhangli",
      "age": 25,
      "sex": "male",
      "strategyId": 1,
      "cardNo": "123015"
    },
    {
      "person_id": "20190629153538MFwkbpkVerXQVCmiYi",
      "person_name": "Mingming",
      "age": 26,
      "sex": "female",
      "strategyId": 2,
      "cardNo": "123016"
    }
  ]
}

```

Add or modify

Target	Type	Description
facelib_manage	object	Mark face database
type	int	type = 1:modify face dataface
seq	int	server operation serial number
id	string	id Modify the information when it already exists on the tablet camera, otherwise add
name	string	Person's name, maximum length 35 bytes
age	int	Person's age
sex	string	Person's gender, "male", "female"
image	String array	The base64 encoding of the picture. Adding people is a must for this item. When modifying, this item can be determined according to whether the picture is modified.
strategyId	int	Personnel strategy id
cardNo	string	Card ID, maximum length 63 bytes

Example:

```
{
  "facelib_manage": {
    "type": 1,
    "seq": 1566358435,
    "face": {
      "id": "1530327995_hW7UO03",
      "name": "LILEI",
      "age": 26,
      "sex": "male",
      "strategyId": 2,
      "cardNo": "11223344",
      "image": [
        "base64data ..."
      ]
    }
  },
  "time_interval": 3
}
```

Note: The tablet camera will return the result of this operation at the next heartbeat. The next seq item of the next heartbeat is equal to the seq item of this command, indicating that the operation was successful, otherwise it was unsuccessful.

Delete face database

Content:

Target	Type	Description
facelib_manage	object	Mark face dataface
type	int	type = 3:Delete designated person face database type = 4:Delete all face database
seq	int	Server operation serial number
person_ids	string	When deleting the person information of the specified id, you can specify multiple ids at a time, separated by commas, and delete up to 20 at a time. When deleting everyone's face library, don't bring this item

Delete designated person_id:

```
{
  "facelib_manage": {
    "type": 3,
    "seq": 1566358435,
    "person_ids": "1530327995_hW7UO03, 1530327995_hW7UO04"
  },
  "time_interval": 3}

```

Delete all face database:

```
{
  "facelib_manage": {
    "type": 4,
    "seq": 1566358435
  },
  "time_interval": 3
}

```

Note: The tablet camera will return the result of this operation at the next heartbeat. The next seq item of the next heartbeat is equal to the seq item of this command, indicating that the operation was successful, otherwise it was unsuccessful.

4.Snap to upload picture information

The tablet camera detects personnel identification and uploads relevant snapshot information to the http server.

The http server receives the snapshot information and returns a 200 OK HTTP header for confirmation. The HTTP body does not need to carry data.

Note: The HTTP header and JSON content are sent twice when the tablet camera uploads the image. First send the HTTP header with the Expect: 100-continue field, wait for the http server to respond, then send the JSON, and finally wait for the server's 200

OK confirmation.

The http server can not respond to the Expect: 100-continue field, and the tablet camera will still upload the subsequent JSON.

Content:

Target	Type	Description
cid	string	Tablet camera mark number
type	int	type=1:Upload snapshot results
time	int	UTC time sent by this command
faces	String group	Snapshot uploaded image base64 encoding
result	Obj	Face information in the ratio
person_id	string	Person ID, maximum length 35 bytes
person_name	string	Person's name, maximum length 63 bytes
age	int	Person's age
sex	string	Person's gender, female/male
confidence	float	Similarity/confidence, 0 if not matched
person_type	int	0: Comparison failed 1: Compare and pass 2: Images captured at intervals
temperature	float	Temperature value (accurate to one decimal place)
facemask	int	0: no mask 1: wear a mask

Example:

```
{
  "cid": "294300000111",
  "time": 1568101332,
  "faces": [
    "base64..."
  ],
  "result": {
    "age": 0,
    "confidence": 0.9300000071525574,
    "person_id": "1567910643_dgQdv2x9",
    "person_name": "Huhao",
    "person_type": 1,
    "sex": " male"
  },
  "temperature":36.5,
  "facemask":1
}
```

5. Pass strategy

1. The working mechanism of strategy

You can flexibly manage the passage of personnel by formulating strategies.

Each policy contains multiple time rules (rule), the rules can be configured to take effect from the start and end date, week and multiple time periods, and whether to allow or prohibit the passage of time.

You can assign a strategy to each person by editing the person information, one for each person. After the tablet camera recognizes the corresponding person, it will execute the corresponding strategy. If the current time is within the strategy, it will be executed according to the policy rules. If the current time is not within the strategy, it will be banned. If the strategy to which the person belongs does not exist or the strategy is not enabled, the passage is prohibited.

Strategy compatibility design:

In order to be compatible with the old version of the protocol without strategy, the zero value of strategyId is used as a special reserved value. If a person attribute strategyId=0, he will not be restricted by strategy. The strategy with strategyId=0 will fail. However, when adding a person, it is allowed to set the strategyId of the person information to 0. If configured in this way, the person will not be restricted by the policy mechanism.

Addition of policy delivery:

The policy modification adopts the method of completely updating and covering. Every time the policy is reissued, all the previous policies will be discarded. Therefore, changing the policy needs to be completely delivered at once. Only the successfully added strategy will be retained every time it is delivered, so if none of the strategies in the one-time delivery are legal, the local strategy is cleared (for example, only one strategyId=0 is issued).

After the delivery is completed, the tablet camera will immediately report whether each strategy is added successfully (except for strategies that cannot be parsed to strategyId normally) regardless of the heartbeat interval. The result is reported in the newly initiated connection by the tablet camera

2. Strategic network protocol

Target	Type	Description
strategy_manage	{ }	Fixed fields for policy operations
strategyList	{ }	Strategy content list (maximum support 16 strategies)
strategyId	int	The unique ID of the strategy
name	string (max31)	Strategy name
state	int	Strategy status: 1 Open 0 Close

strategyRuleList	Array	The set of time rules for the strategy (maximum 16 for each strategy)
ruleId	int	Rule id
begDate	string	Start date, format: 2019/01/01
endDate	string	End date, format: 2019/11/11
timeRange	string	Time range: 0-24, use',' stitching (up to 16),example:08:00:00-08:15:00,08:15:00-12:00:00
week	string	Week: 1-7, use',' stitching (7 means Sunday), for example: 1,2,6,7
type	int	Is it allowed to pass within the time rule: 1 allow 0 Not allowed

Strategy delivery example:

```
{
  "strategy_manage": {
    "strategyList": [
      {
        "strategyId": 1,
        "name": " Test strategy 1",
        "state": 1,
        "strategyRuleList": [
          {
            "ruleId": 1,
            "begDate": "2020/03/01",
            "endDate": "2019/6/30",
            "timeRange": "17:30:00-23:59:59,00:00:00-08:30:00",
            "week": "1,2,3,4,5,6,7",
            "type": 0
          },
          {
            "ruleId": 2,
            "begDate": "2020/03/01",
            "endDate": "2019/6/30",
            "timeRange": "08:30:00-17:30:00",
            "week": "1,2,3,4,5,6,7",
            "type": 1
          }
        ]
      },
      {
        "strategyId": 2,
        "name": " Test strategy 2",
```

```

"state": 1,
"strategyRuleList": [
  {
    "ruleId": 1,
    "begDate": "2020/03/01",
    "endDate": "2019/6/30",
    "timeRange": "17:30:00-23:59:59,00:00:00-08:30:00",
    "week": "1,2,3,4,5,6,7",
    "type": 0
  },
  {
    "ruleId": 2,
    "begDate": "2020/03/01",
    "endDate": "2019/6/30",
    "timeRange": "08:30:00-17:30:00",
    "week": "1,2,3,4,5,6,7",
    "type": 1
  }
]
.....
}
}
}

```

Return strategy operation result

Target	Type	Description
cid	String	Tablet camera id
type	int	The equipment reply operation result is fixed type=7
time	int	Time
strategy_manage	{}	Fixed field
resultList	array	Operation result list
strategyId	int	The unique ID of the strategy
result	int	Operation result: 0: ok 1: fail

Example

```

{
  "cid": "042082065161",
  "type": 7,
  "time": 1584424711,

```

```

"strategy_manage": {
  "resultList": [
    {
      "strategyId": 1,
      "result": 0
    },
    {
      "strategyId": 2,
      "result": 0
    },
    ...
  ]
}

```

Strategy query

The server issues a policy query request in response to the tablet camera 's heartbeat time

Target	Type	Description
strategy_manage	{}	Fixed field
operType	int	Fixed field:2

Query strategy delivery example:

```

{
  "strategy_manage": {
    "operType": 2
  }
}

```

The tablet camera will carry the contents of the policy list at the next heartbeat time

Target	Type	Description
cid	String	Tablet camera id
type	int	The tablet camera reply to the query result is fixed type=7
time	int	Time
total_strategy	int	Strategy quantity
Strategy List	Array	Strategy list
strategyId	int	Strgtegy Unique identification code
name	string	Strgtegy name
state	int	Strgtegy status: 1 open 0 close
strategy Rule List	Array	Collection of time rules for strategies (max 16)

ruleId	int	Rule id
Beg Date	string	Start data, format: 2019/01/01
End Date	string	End data, format: 2019/11/11
Time Range	string	Time range: 0-24, use',' stitching (up to 16), example : 08:00:00-08:15:00,08:15:00-12:00:00
week	string	Week: 1-7, use',' stitching (7 means Sunday), for exampl: 1,2,6,7
type	int	Is it allowed to pass within the time rule: 1 allow 0 Not allowed

Query return example:

```
{
  "cid": "294300000005",
  "type": 7,
  "time": 1542430377,
  "total_strategy": 2,
  "strategyList": [
    {
      "strategyId": 1,
      "name": "Strategy1",
      "state": 1,
      "strategyRuleList": [
        {
          "ruleId": 1,
          "begDate": "2020/03/01",
          "endDate": "2020/04/02",
          "timeRange": "07:30:40-10:30:20",
          "week": "6,7",
          "type": 0
        },
        {
          "ruleId": 2,
          "begDate": "2020/03/03",
          "endDate": "2020/05/04",
          "timeRange": "07:30:00-08:30:00,07:30:00-08:30:00",
          "week": "1,2,3,4,5",
          "type": 1
        }
      ]
    }
  ],
  {
    "strategyId": 2,
```

```

"name": "test_strategy_2",
"state": 0,
"strategyRuleList": [
  {
    "ruleId": 1,
    "begDate": "2020/03/01",
    "endDate": "2020/04/02",
    "timeRange": "07:30:40-10:30:20",
    "week": "1,2,3",
    "type": 1
  },
  {
    "ruleId": 2,
    "begDate": "2020/03/03",
    "endDate": "2020/05/04",
    "timeRange": "8:59:00-9:30:00,18:29:00-19:00:00",
    "week": "4,5,6",
    "type": 1
  }
]
},
.....
]
}

```

6. Pass record failure timing retransmission

Snapshots of failed uploading pass records will be saved and re-transmitted by machine. The tablet camera can save up to 2,000 failed picture records locally, and the oldest ones will be discarded when the limit is exceeded.

There are two modes of retransmission:

1. Scheduled retransmission mode: after the scheduled time is reached, retransmit one by one in the order from old to new. If it fails, skip and continue to transmit the next one. The failed picture will still be retained. Wait until the next retransmission time comes pass.
2. Idle retransmission mode: When no new pass record is generated within 3 minutes, it will be retransmitted, retransmitting one by one in the order from old to new. If it fails, wait for the next try. During the upload process, if there are people passing by, the retransmission stops and waits for the next free 3 minutes.

Note: The time field in the picture information during retransmission does not recalculate the time, it is still the time when the pass record was generated. Modify setting:

Target	Type	Description
reupload	{ }	Fixed fields for retransmission operations

type	int	Operation type: 1 Modify setting 2 Search setting
seq	int	Operation serial number
reupload_time	String	Timed retransmission time, h:m:s
mode	int	Retransmission method: 1- Scheduled retransmission 2- 3 minute idle retransmission-[default]

Setting example:

```
{
  "reupload": {
    "type": 1,
    "seq": 11223456,
    "reupload_time": "19:45:10",
    "mode": 2
  }
}
```

Return the modified result

The tablet camera will return the result immediately after modifying the settings:

Target	Type	Description
type	Int	Fixed type=8
cid	string	Tablet camera Identification Number
time	int	Sending time
reupload	{ }	Fixed field
seq	int	At the time of delivery seq
errcode	int	0- success 1- Person parameter erro

Return example:

```
{
  "cid": "042071005218",
  "type": 8,
  "time": 1586159775,
  "reupload": {
    "seq": 11223456,
    "errcode": 0
  }
}
```

Search example:

```
{
  "reupload": {
```

```

    "type": 2
  }
}

```

Example query results:

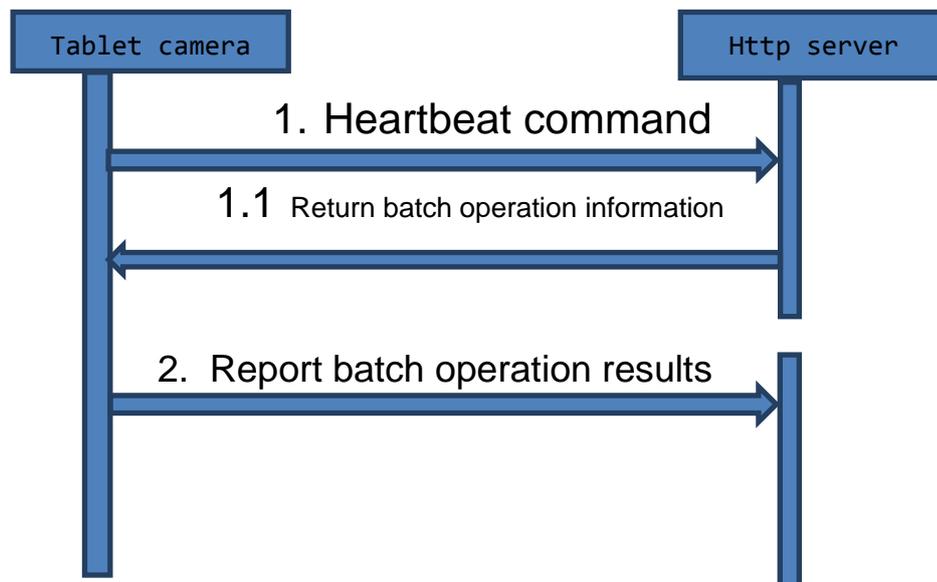
Target	Type	Description
type	Int	Fixed :type=8
cid	string	Tablet camera Identification Number
time	int	Sending time
reupload	{ }	Fixed field
reupload_time	String	Timed retransmission time, hour: minute: second
mode	int	Retransmission method: 3- Scheduled retransmission 4- 3 minutes idle retransmission-[default]
recordes_num	int	Number of retransmissions saved locally on the tablet camera

```

{
  "cid": "042071005218",
  "type": 8,
  "time": 1586151346,
  "reupload": {
    "reupload_time": "00:00:00",
    "mode": 2,
    "recordes_num": 28
  }
}

```

7. Person Batch Import



2.1 Return batch operation information



Protocol process

1. The tablet camera reports a normal heartbeat, and when the server responds to the heartbeat, it sends out personnel information in batches;
2. The equipment batch operation personnel library, after the operation is completed, the equipment will actively report the operation results regardless of the heartbeat interval;
3. The server continues to deliver after receiving the report, and so on, and the tablet camera resumes the heartbeat interval after the push is completed.

Remarks: The batch personnel operation is still a short connection, and the tablet camera reports once and the server delivers once to complete the connection.

The tablet camera will report the results of each personnel operation (except for the failure to parse the personnel id field).

server release

The server delivers personnel information and operations to the tablet camera in batches:

The id, strategyId and image of personnel information are required fields when importing.

If strategy is not needed, it can be set to zero "strategyId":0

Target	Type	Description
facelib_batch	{ }	Fixed field
personList	Array	Personnel list, support batch operation of multiple pieces of personnel information, limit to send a maximum of 20 at a time
operType	Int	Operation type: 1 The person who does not exist in the personnel library is added, and the person who is present is modified 3 Delete (If it is deleted, only the id field in the personnel information is valid)
id	String (max35)	Person id, a unique identification code for each person
cardNo	String (max35)	Card ID
name	String (max63)	Name
age	Int	Age
sex	String	Gender"male", "female"
strategyId	int	Personnel strategyid
image	String	Base64 Encoded picture (Note: Unlike single face import, square brackets are not used here)

Example:

Example of personnel information delivered by the tablet camera:

```

{
  "facelib_batch": {
    "personList": [
      {
        "operType": 1,
        "id": "1530327995_hW7UO01",
        "cardNo": "123",
        "name": "name1",
        "age": 24,
        "sex": "female",
        "strategyId": 1,
        "image": "base64data ..."
      },
      {
        "operType": 3,
        "id": "1530327995_hW7UO02"
      },
      {
        "operType": 1,
        "id": "1530327995_hW7UO03",
        "cardNo": "789",
        "name": "name3",
        "age": 16,
        "sex": "female",
        "strategyId": 3,
        "image": "base64data ..."
      },
      .....
    ]
  }
}

```

Tablet camera reply

The tablet camera replies to the import result

Target	Type	Description
cid	String	Tablet camera id
type	int	The tablet camera reply import result is fixedtype=9
time	int	Time
facelib_batch	{}	Fixed field
resultList	array	Operation result list
operType	Int	Operation type: 1 The person who does not exist in the

		personnel library is added, and the person who is present is modified 3 Delete (If it is deleted, only the id field in the personnel information is valid)
id	String	Person id, Unique identification code for each person
result	int	Operation result: 0: Success 1: failed 2: The personnel information parameter is wrong 3: wrong picture 4: The staff pool is full 5: Timeout to get import result 6: The system is busy

Example:

```
{
  "cid": "042082065161",
  "type": 9,
  "time": 1584424711,
  "facelib_batch": {
    "resultList": [
      {
        "operType":1,
        "id": "1530327995_hW7UO01",
        "result":0
      },
      {
        "operType":3,
        "id": "1530327995_hW7UO02",
        "result":1
      },
      .....
    ]
  }
}
```

8. Software upgrade push

Protocol process:

1. The tablet camera reports the heartbeat normally, and the http server issues the

- upgrade package version number and download address in the heartbeat reply;
2. The tablet camera pauses heartbeat and uploads with snapshots;
 3. The tablet camera verifies the version number. If the version number is abnormal, it returns an error message and resumes heartbeat and snapshot upload;
 4. The tablet camera connects to the download address to obtain the upgrade package. If the download fails, an error message is returned, and the heartbeat and snapshot upload are restored;
 5. After receiving the upgrade package, the tablet camera will reply to the http server with confirmation information;
 6. The tablet camera is upgraded and restarted.

Upgrade commands issued by the server

Target	Type	Description
fw_upgrade	{}	Fixed fields for upgrade operations
version	String	version number
download_url	String	Upgrade package download address

Example:

```
{
  "fw_upgrade": {
    "version": "V1.2.1.2",
    "download_url": "http://xxxxxxx/xxx.zip"
  }
}
```

After the update, the tablet camera returns immediately without waiting for the heartbeat:

Target	Type	Description
app_version	string	Tablet camera version number
cid	string	Tablet camera identification number
seq	int	Set the heartbeat sending time or the confirmation of the last successful face bank operation
uptime	int	UTC time for this boot
type	int	Update reply message fixed type=6
error	Int	error code: 1: indicates that the upgrade server cannot be connected 2: Can't get files 3: The file is too large 4: The upgrade connection is interrupted 5: The upgrade file does not match 6: ready to restart after the upgrade

Example:

```
{
  "app_version": "1.0.1.1",
  "cid": "0012b516a202",
```

```

"seq": 1569497378,
"uptime": 1569497378,
"type": 6,
"error": 6
}

```

9.parameter configuration

Process:

1. The tablet camera reports the heartbeat, and the configuration parameters are issued when the server replies to the heartbeat;
2. The tablet camera applies configuration parameters, and the next heartbeat reply to the http server for confirmation.

Parameter delivery

Target	Type	Description
config	{}	Fixed fields of configuration parameters
opType	Int	Operation type: 0: configuration parameter 1: Query parameters (the following fields are not filled in during query)
seq	int	Operation serial number, required field when opType=0
face_rec	{}	Face recognition configuration items
face_cap	{}	Capture configuration items
facemask	{}	Mask configuration items
temperature	{}	Body temperature configuration items
relay	{}	Relay configuration items
misc	{}	Miscellaneous

Face recognition configuration items "face_rec":{}

Target	Type	Description
enable	int	Face recognition switch: 0: close 1: On-[default]
voice	int	Voice broadcast 0: No broadcast 1: Full broadcast ("authorized", "unauthorized") 2: Abnormal broadcast ("Unauthorized")-[default]
relay*	int	Related: 0: not related

		1: Association, open the door without authorization-[default]
liveness	float	Living threshold for face recognition (0-1, the larger the value, the more likely it is to be living)-[default 0.2]
score	float	Face recognition score (similarity) threshold (0-100) -[Default 70]
times	int	The threshold of the number of times that the same person is identified (1-10), that is, the number of times that the threshold of the identification score is not reached, is determined to be a stranger after being exceeded. -[Default 3]
distance	int	Recognition distance threshold: 0: 0.5m 1: 1m - [default] 2: 1.5m 3: 2m 4: 2.5m
face_expos	int	Face exposure threshold (0-100)-[default 50]

Face capture “face_cap”:{}

Target	Type	Description
interval	int	Time interval for taking pictures, in seconds-[default 3]
goodness	float	Capture picture quality parameters (0-1, the larger the value, the better the quality) -[Default 0.7]
blur	float	Capture image blur parameters (0-1, the smaller the value, the clearer) -[Default 0.3]
time_overlay	int	Snapshot picture overlay time information: 0: no overlay-[default] 1: Overlay
upload_mode	int	Common snapshot upload mode: 1: Only upload the bottom library personnel 2: Upload the bottom library personnel and strange visitors-[default]

Mask configuration items “facemask”:{}

Target	Type	Description
enable	int	Mask detection switch: 0: close 1: On-[default]
mode	int	Mask detection mode options: 0: No masks (epidemic mode)-[default] 1: Wear a mask (non-epidemic mode)
voice	int	Voice broadcast 0: No broadcast 1: Abnormal broadcast (in epidemic mode "please wear mask", non-epidemic mode "please remove mask")-[default]
relay	int	Related: 0: not related-[default] 1: related, the mask does not open the door abnormally

Body temperature configuration items “temperature”:{}

Target	Type	Description
enable	int	Body temperature detection switch: 0: close 1: On-[default]
high_threshold	float	High temperature threshold-[default 37.3]
voice	int	Voice broadcast: 0: No broadcast 1: Only broadcast exception-[default] 2: Temperature value broadcast, abnormal broadcast
relay	int	Related: 0: not related 1: Correlation, abnormal temperature does not open the door-[default]
offset	float	Temperature compensation coefficient
unit	int	Temperature unit: 0: Celsius-[default] 1: Fahrenheit
environment	int	Temperature measurement scene: 0-Indoor-[default] 1-outdoor

Relay item "relay": {}

Target	Type	Description
duration	int	Relay signal duration (1-100) (unit: one hundred milliseconds)- default [2]
state	int	The working state of the relay: 0: Connect when the status is normal-[default] 1: Turn on when the status is abnormal

Miscellaneous "misc":{}

Target	Type	Description
version	int	Configuration parameter version (can only be queried, not configurable)

Distribution configuration example :

```
{
  "config": {
    "opType": 0,
    "seq": 123456,
    "face_rec": {
      "enable": 1,
      "voice": 1,
      "relay": 1,
      "liveness": 0.2,
      "score": 70,
      "times": 5,
      "distance": 1,
      "face_expos": 50
    },
    "face_cap": {
      "interval": 10,
      "goodness": 0.7,
      "blur": 0.3,
      "time_overlay": 0,
      "upload_mode": 1
    },
    "facemask": {
      "enable": 1,
      "mode": 0,
      "voice": 1,
      "relay": 0,
      "environment": 0
    }
  },
}
```

```

    "temperature": {
        "enable": 1,
        "high_threshold": 37.3,
        "voice": 1,
        "relay": 1,
        "offset": 0,
        "unit": 0
    },
    "relay": {
        "duration": 2,
        "state": 0
    }
}
}
}

```

The tablet camera returns an error code:

The tabket camnera will immediately report the error code regardless of the heartbeat interval.

Target	Ttype	Description
cid	string	Tablet camera identification number
type	int	Configure the fixed fields returnedtype=10
time	int	Sending time
config	{}	Fixed field
seq	int	The delivered seq, if there is no resolution to the delivered seq, there is no field in the return
errcode	int	error code: 0: Success 1: Incoming error (incoming errors in seq and opType fields are also classified here) 2: configuration failed

Example:

```

{
    "cid": "042082065161",
    "type": 10,
    "time": 1585047588,
    "config": {
        "seq": 1584940620,
        "errcode": 0
    }
}
}

```

Search example:

```
{
  "config": {
    "opType": 1
  }
}
```

The query results will be reported at the next heartbeat time

Target	Ttype	Description
type	int	Fixed when returning to configuration query type=10
cid	string	Tablet camera identification number
time	int	Time

Example query results:

```
{
  "cid": "042082065161",
  "type": 10,
  "time": 1584945022,
  "config": {
    "face_rec": {
      "enable": 1,
      "voice": 1,
      "relay": 1,
      "liveness": 0.20,
      "score": 70.00,
      "times": 5,
      "distance": 1,
      "face_expos": 50
    },
    "face_cap": {
      "interval": 10,
      "goodness": 0.70,
      "blur": 0.30,
      "time_overlay": 0,
      "upload_mode": 1
    },
    "facemask": {
      "enable": 1,
      "mode": 0,
      "voice": 1,
      "relay": 0
    }
  },
}
```

```

    "temperature": {
        "enable": 1,
        "high_threshold": 37.30,
        "voice": 1,
        "relay": 1,
        "offset": 0.00,
        "unit": 0,
        "environment":0
    },
    "relay": {
        "duration": 2,
        "state": 0
    },
    "misc": {
        "version": 1
    }
}
}
}

```

10. Log operations

Issue operation instructions

Target	Ttype	Description
facelog	{}	Fixed field
opType	int	Operation type: 1: Clear the log
seq	int	Operation serial number

Distribution example:

```

{
    "facelog": {
        "opType": 1,
        "seq": 123456
    }
}

```

Return confirmation

Target	Ttype	Description
cid	string	Tablet camera identification number
type	int	Fixed fields returned by log operationstype=11
time	int	Sending time
facelog	{}	Fixed field

errcode	int	0: Success 1: Input error 2: failed
seq	int	The delivered seq, if there is no resolution to the delivered seq, there is no field in the return

Example:

```
{
  "cid": "042082065161",
  "type": 11,
  "time": 1584945022,
  "facelog": {
    "seq": 123456,
    "errcode": 0
  }
}
```

11. Tabelt camera reboot:

Distribution example{

```
  "reboot": 1
}
```

12. Tablet camera control instruction:

Issue operation instructions

Target	Type	Description
control	{ }	Fixed field
opType	int	Operation type: 1: Switch the relay once
seq	int	Operation serial number

Distribution example:

```
{
  "control": {
    "seq": 12123,
    "opType": 1
  }
}
```

Return confirmation

Target	Type	Description
cid	string	Operation serial number

type	int	Fixed fields returned remotely type=12
time	int	Sending time
control	{ }	Fixed field
errcode	int	0: Success 1: Input error 2: failed
seq	int	The delivered seq, if there is no resolution to the delivered seq, there is no field in the return

Examples:

```
{
  "cid": "042082065161",
  "type": 12,
  "time": 1584945022,
  "control": {
    "seq": 12123,
    "errcode": 0
  }
}
```

13.Alarm report:

Alarm data report

Target	Type	Description
alarm	{ }	Fixed field
seq	int	Operation sequence number
time	int	Timestamp
alarmtype	int	Alarm type: 1: Door sensor alarm;
alarmdata	string	Reserved

示例:

```
{
  "alarm": {
    "seq": 12123,
    "time": 175682364,
    "alarmtype": 1,
    "alarmdata": ""
  }
}
```